Computational Methods in Applied Sciences

Edmondo Minisci · Massimiliano Vasile
Jacques Periaux · Nicolas R. Gauger
Kyriakos C. Giannakoglou
Domenico Quagliarella *Editors*

Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences

ECCOMAS
European Community
on Computational Methods
in Applied Sciences

Springer

Computational Methods in Applied Sciences

Volume 48

Series editor

E. Oñate
CIMNE
Edificio C-1, Campus Norte UPC
Gran Capitán, s/n.
08034 Barcelona, Spain
onate@cimne.upc.edu

More information about this series at http://www.springer.com/series/6899

Edmondo Minisci · Massimiliano Vasile
Jacques Periaux · Nicolas R. Gauger
Kyriakos C. Giannakoglou
Domenico Quagliarella
Editors

Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences

Springer

Editors

Edmondo Minisci
Aerospace Centre of Excellence
University of Strathclyde
Glasgow
UK

Massimiliano Vasile
Aerospace Centre of Excellence
University of Strathclyde
Glasgow
UK

Jacques Periaux
CIMNE
Universitat Politècnica de Catalunya
Barcelona
Spain

Nicolas R. Gauger
Chair for Scientific Computing
TU Kaiserslautern
Kaiserslautern
Germany

Kyriakos C. Giannakoglou
School of Mechanical Engineering
National Technical University of Athens
Athens
Greece

Domenico Quagliarella
Fluid Mechanics Department,
Multidisciplinary Analysis and
Optimization Group
Italian Aerospace Research Center
Capua
Italy

ISSN 1871-3033
Computational Methods in Applied Sciences
ISBN 978-3-319-89986-2 ISBN 978-3-319-89988-6 (eBook)
https://doi.org/10.1007/978-3-319-89988-6

Library of Congress Control Number: 2018938789

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

EUROGEN 2015 was the 11th of a series of International Conferences devoted to bringing together specialists from Universities, Research Institutions and Industries developing or applying Evolutionary and Deterministic Methods in Design Optimization, with emphasis on solving industrial and societal problems.

The conference was organised around a number of parallel symposia, regular sessions and keynote lectures and was open to everybody who wanted to present recent advancements on optimization methods and applications. In particular, the conference focused on:

- Surrogate-Based Optimization in Aerodynamic Design
- Adjoint Methods for Steady and Unsteady Optimization
- Multi-disciplinary Design Optimization
- Holistic Optimization in Marine Design
- Game Strategies Combined with Evolutionary Computation
- Optimization under Uncertainty
- Topology Optimization
- Optimal Planning
- Shape Optimization
- Production Scheduling

EUROGEN-2015 was an ECCOMAS Thematic Conference, organized by the University of Strathclyde, in association with ECCOMAS and ERCOFTAC. The conference took place at University of Strathclyde's flagship Technology and Innovation Centre (TIC), in Glasgow (UK), 14–16 September, 2015. It attracted nearly 100 delegates, with almost 40% of the attendees being postgraduate students.

Among the 79 contributions to the conference, 33 extended full papers were selected for publication in this volume after peer review by the members of the organizing committee. The 33 papers are grouped into the following parts:

- Surrogate-Based Optimization in Aerodynamic Design
- Adjoint Methods for Steady and Unsteady Optimization
- Holistic Optimization in Marine Design

- Game Strategies Combined with Evolutionary Computation
- Optimization Under Uncertainty
- Algorithms and Industrial Applications

This volume presents up-to-date material on the state of the art in Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems from Europe, Asia and America.

The Scientific Organizing Committee and the Local Organizing Committee acknowledge the sponsorship of the following organizations through financial support or/and assistance during the development of the event: European Community on Computational Methods in Applied Sciences (ECCOMAS), European Research Community on Flow, Turbulence and Combustion (ERCOFTAC), IEEE Computational Intelligence in Aerospace (IEEE-CIAS), Future Air-Space Transportation Technology Laboratory (FASTT-Lab), OPTIMAD Engineering Srl, University of Strathclyde.

Special thanks are addressed to all the members of the European Scientific Committee, and the International Corresponding members.

Finally, the editors acknowledge Nathalie Jacobs (Springer) and Eugenio Oñate for the interest to this series in publishing the most representative scientific and industrial material presented in the EUROGEN 2015 ECCOMAS Thematic Conference in the Springer—ECCOMAS Series entitled: Computational Methods in Applied Sciences, and Anneke Pot (Springer) for the infinite patience and the invaluable support to put together the book.

Glasgow, UK	Edmondo Minisci
Glasgow, UK	Massimiliano Vasile
Barcelona, Spain	Jacques Periaux
Kaiserslautern, Germany	Nicolas R. Gauger
Athens, Greece	Kyriakos C. Giannakoglou
Capua, Italy	Domenico Quagliarella

Contents

Part I Keynote

Part II Surrogate-Based Optimization in Aerodynamic Design

Part VI Optimisation Under Uncertainty

Part VII Algorithms and Industrial Applications

Part I
Keynote

Risk, Optimization and Meanfield Type Control

Olivier Pironneau and Mathieu Laurière

Abstract Risk is usually a criteria which involves the world's state; for instance the best policy to extract oil from a well of finite resource depends on the price of oil which in turn depends on how much the world's oil extractors produce. Many optimization of systems with respect to profit and risk involve a very large number of players who optimize the same criteria. Then the profit is the result of a global optimization problem, which is coupled with a each player's system design where price appears as a passive variable. Meanfield type control is a mathematical tool which can help solve such problem in the presence of randomness, an aspect essential for the modeling of risk. We shall give a few examples and compare solutions by calculus of variations plus gradient algorithms with extended dynamic programming and fixed point.

1 Introduction

Risk quantification is important nowadays; several Risk Institutes have appeared in universities around the world. Its emergence as a criterium of optimization in new fields seems to be due to the computing power at our disposal, compatible now with solution of large nonlinear systems with uncertainties.

However Risk in an old concept for insurance companies and in banking. Risk is less when the number of contracts is large in insurance and yet the probability of ruin of the company is still an important index to compute [1].

In Banking risk is everywhere: risk of default, credit risk [10], risk of propagation of default, etc. Basel III imposes some stress tests to banks; they require to compute

O. Pironneau (✉)
Laboratoire Jacques-Louis Lions (LJLL), Sorbonne Université, UPMC Univ Paris 06,
UMR 7598, Boite courrier 187, 75252 Paris Cedex 05, France
e-mail: olivier.pironneau@upmc.fr

M. Laurière
Math dept. Université Denis Diderot, Paris, France
e-mail: mathieu.lauriere@gmail.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_1

the CVA (credit valuation adjustment [3]) of each financial asset and derivative in the banks' portfolio leading to very time consuming computations and optimisation problems.

Risk of failure of an engineering system requires the probability of rare events when the system is subjected to random data; uncertainty quantification [9], while still computationally very demanding, is routinely practiced in some industries.

It is also possible to combine engineering design optimization with financial risk analysis. For instance a set of wind mills is planed in an area where the wind is random [8]. There are many design parameters such as their shape, position and numbers; each have a cost, an efficiency and a longevity which have to be optimized in the face of alternative energy sources, themselves subject to uncertainties [11]. When there are many of these optimizable wind energy fields, they have a global effect on the price of energy.

Several new mathematical studies have focused on problems where a system is made of a large number of similar units which contributes to the properties of the global system and the optimization is made both at the unit and global levels. Game theory is the appropriate tool and the global system is called a meanfield game [21, 23]. Optimization of such systems leads to so called meanfield type control problems [6].

2 A General Framework for Meanfield Type Control

Consider the problem of minimizing $J(u) := J_0(u)$ with $u \in \mathcal{U}_d$ and

$$J_\tau(u) := \int_\tau^T \mathbb{E}[\tilde{H}(X_t, t, u(X_t, t), \mathbb{E}[\tilde{h}(X_t, t, u(X_t, t))])]\mathrm{d}t + \mathbb{E}[G(X_T, \mathbb{E}[g(X_T)])] \tag{1}$$

$$\mathcal{U}_d = \{u \ : \ u(x,t) \in \mathcal{V}_d \ \forall x, t\} \text{ for some } \mathcal{V}_d \subset \mathbb{R}^d. \tag{2}$$

$$\mathrm{d}X_t = u(X_t, t)\mathrm{d}t + \sigma(X_t, t)\mathrm{d}W_t, \quad t \in [0, T], \quad X_0 \text{ given} \tag{3}$$

where $u(x,t) \in \mathbb{R}^d$, $\sigma(X_t, t) \in \mathbb{R}^{d\times k}$ and W_t is a k-vector of independent Brownian motions and where $\tilde{h}$, g, $\tilde{H}$, G are C^1 functions taking values in $\mathbb{R}^r$, $\mathbb{R}^s$, $\mathbb{R}$ and $\mathbb{R}$, respectively.

Under some regularity assumptions detailed in [22], (3) can be replaced by the Fokker-Planck equation for the probability density (PDF) of X_t,

$$\partial_t \rho + \nabla \cdot (u\rho) - \nabla^2 : \left(\tfrac{1}{2}\sigma\sigma^T \rho\right) = 0, \quad \rho|_{t=0} = \rho_0(\cdot). \tag{4}$$

where ∇^2 is the $d \times d$ matrix operator of element ∂_{ij}. The notation $A : B$ stands for $\sum_{i,j=1}^d A_{ij}B_{ij}$ and $\nabla \cdot u$ stands for $\sum_{i=1}^d \partial_i u_i$. Hence the following problem is similar to the stochastic optimization of (1):

$$\min_{u\in\mathscr{U}_d} J=\int_0^T\int_{\mathbb{R}^d} H(x,t,u(x,t);\rho_t)\rho(x,t)\mathrm{d}x\mathrm{d}t+\int_{\mathbb{R}^d} G(x,\xi)\rho(x,T)\mathrm{d}x,$$

where $H(x,t,u;\rho_t):=\tilde{H}(x,t,u,\chi(t))$ with (4) and

$$\chi(t):=\int_{\mathbb{R}^d} h(x,t,u(x,t),\rho(x,t))\rho(x,t)\mathrm{d}x,\quad \xi:=\int_{\mathbb{R}^d} g(x)\rho(x,T)\mathrm{d}x \tag{5}$$

Notice that H has a functional dependency on $x\mapsto\rho_t(x)$. Here (5) is more general than (1); they match when $h=\tilde{h}$ is not a function of ρ. Define the Quadratic Programming value function by

$$V(\tau;\rho_\tau):=\min_{u\in\mathscr{U}_d}\int_\tau^T\int_{\mathbb{R}^d} H(x,t,u(x,t);\rho_t)\rho_t(x)\mathrm{d}x\mathrm{d}t+\int_{\mathbb{R}^d} G(x;\rho_T)\rho_T\mathrm{d}x,. \tag{6}$$

where ρ_t is the solution of (4) on $(0,T]$ with initial condition ρ_τ at τ. Let $\hat{u}\in\mathscr{U}_d$ be a solution and $\hat{\rho}$ the corresponding solution of (5). Let $V'(x,\tau;\rho)$ be the Rietz representative of the derivative of V with respect to ρ, i.e.

$$V(x,\tau;\rho+\nu)=V(x,\tau;\rho)+\int_{\mathbb{R}^d} V'(\tau;\rho)(x)\nu(x)\mathrm{d}x+o(||\nu||_2),\ \forall\nu\in L^2(\mathbb{R}^d);$$

Proposition 1 (Extended Hamilton-Jacobi-Bellman Principle)

$$0=\min_{v\in\mathscr{V}_d}\int_{\mathbb{R}^d}\Big[H(x,\tau,v;\rho_\tau)+\rho_\tau\partial_\rho\tilde{H}+\Big(\int_{\mathbb{R}^d}\partial_\chi\tilde{H}\rho_\tau\mathrm{d}x\Big)(h+\rho_\tau\partial_\rho h) \tag{7}$$

$$+\ \partial_\tau V'+v\cdot\nabla V'+\mu:\nabla^2 V'\Big]\rho_\tau\mathrm{d}x;\qquad V'_{|T}=G+g\int_{\mathbb{R}^d}\partial_\xi G\hat{\rho}_T\mathrm{d}x. \tag{8}$$

Remark 1 V' is related to V at the optimum by

$$\int_{\mathbb{R}^d} V'(\tau;\rho_\tau)\rho_\tau\mathrm{d}x=\int_\tau^T\int_{\mathbb{R}^d}\Big(\rho_t\partial_\rho\tilde{H}+\Big(\int_{\mathbb{R}^d}\partial_\chi\tilde{H}\rho_t\mathrm{d}x\Big)(h+\rho_t\partial_\rho h)\Big)\rho_t\mathrm{d}x\mathrm{d}t$$

$$+\int_{\mathbb{R}^d}\Big(\Big(G+g\int_{\mathbb{R}^d}\partial_\xi G\rho_T\mathrm{d}x\Big)\cdot\rho_T\Big)\rho_T\mathrm{d}x+V(\tau;\rho_\tau). \tag{9}$$

3 Portfolio Optimization

A portfolio of value x_t is made of a risky asset S^1_t and cash in a saving account S^0_t. These are given by

$$\mathrm{d}S^0_t = aS^0_t\mathrm{d}t, \quad \mathrm{d}S^1_t = \alpha S^1_t\mathrm{d}t + \sigma S^1_t\mathrm{d}W_t \tag{10}$$

where a is the interest rate at time t, α is the drift of S^1, σ its volatility, W_t a standard Brownian motion. Following [2], assume the amount invested in the risky asset at time t is worth v_t; the rest is $x_t - v_t$ in cash; over a period dt it changes dx_t is due to the change dS^1_t, i.e. $v_t(\alpha\mathrm{d}t + \sigma\mathrm{d}W_t)$ plus the change due to dS^0_t: $(x_t - v_t)a\mathrm{d}t$:

$$\mathrm{d}x_t = (x_t - v_t)a\mathrm{d}t + v_t(\alpha\mathrm{d}t + \sigma\mathrm{d}W_t) \tag{11}$$

A widely accepted definition of an optimally managed portfolio, in the sense of risk versus yield, is when $\{v_t\}_0^T$ minimizes of the mean variance $2J := \mathbb{E}[x_T^2 - (\mathbb{E}[x_T])^2]$ at time T of exercise while maximizing the mean $\mathbb{E}[x_T]$; the 2 is there for numerical convenience. For each Nash equilibrium there will be a constant γ such that $\{v_t\}_0^T$ minimizes $J := \frac{\gamma}{2}\mathbb{E}[x_T^2 - (\mathbb{E}[x_T])^2] - \frac{1}{2}\mathbb{E}[x_T]$. This is a non-standard stochastic control problem because J is function not only of x_T but also of $\mathbb{E}[x_T]$; it is known as a mean-field type control problem. However it can be converted into a deterministic control problem by introducing the probability density function of x_t, which is solution of the Fockker-Planck equation associated with (11):

$$\partial_t\rho + \partial_x[(ax + bv)\rho] - \partial_{xx}[\frac{\sigma^2 v^2}{2}\rho] = 0, \quad x \in \Omega := \mathbb{R}^+, \quad t \in (0, T] \tag{12}$$

where $b := \alpha - a$ and $\rho_{|0}$ is given, positive with unit mean. The problem is then to

$$\min_{v(x,t)\in\mathscr{V}} J = \frac{1}{2}\int_\Omega (\gamma x^2 - x)\rho_T\mathrm{d}x - \frac{\gamma}{2}\left[\int_\Omega x\rho_T\mathrm{d}x\right]^2 \text{ subject to (12).} \tag{13}$$

We assume that v is a feedback function $x, t \to v(x, t)$; there are bounds on v, for instance, $\mathscr{V} = \{v : m \le v(x, t) \le M, \ \forall x, t\}$. The problem deviates slightly from framework [20] in that the control appears in the volatility but the methodology is the same. With the same notations $d = 1$ and

$$\tilde{H} = 0, \ \tilde{h} = 0, \ G = \frac{1}{2}(\gamma x^2 - x - \gamma x\int_\Omega x\rho_T\mathrm{d}x), u = ax + bv, \ \mu = \frac{\sigma^2 v^2}{2} \tag{14}$$

An adjoint state $\rho^* := V'$ is introduced, solution of

$$\partial_t\rho^* + (ax+bv)\partial_x\rho^* + \frac{\sigma^2 v^2}{2}\partial_{xx}\rho^* = 0,\ \rho^*_{|T} = \frac{1}{2}(\gamma x^2 - x) - \gamma x\int_{\mathbb{R}} x\rho_T \mathrm{d}x \tag{15}$$

with $\rho^*(\pm\infty) = 0$. The optimality conditions are, for all admissible δv,

$$\delta J = \int_{\mathbb{R}\times(0,T)} [(b\partial_x\rho^* + \sigma^2 v\partial_{xx}\rho^*)\rho\delta v] \tag{16}$$

It leads to

$$\begin{aligned} &b\partial_x\rho^* + \sigma^2 v\partial_{xx}\rho^* = 0\ \ \forall x,t\ \text{ where } m < v(x,t) < M\\ &b\partial_x\rho^* + \sigma^2 v\partial_{xx}\rho^* \geq 0\ \ \forall x,t\ \text{ where } v(x,t) = m,\\ &b\partial_x\rho^* + \sigma^2 v\partial_{xx}\rho^* \leq 0\ \ \forall x,t\ \text{ where } v(x,t) = M \end{aligned} \tag{17}$$

3.1 Polynomial Solution

Assume $\rho^* = qx^2 + rx + s$ and $v = Ax + B \in (m, M)$. Then the adjoint equation gives solvable ODEs for $q(t)$, $r(t)$ and $s(t)$. Because of the constraints, the general solution has 3 regimes as shown on Fig. 1.

Proof In what follows we denote $\mathbb{E}[x_T] := \int_{\mathbb{R}} x\rho_T(x)\mathrm{d}x$ and I some interval for x.

By assuming ρ^* and v polynomial, the adjoint equation becomes:

$$\dot{q}x^2 + \dot{r}x + \dot{s} + (ax + b(Ax+B))(2qx + r) + \sigma^2(Ax+B)^2 q = 0$$

In turn, it implies

$$\dot{q} + q(2a + 2bA + A^2\sigma^2) = 0, q(T) = \frac{\gamma}{2}$$

$$\dot{r} + (a + bA)r + 2qBb + 2\sigma^2 ABq = 0,\ \ r(T) = -\frac{1}{2} - \gamma\int_{\mathbb{R}} x\rho_T$$

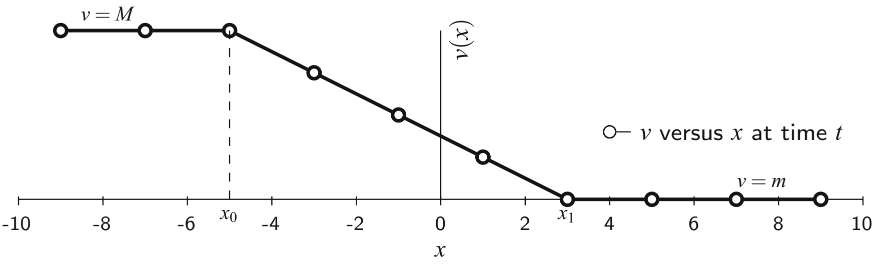


Fig. 1 The control is $v = M$ when $x < x_0$, affine when $x_0 < v < x_1$, and $v = m$ when $x > x_1$ with $x_0 = -\frac{M\sigma^2}{b} + (\frac{1}{2\gamma} + \mathbb{E}[x_T])e^{-a(T-t)}$, $x_1 = -\frac{m\sigma^2}{b} + (\frac{1}{2\gamma} + \mathbb{E}[x_T])e^{-a(T-t)}$

$$\begin{aligned} \dot{s} + rBb + \sigma^2 B^2 q = 0,\ \ s(T) = 0 \\ b\partial_x \rho^* + \sigma^2 v \partial_{xx}\rho^* = 2(b + \sigma^2 A)qx + br + 2\sigma^2 Bq \quad . \end{aligned} \tag{18}$$

Regime 1: $v = Ax + B = m(t)\ \forall x \in I \quad \Rightarrow \quad A = 0, B = m$. This happens only when $b(2qx + r) + 2m\sigma^2 q \geq 0$. Then

$$q = \frac{\gamma}{2} e^{2a(T-t)},\ \ r = -(\frac{1}{2} + \gamma \mathbb{E}[x_T])e^{a(T-t)} - b\gamma e^{a(T-t)} \int_t^T m(\tau) e^{a(T-\tau)} \mathrm{d}\tau.$$

Hence

$$x > -\frac{m\sigma^2}{b} - \frac{r}{2q} = -\frac{m\sigma^2}{b} + (\frac{1}{2\gamma} + \mathbb{E}[x_T])e^{-a(T-t)} + b\int_t^T m(\tau)e^{-a(\tau-t)}\mathrm{d}\tau$$

is required for this regime.

Regime 2: $b + \sigma^2 A = br + 2\sigma^2 Bq = 0$ and $m < v = Ax + B < M$. Then after some algebra

$$-\frac{M\sigma^2}{b} + (\frac{1}{2\gamma} + \mathbb{E}[x_T])e^{-a(T-t)} \ < x < -\frac{m\sigma^2}{b} + (\frac{1}{2\gamma} + \mathbb{E}[x_T])e^{-a(T-t)}.$$

Regime 3: Similar to Regime 1, $v = M(t)$ requires

$$x > -\frac{M\sigma^2}{b} + (\frac{1}{2\gamma} + \mathbb{E}[x_T])e^{-a(T-t)} + b\int_t^T M(\tau)e^{-a(\tau-t)}\mathrm{d}\tau. \tag{19}$$

Remark 2 The advantage here compared with [2] is that we do not need to guess the shape of the control nor of the adjoint state, once it is assumed polynomial. The analysis also handles constraints.

4 Find the Best Production Strategy for an Exhaustible Resource

Following [17], consider a continuum of producers exploiting an oil field. Each producer's goal is to maximize his profit, knowing the price of oil; however, this price is influenced by the quantity of oil available on the market, which is the sum of all that the producers have decided to extract at a given time. Hence, while each producer does not affect the price of oil, in the end the global problem must take into account the market price as a function of oil availability. For a deeper understanding of the relation between the individual and global games, readers are referred to [12].

4.1 Notations

Let X_0 be the initial oil reserve and X_t be the quantity of oil left in the field at time t, as seen by a producer. It is modeled by

$$\mathrm{d}X_t = -a_t\mathrm{d}t + \sigma X_t\mathrm{d}W_t, \quad X_0 \text{ given by its PDF,} \tag{20}$$

where a_t is the extraction rate, i.e. $a_t\mathrm{d}t$ is the quantity extracted by the producer in the time interval $(t, t + \mathrm{d}t)$. Here W is a standard Brownian motion reflecting the incertitude of the producer about the remaining reserve; $\sigma > 0$ is a volatility parameter, assumed constant.

We assume that $a_t := a(X_t, t)$ is a deterministic feedback function of t and X_t.

The cost of oil extraction is assumed to be $C(a) := \alpha a + \beta a^2$, for some positive constants α and β. The price of oil as a function of extraction is assumed to be $p_t := \kappa\mathrm{e}^{-bt}(\mathbb{E}(a_t))^{-c}$, with positive κ, b and c. Note that by construction X_t takes only positive values and ought to be bounded by, say L, the maximum estimate of the reservoir content. However, nothing in the model enforces these constraints.

Each producer optimizes his integrated profit up to time T, discounted by the interest rate r; however he wishes also to drain the oil field, i.e., achieve $X_T = 0$. Thus his goal is to maximize over $a(\cdot, \cdot) \geq 0$ the functional:

$$J(a) := \mathbb{E}\left[\int_0^T (p_t a_t - C(a_t))\mathrm{e}^{-rt}\mathrm{d}t\right] - \gamma\mathbb{E}[|X_T|^\eta], \quad \text{subject to (20);} \tag{21}$$

γ and η are penalization parameters. Replacing p and C by their expressions gives

$$J(a) = \mathbb{E}\left[\int_0^T (\kappa\mathrm{e}^{-bt}(\mathbb{E}[a_t])^{-c}a_t - \alpha a_t - \beta(a_t)^2)\mathrm{e}^{-rt}\mathrm{d}t\right] - \gamma\mathbb{E}[|X_T|^\eta].$$

4.2 Dynamic Programming Solved by a Fixed Point Algorithm

To connect to Sect. 2 let us work with $u = -a$, the reserve depletion rate. For the time being, we shall ignore the constraints $L \geq X_t \geq 0$ and $u \leq 0$; so $\mathcal{V}_d = \mathbb{R}$. Moreover we shall work with $\eta = 2$ and comment on $\eta > 2$ at the end.

Recall that $\rho(\cdot, t)$, the PDF of X_t, is given by the Fokker-Planck equation :

$$\partial_t \rho - \frac{\sigma^2}{2}\partial_{xx}\left(x^2\rho\right) + \partial_x\left(\rho u\right) = 0 \qquad (x,t) \in \mathbb{R}\times]0,T], \tag{22}$$

with initial condition: $\rho_{|t=0} = \rho_0$ given. Now $\overline{u}_t := \int_{\mathbb{R}} u_t \rho_t \mathrm{d}x = \mathbb{E}[-a_t]$ and

$$\tilde{J}(\tau; \rho_\tau, u) := \int_\tau^T \int_{\mathbb{R}} \left(\kappa \mathrm{e}^{-bt}(-\overline{u}_t)^{-c} u_t - \alpha u_t + \beta u_t^2\right)\mathrm{e}^{-rt}\rho_t \mathrm{d}x\mathrm{d}t \tag{23}$$

$$+ \int_{\mathbb{R}} \gamma |x|^\eta \rho_T \mathrm{d}x \;\text{ subject to (22) with } \rho_{|t=\tau} = \rho_\tau. \tag{24}$$

The goal is now to minimize $\tilde{J}$ with respect to u. In this example we have

$$H(x,t;u_t,\rho_t) = (\kappa \mathrm{e}^{-bt}\chi_t^{-c} u(x,t) - \alpha u(x,t) + \beta u(x,t)^2)\mathrm{e}^{-rt},$$

$$\chi_t = -\int_{\mathbb{R}} u_t \rho_t \mathrm{d}x, \;\text{ hence }\; h(x,t,u,\rho) = -u(x,t) \;\text{ and }\; G(x,\chi_T) = G(x) = \gamma|x|^\eta.$$

By Proposition 1 we have $V'_{|T} = \gamma|x|^\eta$ and V' satisfies

$$\partial_t V' + \frac{\sigma^2 x^2}{2}\partial_{xx} V' + u\partial_x V' = -\left[H + \rho\partial_\rho H + (h + \rho\partial_\rho h)\int_{\mathbb{R}} \partial_\chi H \rho \mathrm{d}x\right] \tag{25}$$

$$= -\left(\kappa(1-c)\mathrm{e}^{-bt}(-\overline{u})^{-c} u - \alpha u + \beta u^2\right)\mathrm{e}^{-rt.} \tag{26}$$

because $\partial_\rho H = \partial_\rho h = 0$ and $\partial_\chi H = c\kappa \mathrm{e}^{-bt}\chi^{-c-1} u$. Moreover, by (7),

$$-\partial_x V' = \left(\kappa(1-c)\mathrm{e}^{-bt}(-\overline{u})^{-c} - \alpha + 2\beta u\right)\mathrm{e}^{-rt}, \tag{27}$$

$$\text{giving:} \qquad u(x,t) = \frac{1}{2\beta}\left[\alpha - \mathrm{e}^{rt}\partial_x V' - \kappa(1-c)\mathrm{e}^{-bt}(-\overline{u})^{-c}\right]. \tag{28}$$

Now, using (27) to eliminate $\partial_x V'$ in (26) and the expression above for u leads to

$$\partial_t V' + \frac{\sigma^2 x^2}{2}\partial_{xx} V' = \frac{\mathrm{e}^{-rt}}{4\beta}\left(\alpha - \mathrm{e}^{rt}\partial_x V' - \kappa(1-c)\mathrm{e}^{-bt}(-\overline{u})^{-c}\right)^2. \tag{29}$$

Note that this equation for V' depends only on $\overline{u}$ and not on u. Nevertheless (28)–(29) is a rather complex partial-integro-differential system; we can now sketch a numerical method to solve it:

```
Algorithm 1.
Choose $\omega \in (0, 1)$. initialize $u_0$, set $i = 0$.
  1. Compute $\rho_i$ by solving (22) ;
  2. Compute $\overline{u}_i = \int_{\mathbb{R}} u_i \rho_i$ ;
  3. Compute $V_i'$ by (29);
  4. Compute $\tilde{u}_{i+1}$ by (28) and set $u_{i+1} = u_i + \omega(\tilde{u}_{i+1} - u_i)$;
If not converged, set $i = i + 1$ and go back to step 1.
```

Although it seems to work numerically in many situations, as we shall see below, nothing is known on the convergence of this fixed-point type algorithm.

Even though the problem is not linear-quadratic, when $\eta = 2$ we can still look for V' solution of (29) in the form $V'(x,t) = P_t x^2 + z_t x + s_t$. where P_t, z_t, s_t are functions of time only. Identification of all terms proportional to x^2 gives,

$$\dot{P}_t + \sigma^2 P_t = \frac{\mathrm{e}^{rt}}{4\beta} P_t^2, \quad P_T = \gamma, \quad \Rightarrow \; P_t = \frac{4\beta\mu\gamma \mathrm{e}^{(T-t)\mu}}{\gamma \mathrm{e}^{(T-t)\mu} - \gamma + 4\beta\mu}.$$

Then, u is found by (28). In particular $\partial_x u = -\frac{1}{8\beta}\partial_{xx} V' = -\frac{1}{4\beta} P_t$. However, the Fokker-Planck equation must be solved numerically to compute $\overline{u}$. This gives us a method to validate the algorithm.

4.3 Numerical Implementation

To implement Algorithms 1, we need to localize the PDE. As $x < 0$ makes no sense for this application, we shall work on $Q_L = [0, L] \times [0, T]$ instead of $\mathbb{R} \times [0, T]$; a stopping time for the event $X_t = 0$ would be better, but too costly. At $x = L$, we set $\rho(L, t) = 0, \forall t$, which makes sense when L is large.

Assigning boundary conditions to (29) is problematic. Our numerical tests show that the computations depend strongly on L when it is not done correctly. When $\eta = 2$, we know that V' and ρ^* have asymptotically the same behavior as $P_t x^2$, giving $\frac{1}{2}\sigma^2 x^2 \partial_x V' = \sigma^2 x^3 P_t = \sigma^2 x V'$, a relation which can be used as a boundary condition in the weak form of the equation (and similarly for ρ^*): find $\rho \in H^1(Q_L)$ with V_T' given and

$$\int_{Q_L} \left[-\nu\partial_t V' + \frac{\sigma^2}{2}\partial_x(\nu x^2)\partial_x V' \right] \mathrm{d}x\mathrm{d}t + \int_0^T \sigma^2 L V'(L,t)\nu(L,t)\mathrm{d}t$$
$$+ \int_{Q_L} \frac{\mathrm{e}^{-rt}}{4\beta}\left(\alpha - \mathrm{e}^{rt}\partial_x V' - \kappa(1-c)\mathrm{e}^{-bt}(-\overline{u})^{-c}\right)^2 \nu\mathrm{d}x\mathrm{d}t = 0, \qquad (30)$$

for all $\nu \in H^1(Q_L)$ with $\nu_T = 0$.

To solve this non-linear PDE we use the fact that it is embedded into an iterative loop in Algorithm 1, and semi-linearise it by evaluating the square term in the last integral as a product of the same, where one factor is evaluated at the previous iteration.

To discretise it we have used a space-time finite element method of degree 1 over triangles covering Q_L. Admittedly it is an unusual method! However, it is somewhat similar to a central difference method and it is feasible because the problem is one dimensional in space and because it allows exact conservativity and exact duality with respect to time and space in the integrations by parts. It handles also automatically the storage of ρ, u, V, $\overline{u}$ at all times and solve the backward (and forward) equation at all time steps by a single linear system. The linear systems are solved with the library MUMPS as implemented in `freefem++` [18].

We used 50 points to discrete $(0, L)$, $L = 10$ and 50 time steps for $[0, T]$, $T = 5$. The following values have been used: $\alpha = 1$, $\beta = 1$, $\gamma = 0.5$, $\kappa = 1$, $b = 0.1$, $r = 0.05$, $\sigma = 0.5$ and $c = 0.5$. The initial condition ρ_0 is a Gaussian curve centered at $x = 5$ with volatility 1. We initialized u by $u_0 = -\alpha/(2\beta)$. A local minimum u_e is known from the Riccati equation; the error $\|u - u_e\|$ is used as a stopping criteria in Algorithm 1. We chose $\omega = 0.5$. Figure 2 shows the optimal control as a function of

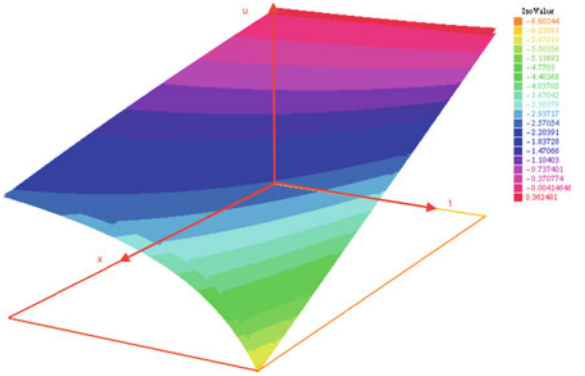


Fig. 2 Optimal $x, t \to u(x, t)$ and the Riccati solution slightly below

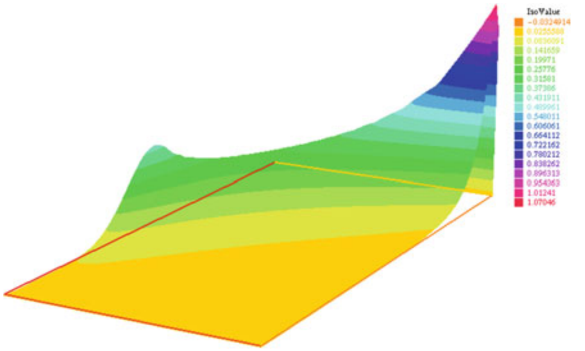


Fig. 3 PDF of resource X_t: $x, t \to \rho(x, t)$

Table 1 Algorithm 1. Convergence history: H^1-error, $\int(\partial_x u - \partial_x u_e)^2 \mathrm{d}x\mathrm{d}t$ versus iteration number k

k	1	2	3	4	5	6	7	8	9	10
Error	1035	661.2	8.605	44.7	3.27	0.755	0.335	0.045	0.015	0.003

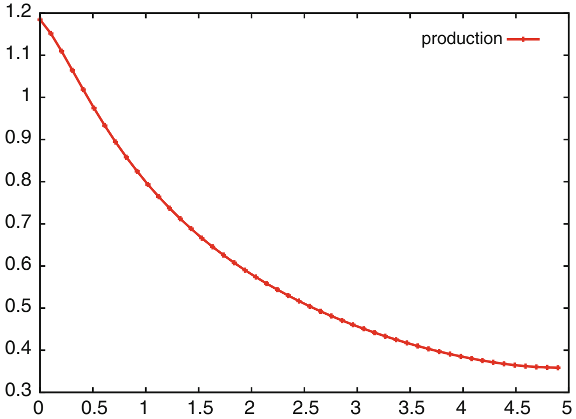


Fig. 4 Total (instantaneous) production versus time

(x, t). For each t the control is linear in x, as predicted by the Riccati equation; the quadratic part of the Riccati solution of Sect. 4.2 is also plotted, and a small difference is seen on the plot (two surfaces close to each other are displayed). Figure 3 shows the evolution in time of the PDF ρ for all $x > 0$ of the resource distribution X_t. At time 0 it is a Gaussian distribution centered at $x = 5$; at time T the distribution is concentrated around $x = 0.5$, so most producers have pumped 90% of the oil available to them. Table 1 gives the convergence history of the algorithm.

All above is obtained with $L = 10$, but there is almost no difference with $L = 40$. Figures 4 and 5 present the evolution of production $(-\bar{u}_t)$ and price $p_t = \kappa \mathrm{e}^{-bt}(-\bar{u}_t)^{-c}$.

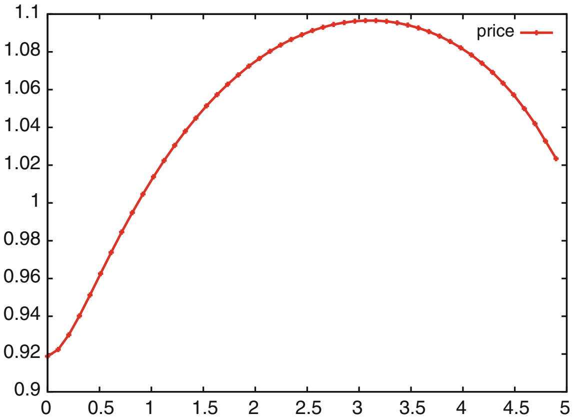
5 Numerical Solution of a Systemic Risk Problem

Consider N banks, each with

$$\mathrm{d}X_{j_t} = -hV'(X_{j_t})\mathrm{d}t + \sigma \mathrm{d}W_t + \theta\Big(\frac{1}{N}\sum_k X_{kt} - X_{j_t}\Big),\ \ j = 1..N$$

with $V = \frac{1}{2}x^2(x^2 - 1)$. There are two limit states: $\lim_{t\to\infty} \in \{X_j X^{\pm} = \pm 1\}$. We assume that X^- is the normal state while X^+ is the dangerous state. The parameter

Fig. 5 Oil Price versus time



h is the height of barrier potential, σ is the randomness intensity and θ is the speed of return to equilibrium.

When $N \to \infty$ the PDF of $N^{-1}\sum_1^N X_j$ tends to μ solution of [13]

$$\partial_t\mu + \partial_x(b\mu) - \frac{\sigma^2}{2}\partial_{xx}\mu = 0,\ b = -hV'(x) + \theta(\mathbb{E}[x] + x\ \mathbb{E}[x] = \int_{\mathbb{R}} x\mu(x)\mathrm{d}x \qquad (31)$$

In [14, 16] it is shown that the rare event probability that the state of the system evolves from X^- to X^+ is proportional to $\exp(-\frac{N}{2\sigma^2}\int_{R\times(0,T)} b^2\rho\mathrm{d}x\mathrm{d}t)$

With $\kappa = h - \theta$ and $\chi = \theta\int_{\mathbb{R}} x\rho\mathrm{d}x$, the problem is restated as

$$\min_u J = \frac{1}{2\sigma^2}\int_{\mathbb{R}\times(0,T)} (hx^3 - \kappa x - \chi + \frac{u}{\rho})^2\rho\ :\ \partial_t\rho - \frac{\sigma^2}{2}\partial_{xx}\rho = -\partial_x u,\ \ \rho_{|0}, \rho_{|T}\ \text{given.}$$

Finding solutions of the Fokker-Planck equation with initial and final conditions is quite difficult (see a Monge-Kantorovich solution in [4]). We have tried use penalisation to impose the final condition with very mediocre results [19]. Now in one dimension we notice that for a given $\tilde{u}$, if we denote $\tilde{\rho}$ the solution of the PDE with the initial condition only, then $\rho = \tilde{\rho} + \frac{t}{T}(\rho_T - \tilde{\rho}_T)$ satisfies the PDE and both conditions at 0 and T provided $u = \tilde{u} - \frac{1}{T}\int^x(\rho_T - \tilde{\rho}_T)\mathrm{d}x + \frac{t\sigma^2}{2T}\partial_x(\rho_T - \tilde{\rho}_T)$.

The problem has been solved by 3 software packages which do not require derivatives: NEWUOA, CMAES and IPOT. For the first 2, the control is in the space of polynomial of degree n or less in $x \times t$, n = 2 and 3. Both packages give the same results, J decreased from $J = 2.058$ to $J = 1.072$. The solution is shown on Fig. 6 (all but bottom right). IPOPT is a differentiable optimization package so at each finite element node there is a degree of freedom for the control. Then J can reach very small values; for instance $J = 0.49$ after 10 iterations; the control is shown on Fig. 6 (bottom right); but as the number of iterations is increased the control becomes more

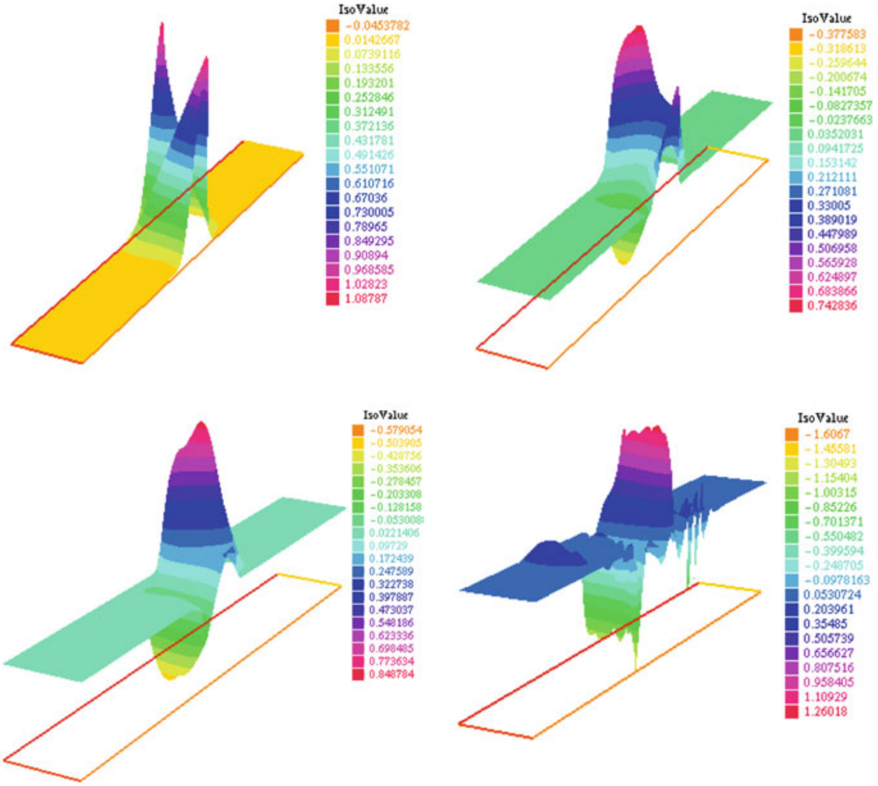


Fig. 6 Top left: $x, t \mapsto \rho(x, t)$; $t = 0$, $x = -L$ is the top corner of the rectangle and $x = L$, $t = T$ is its lowest corner. All 3 packages give almost identical ρ. Top right: the control $b = u/\rho$ when a polynomial approximation is used for ρ of degree 2. Bottom left: same but with polynomials of degree 3. Bottom right: same but with iPOPT

and more oscillatory. Some regularisation is required. In conclusion while one can find a strategy that steers the system from a stable solution to an unstable one, yet it is difficult to give the probability of that event.

References

1. Albrecher, H., Boxma, O.: A ruin model with dependence between claim sizes and claim intervals. Insur. Math. Econ. **35**(2), 245–254 (2004)
2. Andersson, D., Derviche, B.: A maximum principle for SDEs of mean-field type. Dyn. Games Appl. **3**, 537552 (2013)
3. Basel Committee on Banking Supervision: Basel III counterparty credit risk—frequently asked questions. http://www.bis.org/publ/bcbs209.pdf

4. Benamou, J.D., Brenier, Y.: A computational fluid mechanic solution of the Monge-Kantorivich mass transfer problem. Numerische Mathematik **84**(3), 375–393 (2000)
5. Bensoussan, A., Frehse, J.: Control and nash games with mean field effect. Chin. Ann. Math. Ser. B **34B**(2), 161192 (2013)
6. Bensoussan, A., Frehse, J., Yam, S.C.P.: Mean-Field Games and Mean-Field Type Control. Springer Briefs in Mathematics (2014)
7. Bensoussan, A., Frehse, J., Yam, S.C.P.: The master equation in mean-field theory. In: Asymptotic Analysis (To appear)
8. Bensoussan, A., Bertrand, P., Brouste, A.: A generalized linear model approach to seasonal aspects of wind speed modeling. J. Appl. Stat. **41**(8), 1694–1707 (2014)
9. Bijl, H., Lucor, D., Mishra, S., Schwab, C. (eds.): Uncertainty Quantification in Computational Fluid Dynamics. Lecture Notes in Computational Science and Engineering. Springer (2013)
10. Bielecki, T.R., Rutkowski, M.: Credit Risk: Modeling, Valuation and Hedging. Springer, Berlin (2002)
11. Brouste, A.: Estimation of wind turbine energy production. Seminaire FIME, IHP, Feb 2015
12. Carmona, R., Delarue, F., Lachapelle, A.: Control of McKean-Vlasov dynamics versus mean field games. In: Stochastic Analysis and Applications (2014) (To appear)
13. Dawson, D.A.: Critical dynamics and fluctuations for a mean-field model of cooperative behavior. J. Stat. Phys. **31**, 29–85 (1983)
14. Dawson, D.A., Gartner, J.: Large deviations from the McKean-Vlasov limit for weakly interacting diffusions. Stochastics **20**, 247–308 (1987)
15. Fleming, W.H., Soner, H.M.: Controlled Markov Process and Viscosity Solutions, 2nd edn. Springer (2006)
16. Garnier, J., Papanicolaou, G., Yang, Tzu-Wei: Large deviations for a mean field model of systemic risk. SIAM J. Finan. Math. **4**(1), 151184 (2013)
17. Gueant, O., Lasry, M., Lions, P.L.: Mean Field Games and Applications. Paris-Princeton Lectures on Mathematical Finance. Lecture Notes in Mathematics. Springer (2011)
18. Hecht, F.: New development in freefem++. J. Numer. Math. **20**(3–4), 251–265 (2012)
19. Laurière, M., Pironneau, O.: Dynamic programming for mean-field type control. C.R. Acad. Sci. Ser. **I**, 1–6 (2014)
20. Laurière, M., Pironneau, O.: Dynamic programming for mean-field type control. JOTA (to appear)
21. Lasry, J.M., Lions, P.L.: Mean-field games. Jpn. J. Math. **2**, 229–260 (2007)
22. Le Bris, C., Lions P.L.: Existence and uniqueness of solutions to Fokker-Planck type equations with irregular coefficients. Comm. Partial Differ. Equ. **33**, 1272–1317 (2008)
23. Lions, P.L.: Mean-field games. Cours au Collège de France (2007–2008). http://www.college-de-france.fr/site/pierre-louis-lions/course-2007-2008_1.htm
24. Neufeld, A., Nutz, M.: Nonlinear Lévy processes and their characteristics. arXiv:1401.7253v1
25. Øksendal, B., Sulem, A.: Applied Stochastic Control of Jump Diffusions. Springer (2005)
26. Shen, M., Turinici, G.: Liquidity generated by heterogeneous beliefs and costly estimations. Netw. Heterogen. Media AIMS-Am. Inst. Math. Sci. **7**(2), 349–361 (2012)
27. Touzi, N.: Optimal Stochastic Control, Stochastic Target Problems and Backard SDE. Field Institute Monographs 29. Springer (2013)
28. Yong, J., Zhou, X.Y.: Stochastic Control. Applications of Mathematics Series, vol 43. Springer (1999)

Part II
Surrogate-Based Optimization in Aerodynamic Design

A Review of Surrogate Modeling Techniques for Aerodynamic Analysis and Optimization: Current Limitations and Future Challenges in Industry

Raul Yondo, Kamil Bobrowski, Esther Andrés and Eusebio Valero

Abstract Recent progresses in aircraft aerodynamics have witnessed the introduction of surrogate-based approaches in design analysis and optimization as an alternative to address the challenges posed by the complex objective functions, constraints, the high-dimensionality of the aerodynamic design space, the computational burden, etc. The present review revisits the most popular sampling strategies used in conducting physical and simulation-based experiments in aircraft aerodynamics. Moreover, a comprehensive and state-of-the art survey on numerous surrogate modeling techniques for aerodynamic analysis and surrogate-based optimization (SBO) is presented, with an emphasis on models selection and validation, sensitivity analysis, infill criterion, constraints handling, etc. Closing remarks foster on the drawbacks and challenges linked with SBO aircraft aerodynamic industrial applications, despite its increased interest among the academic community.

1 Introduction

High dimensional aerodynamic design problems require time consuming and computationally expensive simulations or physical experiments to evaluate the complex objectives functions and constraints for analysis and optimization. As an example

R. Yondo (✉) · K. Bobrowski · E. Valero
Universidad Politécnica de Madrid (UPM), Plaza Cardenal Cisneros 3,
28040 Madrid, Spain
e-mail: yondo.raul@upm.es

K. Bobrowski
e-mail: kamil.bobrowski@upm.es

E. Valero
e-mail: eusebio.valero@upm.es

E. Andrés
Instituto Nacional Técnica Aeroespacial (INTA/ISDEFE), INTA Torrejón de Ardoz,
28850 Madrid, Spain
e-mail: eandres@isdefe.es

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_2

while designing a new aircraft, a CFD (Computational Fluid Dynamics) campaign provides the aerodynamicists with flow solutions from only few selected flight conditions because it will be time-consuming and costly to evaluate all of them. To circumvent such burdens and enable a wider exploration of the design space, researchers introduced cheap to evaluate surrogate models to accurately and speedily predict the values of the model objectives and constraints at new design points, without a need to necessarily repeat the original expensive physical or simulation experiment.

Optimization through the use of surrogates models is called Surrogate-Based Optimization (SBO). Gaining a good insight into SBO requires a knowledge in Design of Experiments (DoE-spatial distribution of the samples—training set), simulation-based (high-fidelity models, e.g. Reynold-Averaged Navier Stokes + fine mesh; low-fidelity, e.g. Euler simulations) or physical models (e.g. Wind Tunnel Testing) and Surrogate Models. Moreover, the SBO frameworks are heavily dependent on the design objectives, the constraints. They should quantify the uncertainty (sensitivity analysis) within the surrogate models, deal with ‘noisy’ data and pare down the problems posed by the high-dimensionality of the data (Dimensionality Reduction). A vast number of papers review and group the various releases on Surrogate models and SBO. Popular reviews are the ones from Queipo et al. [1], Forrester et al. [2], Ahmed and Qin 2009 [3] or more recently Viana et al. [4]. However, it is not clear from the readings if the approaches and frameworks presented in the above-mentioned papers are suitable for aerodynamic analysis and optimization, and could fit the needs of the aircraft industry.

Motivated by the lack of a review considering industrial views proper to aircraft aerodynamics design analysis and optimization, the present paper aims at revisiting the Classical DoE (CDoE), the Modern DoE (MDoE, a.k.a Design and Analysis of Computer Experiments (DACE)) in Sect. 2 and provides a comprehensive summary on existing surrogate models categories, including Data Fit Models (DFM), Reduced-Order Models (ROM) and Multi-Fidelity Models (MFM) in Sect. 3. Furthermore, the issues of models selection and validation are investigated. The outline of the subsequent part of the paper is as follows: Sect. 4 introduces the ‘conventional’ SBO framework, with an emphasis on sensitivity analysis, constraints handling and sequential/adaptative sampling (infill criterion approaches) to guide in new samples selection for surrogate models update and/or optimization. Section 5 discusses the limitations, known current and future challenges in applying SBO in aircraft aerodynamics, from an industrial perspective. The last section concludes the review.

2 Design of Experiments (DoE)

Design of Experiments (DoE) was originally introduced by Fisher and al. in the beginning of the 20th century to plan and conduct experiments with the aim of understanding the probabilistic behavior of agricultural crops systems [5]. In the early beginnings, experiments were only concerned with physical experiments, for which the basic principles of DoE (randomization, replication, and blocking) established by Fisher are applicable. Physical experiments means an experimenter ‘physically’ carries out the experiments (example of Wind Tunnel Testing—WTT), that are more often corrupted by random errors (‘noise’) and consequently makes the data modeling complex. With the advent of numerical methods and computing facilities, it became possible to design experiments through computer-based simulations. In contrast to physical experiments where the process looked into is stochastic, simulation-based experiments are deterministic, with no need of replication [6]. Sampling the design space, i.e. selecting a good set of data points (also called observations or training points) became a key issue in computer-based simulations, with the prevailing goal of maximizing the amount of information gained from a limited number of samples.

Sampling techniques can be grouped into two main categories, namely Classical DoE (CDoE) [5] and Modern DoE (MDoE or Design and Analysis of Computer Experiments-DACE in some literature) [7]. Additionally, training point selection approaches can be classified into domain-based (a.k.a non-adaptative, a priori, one-shot, off-line or single-stage) and response-based (a.k.a model-based, adaptative, a posteriori, sequential or online) approaches. In domain-based approaches, training points are chosen based on the information available from the design space (e.g. distance between two training points); whereas in response-based approaches the training points are chosen based on the information provided by the surrogate model (e.g. mean squared error approach) to enhance their efficiency. Domain-based approaches are based on space-filling techniques [6], that try to spread the sample points uniformly throughout the design space (a possible guarantee for an eventually acceptable a priori global approximation of the surrogate model). Generally, it is not possible in aircraft aerodynamics to a priori select the number of samples to establish a given accuracy, mainly because of the non-linearities of the aerodynamic functions. Nevertheless, space-filling approaches provide the opportunity to monitor the refinement of the surrogate model and to choose to stop or prolong the sampling process (update of the design space). A thorough introduction to response-based approaches is given in Rai et al. [8] or Forrester et al. [9] and will be presented in Sect. 4.

2.1 Classical DoE

Classical sampling techniques have known an extensive use in aerospace engineering for various design improvement applications. Originally developed for planning physical experiments, they assume the existence of a random error term in the measured response and a non-repeatability of the latter, even with the use of the same sample during two different measurements; hence the use of replicated sampling. Classical sampling techniques presuppose a uniform distribution of the training points with the aim to have a fixed number of samples in the design space, so as to minimize the influence of the random error term in posterior simulations. Therefore, a significant number of samples tend to locate in the neighborhood or on the boundaries of the design space, with very few samples in the interior of the design space. A comprehensive review on the causes of such samples spread is given by Myers and Montgomery [10]. Some of the CDoEs can be used to plan simulation-based experiments, sometimes arising in good enough approximations of the objective functions [11]. Commonly used classical sampling techniques in Aerospace engineering are:

- Full and Fractional-Factorial designs (a.k.a grid designs) [12],
- Central Composite designs [13],
- Box-Behnken designs [14],
- Plackett-Burman designs [15],
- Optimal designs [16],
- Orthogonal arrays designs often referred as Taguchi designs [17].

2.2 Modern DoE

MDoE sampling techniques have known a comprehensive development in parallel to the evolution of computer science and ease access to computing facilities [6]. Contrary to physical experiments, simulation-based experiments are deterministic, i.e. not only samples with the same input setting produce the same outputs but also no measurements inaccuracies inherent to randomness occur. MDoE have become a substitute to CDoE when the latter are infeasible or too costy. MDoE sampling techniques tend to locate the training data points in the interior of the design space, with the aims to find an approximate model and to minimize the bias errors—local gaps between the true response trend and its estimated trend (surrogate) [9]. Therefore, MDoE sampling techniques require different approaches than CDoE techniques.

Till date, there is no rule of thumb regarding the choice of a specific MDoE sampling strategy for aircraft aerodynamics design and analysis purposes, a good accuracy of the generated approximations being subject not only to the sampling points and the choice of the surrogate, but also the nature of the underlying problems [9]. Throughout the decades, statisticians have developed numerous metrics to ensure a reasonable coverage of the design space [6] (Euclidean distanced-based, Statistical-based, Projection-based, etc.). Besides the previously-mentioned sampling metrics, means that help to a posteriori improve the performance of these a priori sampling strategies (constructed from narrow set of training points) may encompass the use of some optimality criteria or other metrics/criteria that will be discussed in subsequent sections. Hybrid DoEs arising from combinations of the above 'traditional' sampling strategies (CDoE and MDoE) are sometimes possible and have seen an increased interest during the past decade. As examples one may quote Orthogonal Latin Hypercube [18] or Orthogonal Maximin Latin Hypercube [19], etc. Table 1 introduces common a priori MDoE used in aircraft aerodynamics design and analysis. The attached references mainly coin the theories behind each of the MDoE.

Table 1 MDoE used in aircraft aerodynamic design optimization

MDoE	Examples
Random samplings	Simple Random samplings [20]
	Monte Carlo samplings [21]
Quasi-random samplings	Quasi-Monte Carlo designs [22]
	Sobol designs [23]
	Hammersley or Halton samplings [24]
Projection-based designs	MDoE + OAs [17]
	MDoE + Nearly OAs [18]
Euclidean distance-based designs	Maximin [19]
	Minimax [25]
	Voronoi tesselations [26]
	Uniform samplings [27]
Other MDoE	Symmetric samplings [28]
	Latin hypercube samplings [29]
	Anisotropic designs [30]
	Hybrid designs

3 Surrogate Models

Numerous aircraft aerodynamic design analysis and optimization require running computation intensive and complex simulation codes, such as CFD solvers. Despite the availability of advanced computing capabilities, the excessive computational cost makes it absurd to exclusively rely on expensive simulations (also known as high-fidelity simulations) for the purpose of optimization or design space exploration. An attempt to circumvent the computational budget restrictions was introduced by Box and Draper [31] through the use of surrogate models, known as metamodels, emulators, proxy models or response surface models. Surrogate models are 'cheap-to-evaluate' mathematical approximations that mimic the deterministic computationally expensive response or behavior of an original system over the complete or part of the design space [6]; once constructed, a surrogate is used in lieu of the expensive full-order analysis in order to predict the values of the objective functions at locations that do not belong to the set of points used during the fitting process. Most often aerodynamic functions tend to be highly nonlinear and highly dimensional ('curse of dimensionality' [32]), which required to introduce strategies to reduce the required number of evaluations of the modeled functions.

Deciding on a preference for a surrogate model remains a quite cumbersome task. Both the selection and accuracy of such approximations depend not only on the nature of the underlying problem (e.g. of aerodynamic optimization) but also on the industrial common practices and designer experience (e.g. of the a priori knowledge of the related physics and of the features provided by certain surrogate models) [9]. Ostensibly, to attain an acceptable preliminary approximation (that may not be necessary for a local modeling), it is recommended to make use of space-filling samplings with a minimal number of calls of the modeled functions, while retaining most of underlying features. This is generally not trivial in aerodynamic design, since there is few or no a-priori knowledge of the trends of the modeled functions in advance. Furthermore, the use of surrogate models depends on an interest either in improving the global accuracy over the entire domain (global modeling or design space exploration) or in an approximation of the optimum to locally improve the current design (local modeling or design space exploitation). An extensive review of surrogate modeling use in mechanical and aerospace systems is found in Viana et al. [4]. A wide variety of surrogate models have been introduced into aircraft aerodynamic design (flow control, estimation of aerodynamic coefficients, variable-fidelity data calibration, aerodynamic shape optimization, etc.). They can roughly be broken down into three categories:

1. **Data fit surrogates models**—are regression-based or interpolation-based approximations generated from data arising from few calls of the modeled functions. They are ranked as non-physics-based approximations (no attempt is made to ensure that the surrogate model the flow physics). Gaussian Process Regression (a.k.a Kriging), Radial Basis Functions, Artificial Neural Networks, Support Vector Machines, Polynomial Regressions or chaos, Splines, are some of the common data fit metamodels [3, 6, 33].
2. **Reduced-order models (ROMs)**—ROMs bring an added value to data fits models as they offer the possibility to model high dimensional functions (e.g. local flow variables predictions in aerodynamics). These models can be used for aerodynamic flow field reconstruction, incomplete or 'gappy' data set reconstruction, aerodynamic inverse design, features extraction, flow control, etc. Moreover, they could provide the designer with a deeper understanding of the flow physics. Roughly speaking, two types of ROMs have been introduced into aircraft aerodynamics so far:

    a. **ROMs for linear aerodynamic systems**—Linear methods (e.g. Low-rank Gramian approximants or Moment matching methods) [34–36], Reduced-Order Bases (ROB)-based ROMs [34, 36, 37], Proper Orthogonal Decomposition (POD)-based ROMs + Interpolation methods (e.g. POD+RBF, POD+Kriging, etc.) [38, 39], etc.

    b. **ROMs for nonlinear aerodynamic systems**—Linear and weakly-linear methods (e.g. Volterra series or POD-based ROMs) [36, 40], Hyper-reduction methods (e.g. trajectory piecewise linear approximation (TPWL))[41, 42], Manifold learning-based ROMs + Interpolation [43, 44] (e.g. ISOMAP), etc.

3. **Multi-Fidelity Models (MFM)**—Originally developed for gradient-based optimization, MFM have their roots in attempts to solve, for high-speed and low-cost, high-fidelity optimization expensive problems by combining data from low- and high-fidelities [9, 45–47].

4. **Hybrid surrogate models**—The combination of two or more of the various surrogate models introduced above, with or without derivative-based or metaheuristic algorithms [48, 49], has led to new research orientations in developing emulators. Examples are multiple/ensemble of surrogates [50, 51] or gradient/hessian enhanced surrogate models [9, 52].

Error estimation methods are used in assessing and improving the local or global accuracy of the surrogate models. Popular error metrics found in literature and suitable for pure exploration include the Hold-Out method, the Split Sample method, Cross-validation (Random sub-sampling, Leave-One-Out-(LOO), k-fold), Bootstrapping, Schwarzs Bayesian information criterion, Akaikes information criterion, etc. [6, 9]. These figures of merit present advantages and disadvantages, depending on the type of surrogate model and on an interest in global or local accuracy. They may give erronous estimations and therefore could appear not to be suitable (e.g. cross-validation applied to Kriging [33]). Few metrics other than the ones abovelisted have been proposed to overcome this burden: the Mean Square Error (MSE), the Root MSE, the Adjusted MSE, the Weighted Root MSE, the Integrated MSE, the Mean Absolute Error (MAE), the Prediction Error Sum of Squares (PRESS) [9, 53].

4 Surrogate-Based Optimization (SBO)

Till date, numerous industrial aerodynamic engineers still rely on manual optimization to attain better designs. With the development of computational resources added to the improvements achieved in numerical and stochastic methods, simulation-based optimizations (gradient-based, gradient-free, etc.) have gained an increasing interest among the scientific and engineering communities. Taking advantage of the numerous features display by surrogate models and the complexities of aircraft aerodynamic design problems (noise and inaccuracies in aero data, curse of dimensionality, competing objective functions and constraints, non-linearities, computationally expensive simulations, etc.), a novel surrogate-based optimization framework (SBO) [1, 45] have gained popularity in the processes of rapidly and efficiently searching for optima (local or global) in the range of all possible solutions.

SBO is an iterative optimization process within which the modeled functions are evaluated at new sample locations. The choice of these locations is guided by the so-called infill-criteria that lead to the enhancement of the surrogate model in the regions of optimal solutions. One of the most known SBO approach is Efficient Global Optimization (EGO) [54]. Early SBO algorithms utilized only high-fidelity models. Keeping in mind the high computational cost linked with the latter, the trends in the development of new SBO algorithms have moved towards the use of few number of high-fidelity evaluations combine with more low-fidelity ones. The process is continued until some stopping criterion is achieved or the allocated computational budget totally spent. Numerous infill criterion approaches exist to guide the experimenter during the optimization [9, 55–57]:

- Two-stages approaches: Minimizing the Interpolation Surface or the Statistical Lower Bound, Maximizing the Probability of Improvement (PI) or the (Constrained) Expected Improvement (EI), Trust Region Model Management (TRMM), Surrogate Management Framework (SMF),
- One-stage approaches: Goal seeking or optimization.

Moreover, various optimization algorithms have been employed for finding optimal space-filling samplings, including evolutionary, local search or simulated annealing algorithms [6]. Recent approaches proposed in sequential sampling can be read from Pan et al. [2, 58].

To improve the approximations based on surrogate models, it has been a common place to conduct a sensitivity analysis. Such analysis (that could be global or local) allow to identify the correlated design variables, the insensitive design variables, the design variables that contribute most in altering the responses (screening), model tuning, etc. Moreover, sensitivity analysis may be used in model adjustment and validation to reduce uncertainty and identify the optimal regions within the design space, with the aim of evaluating the robustness of the designs [59]. A broad range of methods tackling sensitivity analysis exist, including scatter plots, correlations coefficients, qualitative screening methods, quantitative approaches based on variance decomposition (Fourier Amplitude Sensitivity Test (FAST), Sobol' methods and regression analysis that are restricted to local sensitivities or to specific surrogate model behaviors [59].

Most often aerodynamic objective functions are subject to constraints. If using SBO for speedy optimization purpose, both the objective functions and the constraints are approximated with cheap surrogate models. A wide spectrum of SBO methodologies and constraints handling in the context of SBO is reviewed in Queipo et al. [1].

5 Aircraft Aerodynamic Industrial Applications: Use, Limitations and Future Challenges

The present chapter introduces some of the known current and future challenges in applying aircraft aerodynamic design SBO, from an industrial perspective. Those challenges arose from drawbacks identified from past experiences in an aircraft industry-leading manufacturer [60–64].[1]

1. **Changing and complex requirements**—Optimization procedures in aircraft aerodynamics are very challenging tasks, that differs from their counterparts considered by academics or researchers. Constant test cases are not ubiquitous, with

[1] The content of the present section does not necessarily reflect the position or policies of any of the industrialist firms here-mentioned. Therefore no official endorsement should be inferred.

objectives, constraints, geometry parametrization or IT environments changing frequently. Aeronautical industrialists and academics have a different mindset. While the first think of robust methods applied to complex cases at reasonable time frames, the latter are in general interested in generating new methods with a high innovation rate and strive to continuously raise their recognition.

2. **Integrated designs and accuracy requirements**—High speed civil transport aircrafts in classical configuration are already well optimized in present-days, which makes further progress arduous. Therefore, advanced optimization procedures that involve the complete aircraft should be introduced, in lieu of the actual modus operandi where parts of the aircraft are optimized separately (example of a clean wing optimized first and the engine integrated afterwards). Moreover, further improvements require flow around the aircraft to be calculated with high accuracy, with a drag error estimated below 1%. This can be achieved using high-fidelity data, with sufficiently accurate calibrations of low-fidelity information.
3. **Geometry parametrization**–The choice of the shape parametrization method is of significant importance in any aircraft aerodynamic design problems because it largely affects the efficiency of the underlying shape optimization. A successful parametrization scheme has to offer the capability to cover a large portion of the design space with a limited set of design variables, a large number of design variables generally arising in slow convergence rates. Based on experienced gained, the parametrization strategies to be developed should:

   - be easily implemented in any aerodynamic design problem and be able to tackle the issues posed by the 'curse of dimensionality' (reduced number of design variables) for a complete aircraft configuration, that is to be considered in future optimization processes,
   - offer the possibility to include global parameters (e.g. angle of attack, wing sweep, etc.) as design variables, the advantages such as continuous gradients and the advantages to attain any geometry (be independent of the initial geometry),
   - assess the deformations of aerodynamic surfaces, including intersections, excrescences or gaps,
   - be able to handle geometric constraints methodically.

   Moreover, the parametrization techniques should not limit the use of major numerical optimization schemes nor violate the assumption that the aerodynamic geometry surface should be continuous.
4. **Computer experiments and Surrogate Models**—Much work remain to be achieved in the construction of efficient algorithms tailored to aerodynamic design and to define strategies in the selection of suitable sampling techniques. Aircraft industrial aerodynamic engineers interest lies on space-filling and adaptative designs. There is a desire to improve the related algorithms for effective models refinement, with a strong interest in improving the sequential designs starting from best practices in computer experiments.

Variable-Fidelity Models (VFM) have been applied in some test cases. Aerodynamic engineers seek to bring higher-fidelity analyzes earlier into the design process. Reducing the number of 'high-fidelity' CFD simulations by 70–80% compared to low-fidelity simulations (Euler, empirical methods, etc.) remain an essential goal, while maintaining high accuracy. Therefore, there is an industrial need to develop algorithms that can iteratively calibrates lower-fidelity information to the high-fidelity ones, in order to reach an optimum of the high-fidelity aerodynamic design problem.

POD and its extensions have shown to be suitable in full aircraft configuration cases for flow reconstruction, model reduction, data reconstruction and storage. New trends concerned with POD involve:

- The development of advanced VFM and POD-based methodologies and tools,
- The development of a data model adaptation scheme, targeted towards an overall data changes (targeted data adjustment). For example the use of POD in the prediction of the aerodynamic flow field on a time-dependent deformed shape (wing aeroelasticity, etc.),
- Investigating the use of POD for interpolation between configurations in more complex conditions (high Mach numbers or angles of attack cases, presence of non-linearities, etc.).

Data fit models are extensively used in the industry, especially for modeling problems with low dimensional outputs (e.g. forces measurements during wind tunnel testing). They are applied as direct surrogates of aerodynamic coefficients and serve to estimate POD modal coefficients at different flow conditions other than those associated with the original snapshots.

Another important requirement from an industrial point of view is the ability to reuse constructed surrogate models for posterior tasks, such as optimization problems with different objectives and constraints.

5. **Computational efficiency (costs)**—Performance of the optimization process is the key to its success in industrial application. Important factors to assess the performance are computational resources used (memory, bandwidth, etc.) and CPU time to deliver the results. A selective use of higher-fidelity simulations, in conjunction with surrogate modeling techniques, will allow to attain an acceptable level of accuracy and help control the computational costs. From an industrial point of view, time to deliver the results is a priority, and it is more favorable to utilize, whenever possible, significantly larger number of CPUs to complete optimization as soon as possible, rather than utilizing the resources more efficiently but with longer delivery time (progress with the design loop requires delivering optimization results no longer than a week after the task was formulated). Due to the large amount of data involved, the disk space limitations should be kept in mind. For example, it is sufficient to store flow variables values only on grid surfaces nodes, this approach requiring much less memory than handling a whole volume solution. However, the question on how to spend the allocated computational budget, in more low- and less high-fidelity simulations or just in high-fidelity ones, remain an open issue.

6. **Grid deformations for small and large displacements—Adaptative mesh variations**—Despite the knowledge gained and numerous meshing roundtables, the aircraft aerodynamic community is still demanding for new approaches and improvements related to the problem of mesh deformation algorithms, especially for large deformations. Mesh deformation algorithms are known to be expensive and at the industrial level, there is a need of more and more efficient but computationally cheap tools capable to cope with 3D grids of complete aircraft configurations with thick/fine boundary layers, nacelles, high-lift devices, etc. Designing algorithms to address the cost reduction have therefore be one of the main visions for the next years. In response to these large mesh deformations, 'remeshing' the entire domain has been the solution proposed by some engineers and scientists. Howbeit, regenerating the mesh is extremely time consuming and may require manual adjustments for complex geometries. Therefore, it is desirable in aircraft aerodynamic optimization to develop robust and efficient mesh deformation methods for large displacements that not only lead to significant computational time savings (by automatizing the process) but also preserve a consistent mesh quality and could be applicable to a variety of mesh types. Furthermore, with the increasing interest in adjoint-based optimization, a particular attention should be put in the grid quality aspect because a poor grid will definitely have considerable effects on the gradients.

   Concurrently, one of the industrial unresolved issues with regards to small deformations has been the definition of the validity of POD over a certain range of deformations; a concern with the use of reduced-order models being to develop some extensions of existing ones that could reliably handle the small deformations and their effects on the flow features. Adaptive Mesh Refinement (AMR) remains a concern when trying to improve the resolution of the flow features at reasonable computational cost.
7. **Interactive process with inputs from designers**—'Manual' optimization till date remains the major strategy applied in aerodynamic shape optimization at industrial level. It provided the designer with a good understanding of the underlying physics and an awareness on how the aerodynamics is affected by the changes in geometry. The major disadvantages with such an approach are that it is time-consuming, it limits an efficient exploration of the design space and it largely depends on the designer expertise. Nevertheless, despite these drawbacks, many industrial aerodynamicists wish to have a control on the optimization process, that could be achieved through a 'decision support scheme' [46], making the designer himself a stopping criteria.

   Such a decision support scheme could be efficient in the context of SBO, within which few designers have a good expertise on when to stop the optimization process, without necessarily waiting for the allocated computational budget to run out.

6 Conclusion

The present paper provides a comprehensive and present-day overview of Design of Experiments, Surrogate Models and their applications in support of aircraft aerodynamic design analysis and optimization. Topics such as models selection and validation, constraints handling, sensitivity analysis are outlined. At last, the review addresses the limitations and challenges in use of SBO in the aircraft industry. It is hope the paper will serve as a good starting point for novices and will help aerodynamicists to decide on new research orientations in the domain of SBO. An extensive coverage of the approaches encountered on this short review are to be published as a journal article.

Acknowledgements The first two authors acknowledge the funding support from the European Commission' Seventh Framework Programme (FP7) through the AIRUP (Airbus-UPM European Industrial Doctorate in mathematical methods applied to aircraft design) project. The authors wish to express their appreciations to Dr. Stephan Kappel and Expert Holger Barnewitz from the AIRBUS Flight Physics Department in Bremen, Germany.

References

1. Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanythan, R., Tucker, P.: Surrogate-based analysis and optimization. Prog. Aerosp. Sci. **41**(1), 1–28 (2005)
2. Forrester, A., Keane, A.: Recent advances in surrogate-based analysis optimization. Prog. Aerosp. Sci. **45**(1–3), 50–79 (2009)
3. Ahmed, M., Qin, N.: Surrogate-based aerodynamic design optimization: use of surrogates in aerodynamic design optimization. In: 13th International Conference on Aerospace Sciences and Aviation Technology, May 2009
4. Viana, F., Simpson, T., Balabanov, V., Toropov, V.: Metamodeling in multidisciplinary design optimization: how far have we really come? AIAA J. **52**(4), 670–690 (2014)
5. Montgomery, D.C.: Design and Analysis of Experiments, 8th Edition International Student Version. Wiley (2012)
6. Fang, K.-T., Li, R., Sudjianto, A.: Design and Modeling for Computer Experiments. Chapman & Hall/CRC, Taylor & Francis Group, London (2006)
7. Kleijnen, J.P.: Design and Analysis of Simulation Experiments. LLC, Tilburg University, Tilburg, The Netherlands, Springer Science+Business Media (2008)
8. Rai, R.: Qualitative and Quantitative Sequential Sampling. Ph.D. thesis. University of Texas, Texas (2006)
9. Forrester, A., Sóbester, A., Keane, A.: Engineering Design via Surrogate Modeling. Wiley, University of Southampton (2008)
10. Myers, R., Montgomery, D., Anderson-Cook, C.: Response Surface Methodology: Process and Product Optimization Using Design of Experiments. Wiley, University of Southampton (2009)
11. Sacks, J., Welch, W., Schiller, S., Mitchell, T.: Design of computer experiments. Technometrics **31**(4), 41–47 (1989)
12. Mason, R., Gunst, R., and Hess, J.: Statistical Design and Analysis of Experiments. Wiley (2003)
13. Draper, R.N.: Introduction to Box and Wilson (1951) on the experimental attainment of optimum conditions. In: Breakthroughs in Statistics. Springer Series in Statistics, Methodology and Distribution, pp. 267–269 (1992)

14. Box, B., Behnken, D.: Some new three level designs for the study of quantitative variables. Technometrics **2**, 455–475 (1960)
15. Plackett, R., Burman, J.: The design of optimum multifactorial experiments. Biometrika **33**(4), 305–325 (1946)
16. Steinberg, D., Hunter, G.: Experimental design: review and comment. Technometrics **26**(2), 71–97 (1984)
17. Georgiou, S.D.: Orthogonal designs for computer experiments. J. Stat. Plann. Infer. **141**, 1519–1525 (2010)
18. Gu, L., Yang, J.: Construction of nearly orthogonal Latin hypercube designs. Biometrika **76**(6), 819–830 (2013)
19. Joseph, V., Hung, Y.: Orthogonal-maximin Latin hypercube designs. Statistica Sinica **18**, 171–186 (2008)
20. Yates, D., Moore, D., Starnes, D.: Introduction to the Practice of Statistics, 3rd edn. W. H. Freeman (2008)
21. Kroese, D., Taimre, T., Botev, Z.: Handbook of Monte Carlo Methods. Wiley, New York (2011)
22. Lemieux, C.: Monte Carlo and Quasi-Monte Carlo Sampling. Springer, New York (2009)
23. Sobol, I., Asotsky, D., Kreinin, A., Kucherenko, S.: Construction and comparison of high-dimensional Sobol. Technical paper, Wilmott Magazine, 64–79 (2011)
24. Wong, T., Luk, W., Heng, P.: Sampling with Hammersley and Halton points. J. Graph. Tools **2**(2), 9–24 (1997)
25. Johnson, M., Moore, L., Ylvisaker, D.: Minimax and maximin distance designs. J. Stat. Plann. Infer. **26**, 131–148 (1990)
26. Romero, V., Burkardt, J., Gunzburger, M., Peterson, J., Krishnamurthy, T.: Initial application and evaluation of a promising new sampling method for response surface generation: centroidal Voronoi tessellation. AIAA J. (2003)
27. Fang, K., Lin, D.: Uniform experimental designs and their applications in industry. In: Handbook of Statistics, vol. 22, pp. 131–170 (2003)
28. Ye, K., Lib, W., Sudjianto, A.: Algorithmic construction of optimal symmetric Latin hypercube designs. J. Stat. Plann. Infer. **90**(1), 145–159 (2000)
29. Burrage, K., Burrage, P., Donovan, D., Thompson, B.: Populations of models, experimental designs and coverage of parameter space by Latin hypercube and orthogonal sampling. Procedia Comput. Sci. **51**, 1762–1771 (2015)
30. Kappel, S.: Report on concept of surrogate model enhancement. ALEF Technical Reports - AIRBUS D2.2.2-2 (2011)
31. Box, G., Draper, N.: Empirical Model-Building and Response Surfaces. Wiley (1987)
32. Wang, J.: Geometric Structure of High-Dimensional Data and Dimensionality Reduction. Springer, Heidelberg, Dordrecht, London, New York (2011)
33. Wang, G., Shan, S.: Review of metamodeling techniques in support of engineering design optimization. J. Mech. Des. **129**(4), 370–380 (2006)
34. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. **57**(4), 483–531 (2015)
35. Schilders, W.H., der Vorst, H.A.V., Rommes, J.: Model Order Reduction: Theory, Research Aspects and Applications, vol. 13. Springer, Berlin, Heidelberg (2008)
36. Baur, U., Benner, P., Feng, L.: Model order reduction for linear and nonlinear systems: a system-theoretic perspective. J. Comput. Methods Eng. **21**(331) (2014)
37. Zahr, M.J., Amsallem, D., Farhat, C.; Construction of parametrically-robust CFD-based reduced-order models for PDE-constrained optimization. In: 21st AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Colocated Conferences, (AIAA 2013-2845) (2013)
38. Bui-Thanh, T., Damodaran, M., Willcox, K.: Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. AIAA J. **42**(8), 1505–1516 (2004)
39. Zimmermann, R., Vendl, A., Görtz, S.: Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. AIAA J. **52**(2), 255–266 (2014)

40. Amsallem, D., Zahr, M.J., Farhat, C.: Nonlinear model order reduction based on local reduced-order bases. Int. J. Numer. Methods Eng. **92**(10), 891–916 (2012)
41. Amsallem, D., Zahr, M.J., Washabaugh, K.: Fast local reduced basis updates for the efficient reduction of nonlinear systems with hyper-reduction. Adv. Comput. Math. **41**(5), 1187–1230 (2015)
42. Ryckelynck, D.: A priori hypereduction method: an adaptive approach. J. Comput. Phys. **202**(1), 346–366 (2005)
43. Franz, T., Zimmermann, R., Görtz, S., Karchera, N.: Interpolation-based reduced-order modelling for steady transonic flows via manifold learning. Int. J. Comput. Fluid Dyn. **28**(3–4), 106–121 (2014)
44. Tenenbaum, J., De Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science **290**, 2319–2323 (2000)
45. Leifsson, L., Koziel, S.: Simulation-Driven Aerodynamic Design Using Variable-Fidelity Models. Imperial College Press (2015)
46. Zhang, M., Tomacy, M., Wang, C., Rizzi, A.: Variable fidelity methods and surrogate modeling of critical loads on x-31 aircraft. AIAA J. (2013)
47. Görtz, S., Zimmermann, R., Han, Z.: Variable-fidelity and reduced-order models for aero data for loads predictions. In: Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Results of the Closing Symposium of the German Research Initiative ComFliTe, Braunschweig, Germany, vol. 123, pp. 99–112 (2012)
48. Tian, Z., Fong, S.: Survey of meta-heuristic algorithms for deep learning training. In: Baskan, O. (ed.) Optimization Algorithms—Methods and Applications (2016)
49. Jin, Y.: Surrogate-assisted evolutionary computation: recent advances and future challenges. Swarm Evol. Comput. **1**(2), 61–70 (2011)
50. Viana, F., Haftka, R.T.: Using multiple surrogates for metamodeling. In: 7th ASMO-UK ISSMO International Conference on Engineering Design Optimization, Bath, UK, pp. 1–18 (2008)
51. Goel, T., Haftka, R.T., Shyy, W., Queipo, N.V.: Ensemble of surrogates. Struct. Multidiscip. Optim. **33**, 199–216 (2009)
52. Papalambros, P.Y., Wilde, D.J.: Principles of Optimal Design: Modeling and Computation (Section 8.2—Computing Derivatives). Cambridge University Press, UK (2000)
53. Mack, Y., Goel, T., Shyy, W., Haftka, R.: Surrogate model-based optimization framework: a case study in aerospace design. In: Evolutionary Computation in Dynamic and Uncertain Environments. Studies in Computational Intelligence, vol. 51, pp. 323–342 (2007)
54. Jones, D., Schonlau, M., Welch, W.: Efficient global optimization of expensive black-box functions. J. Glob. Optim. **13**(4), 455–492 (1998)
55. Jones, D.R.: A taxonomy of global optimization methods based on response surfaces. J. Glob. Optim. **21**, 345–383 (2001)
56. Alexandrov, N., Dennis, J., Lewis, R., Torczon, V.: A trust region framework for managing the use of approximation models in optimization. Struct. Optim. **15**(1), 16–23 (1998)
57. Booker, A.J., Dennis, J.E., Frank, P.D., Serafini, D.B.: A rigorous framework for optimization of expensive functions by surrogates. Struct. Optim. **17**(1), 1–13 (1999)
58. Pan, G., Ye, P., Wang, P., Yang, Z.: A sequential optimization sampling method for metamodels with radial basis functions. Sci. World J. **2014**(Article ID 192862) (2014)
59. Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S.: Global Sensitivity Analysis: The Primer. Wiley (2008)
60. Kappel, S.: Report on surrogate model review results. ALEF Technical Reports - AIRBUS D2.2.1-2 (2011)
61. Mavris, D.N., Pinon, O.J.: An overview of design challenges and methods in aerospace engineering. In: Hammami, O., Krob, D., Voirin, J.-L. (eds.) Complex Systems Design and Management: Proceedings of the Second International Conference on Complex Systems Design and Management (CSDM 2011), pp. 1–25. Springer, Berlin, Heidelberg (2012)

62. Grewal, A., Zimcik, D.: Development of reduced order aerodynamic models from an open source CFD code. In: Proceedings of the AVT-173, Virtual Prototyping of Affordable Military Vehicles Using Advanced MDO, no. RTO-MP-AVT-173 in NATO Research and Technology Organization, Bugaria (2011)
63. Abbas-Bayoumi, A., Becker, K.: An industrial view on numerical simulation for aircraft aerodynamic design. J. Math. Ind. **1**(10) (2011)
64. Kroll, N., Gauger, N.R., Brezillon, J., et al.: Flow simulation and shape optimization for aircraft design. J. Comput. Phys. **203**(2), 397–411 (2007)

Constrained Single-Point Aerodynamic Shape Optimization of the DPW-W1 Wing Through Evolutionary Programming and Support Vector Machines

E. Andrés-Pérez, D. González-Juárez, M. J. Martin-Burgos
and L. Carro-Calvo

Abstract The application of surrogate-based methods to the constrained optimization of aerodynamic shapes is nowadays a very active research field due to the potential of these methods to reduce the number of actual computational fluid dynamics simulation runs, and therefore drastically speed-up the design process. However, their feasibility when handling a large number of design parameters, which in fact is the case in industrial configurations, remains unclear and needs further efforts, as demonstrated by recent research on design space reduction techniques and adaptive sampling strategies. This paper presents the results of applying surrogate-based optimization to the three-dimensional, constrained aerodynamic shape design of the DPW-W1 wing, involving both inviscid and viscous transonic flow. The wing geometry is parameterized by a control box with 36 design variables and the applied approach is based on the use of Support Vector Machines (SVMs) as the surrogate model for estimating the objective function, in combination with an Evolutionary

E. Andrés-Pérez (✉)
Fluid Dynamics Branch, Spanish National Institute for Aerospace Technology (INTA/ISDEFE),
Ctra. de Ajalvir Km. 4.5, Torrejón de Ardoz, 28850 Madrid, Spain
e-mail: eandres@isdefe.es; esther.andres@upm.es

E. Andrés-Pérez
Technical University of Madrid (UPM), Madrid, Spain

D. González-Juárez · M. J. Martin-Burgos
Fluid Dynamics Branch, Spanish National Institute for Aerospace Technology (INTA), Ctra. de
Ajalvir Km. 4.5, Torrejón de Ardoz, 28850 Madrid, Spain
e-mail: gonzalezjd@inta.es

M. J. Martin-Burgos
e-mail: martinbj@inta.es

L. Carro-Calvo
Department of Signal Theory and Communications, University of Alcala (UAH), Ctra.
Madrid-Barcelona km 33, 28871 Alcalá de Henares, Madrid, Spain
e-mail: leopoldo.carro@uah.es

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_3

Algorithm (EA) and an adaptive sampling technique focused on optimization, called the Intelligent Estimation Search with Sequential Learning (IES-SL).

1 Introduction and Previous Works

1.1 Introduction

In the last few years, there has been an increasing interest in the topic of Surrogate-based Optimization (SBO) methods for aerodynamic shape design. This is due to the promising potential of these methods to speed-up the design process by the use of a "low cost" objective function evaluation to reduce the required number of expensive computational fluid dynamics (CFD) simulations. However, the application of these SBO methods for industrial configurations still requires facing several challenges, such as the so-called "curse of dimensionality", the ability of surrogates when handling a high number of design parameters, efficient constraints handling, adequate exploration and exploitation of the design space, etc.

1.2 Recent Research Efforts in SBO for Aerodynamic Shape Design

A physics-based surrogate model was recently applied in [1] to the drag minimization of a NACA0012 airfoil in inviscid transonic flow and a RAE2822 airfoil in viscous transonic flow, both using the PARSEC parameterization with up to ten design parameters. The drag minimization problem was also addressed by SBO in [2] for the NLF0416 airfoil, parameterized with ten design parameters.

Moreover, a combination of a genetic algorithm (GA) and an artificial neural network (ANN) was applied in [3] to the shape optimization of an airfoil, parameterized by a modified PARSEC parameterization involving ten design variables. In [4] a surrogate based on Proper Orthogonal Decomposition (POD) was applied to the aerodynamic shape optimization of an airfoil geometry parameterized by sixteen design variables defined with Class Shape Transformation method (CST). In summary, the ability of SBO methods to manage a high number of design parameters still remains an open challenge and have been studied by several authors in the last few years, as well as the strategies for efficient infill sampling criteria with constraint handling [4, 5].

Finally, the authors also presented recent works on this topic [6, 7]. This paper is an extension of previous research, here considering the constrained single-point aerodynamic optimization of the DPW-W1 wing for both inviscid and viscous transonic flow.

1.3 Garteur AD/AG52

This work is part of a GARTEUR (Group for Aeronautical Research and Technology in Europe, www.garteur.org) Action Group that was established to explore these SBO approaches. The main objective of the AG [8] is, by means of a European collaborative research, to make a deep evaluation and assessment of surrogate-based global optimization methods for aerodynamic shape design.

2 Definition of the Optimization Problem

2.1 Baseline Geometry: DPW-W1 Wing

The public domain transonic DPW-W1 wing (a test case of the Third AIAA Drag Prediction Workshop) was used [9, 10]. Reference quantities for this wing are displayed in the following table (Table 1):

The initial geometry (in IGES format) was downloaded from [9]. A set of grids are also available in the website of the 3rd AIAA Workshop on Drag Prediction (Table 2).

2.2 Parameterization

The DPW geometry is parameterized by a 3D control box (displayed in Fig. 1) with 5 control points in direction u, 10 in direction v and 5 in direction w. The parametric

Table 1 DPW reference quantities

Sref (wing ref. area)	290,322 mm^2
Cref (wing ref. chord)	197.55 mm
Xref	154.24 mm (relative to the wing root leading edge)
b/2 (semi span)	762 mm
AR (aspect ratio, AR = b2/Sref	8.0

Table 2 Computational grids

	#points (k)	#surface points (k)	#elements (k)	#surface elements (k)
DPW-EULER	427	135	2112	276
DPW RANS[a]	3770	152	9335	310

[a] The DPW RANS grid was downloaded directly from the 3rd Workshop on Drag Prediction web page

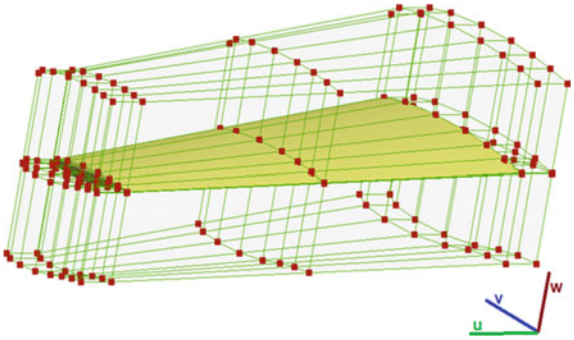


Fig. 1 DPW wing parameterization

u direction corresponds to the y axis, the v direction to the x axis, and the w direction to the z axis.

The design variables are the vertical displacement of those control points set up on the aerodynamic surface. The wing is split in three profile sections and the transition between sections is linear. Each section has 6 active control points for the upper side and other 6 for the lower side, which are independent (the movement of a control point at the upper side does not modify the lower side and vice versa), with a total of 36 design parameters for the whole wing. Authors have previously applied this parameterization technique to other local and global optimization problems [11]. During the optimization performed in this paper, the wing planform will be kept fixed, as well the angle of attack and the twist angle distribution.

2.3 Aerodynamic Constraints

The following aerodynamic constraints are considered:

1. Prescribed constant lift coefficient ($C_L = C_L^0$)
2. Minimum pitching moment: $C_M \geq C_M^0$
3. Drag penalty: If constraint in minimum pitching moment is not satisfied, the penalty will be 1 drag count per 0.01 increment in C_M.

2.4 Geometric Constraints

Each design variable will be constrained by its minimum and maximum values that will be chosen as the + or −20% of their original value. Apart from this, other constraints have been defined, according to [10]:

1. Airfoils' maximum thickness constraints:

$$(t/c)_{section} \geq (t/c)^{0}_{section}$$

where the right term is the maximum thickness for the original wing sections, root, mid-span and tip, which has the value of 13.5%.

2. Beam constraints

First, two locations (x/c) are fixed to represent the beam constraints:

$$(x/c)_{root,1} = (x/c)_{mid-span,1} = (x/c)_{tip,1} = 0.20$$
$$(x/c)_{root,2} = (x/c)_{mid-span,2} = (x/c)_{tip,2} = 0.75$$

The constraint here is that the thickness value of the optimized wing sections at these locations should be greater or equal than the thickness of the original ones. It is defined with the expressions:

$$(t/c)_{root,1} \geq 12\%, (t/c)_{mid-span,1} \geq 12\%, (t/c)_{tip,1} \geq 12\%$$
$$(t/c)_{root,2} \geq 5.9\%, (t/c)_{mid-span,2} \geq 5.9\%, (t/c)_{tip,2} \geq 5.9$$

2.5 Design Point and Objective Function

This paper addresses a single-point optimization of the DPW-W1 wing, for both inviscid and viscous transonic flow. Multi-point optimization will be also considered as a future work within the GARTEUR AG52 group [8]. The flow conditions are: Mach number 0.8, an angle of attack of zero degrees and a Reynolds number of 5×10^6. The design goal is to achieve a geometry with the minimum drag, while maintaining the specified aerodynamic constraints. Aerodynamic constraints are implemented as penalties in the objective function. The pseudo-code implementation is:

```
lift_penalty=1-(Cl/Cl0);
if (lift_penalty<0) lift_penalty=0;
cm_penalty=(Cm0-Cm)*0.0001/0.01;
if (cm_penalty < 0) cm_penalty=0;
objective_function=(((Cd+cm_penalty)/Cd0))+5*lift_penalty;
```

2.6 Computational Grids

The following unstructured grids were used:

3 Description of the Applied Approach

3.1 *Adaptive Sampling Focused on Optimization*

The Intelligent Estimation Search with Sequential Learning (IES-SL) is an algorithm designed to implement an adaptive sampling directly focused on the optimization search. From this point of view, the key feature of this novel approach is to use the surrogate model to estimate the location of the optimum in the real function. To do this, an optimization search is applied over the surrogate, obtaining an estimated value of the real minimum position (an “intelligent guess”). Each of the estimations of the optimum location gives us a new sampling point (it means a new geometry that is also analyzed using the high fidelity CFD solver). Within a try-and-error cycle, the surrogate proposes a new design which is again evaluated by the CFD solver and then, in a sequential learning, the surrogate model is enriched with the associated cost function.

3.2 *Support Vector Regression Algorithm as Surrogate Model*

SVMs represent appealing algorithms for a large variety of regression problems due to they do not only take into account the error approximation to the data, but also the generalization of the model, namely, their capability to improve the prediction of the model when new data are evaluated. This kind of methods can be considered a specific type of ANNs, and are commonly trained by means of a deterministic method known as Sequential Minimal Optimization (SMO) which provides a significant computational complexity reduction. The used SVM method for regression consists of, given a set of training vectors $C = \{(x_i, y_i), i = 1, \ldots, l\}$, training a model of the form $y(x) = f(x) + b = w^T \phi(x) + b$, to minimize a general risk function of the form

$$R[f] = \frac{1}{2}\|w\|^2 + \frac{1}{2}C\sum_{i=1}^{l} L(y_i, f(x)) \tag{1}$$

where w controls the smoothness of the model, $\phi(x)$ is a function of projection of the input space to the feature space, b is a parameter of bias, $\mathrm{x_i}$ is a feature vector of the input space with dimension N, $\mathrm{y_i}$ is the output value to be estimated and $L(y_i, f(x))$ is the loss function selected. In this paper, the L1-SVR (L1 support vector regression) is used, characterized by an ε-insensitive loss function.

$$L(y_i, f(x)) = |y_i - f(x_i)|_\varepsilon \tag{2}$$

In order to train this model, it is necessary to solve the following optimization problem

$$\min(\frac{1}{2}\|w\|^2 + \frac{1}{2}C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \tag{3}$$

subject to:

$$\begin{aligned} & y_i - w^T\phi(x_i) - b \le \varepsilon + \xi_i, i = 1, \ldots, l \\ & -y_i + w^T\phi(x_i) + b \le \varepsilon + \xi_i^*, i = 1, \ldots, l \\ & \xi_i, \xi_i^* \ge 0, i = 1, \ldots, l \end{aligned} \tag{4}$$

The SVM can use different kernels to face non-lineal problems. On this case, a radial basis function has been used as a kernel function. This training procedure must be combined with the search of three parameters (C, ε, and γ, named hyperparemeters) on which the final model depends. The influence of the three parameters on the SVM model can be seen on Eq. (3) where C defines the optimization problem and in Eq. (4) where ε represents the constraints for the optimization problems. Finally, the radial basis kernel depends on the value of γ. To obtain the best SVM performance, a search of the most suitable combination of these three parameters must be carried on, usually by using cross validation techniques over the training set. To reduce the computational time of this process, different methods have been proposed in the literature to reduce the search space related to these parameters. In this case, it has been applied the one developed in [12], which has proven to require pretty short search times.

3.3 Evolutionary Programming

The EA implemented for this work has the following characteristics: the selection operator is applied by replacing a portion of the current generation by new individuals generated from parents. It is considered the replacement of the individuals in the population with fitness value under the population's mean fitness. A multipoint crossover which selects the value of one of the parents with probability 0.5 is applied. Regarding the mutation operator, the values of each new individual are mutated with probability 1/Np, where Np is the number of parameters to be optimized. A mutation parameter can be tuned in order to allow a more global or local search over a certain design variable. More detailed information about the implemented algorithm can be found in [7].

3.4 Handling Constrains Within the Optimization Process

In the context of evolutionary optimization, constraints can be handled by adding penalties to the objective function. These penalties can be imposed in a 'soft' or 'hard' manner. Soft penalties increment the unconstrained objective function proportionally with information about how far from the constraint is a certain solution. This kind of penalties allows the system to work with non-feasible but interesting solutions, improving the search space and finally obtaining feasible solutions. Hard penalties imply that the restriction must be fulfilled at any time of the optimization process. In this case, the solutions are strongly penalized and therefore removed from the search process. The constrained objective function is represented by the following expression:

$$fobj = fobj^{*} + soft\_penalt + hard\_penalt \tag{5}$$

where fobj* is the unconstrained objective function.

In this paper, the constraints within the SBO process are handled in the following way. First, the simulation system M allows computing the aerodynamic characteristics (i.e. C_D, C_L, C_M) of a geometry defined by a set of parameters P.

$$[C_D, C_L, C_M] = M(P) \tag{6}$$

The objective function to be minimized, including the mentioned constraints, can be described as a combination of the output factors from the simulation system (aerodynamic characteristics), and other factors associated with the model (i.e. geometric characteristics). This function can be represented as:

$$fobj = f(C_D, C_L, C_M, P) \tag{7}$$

where the geometric restrictions are calculated from the set of model parameters P and the aerodynamic constrains are directly computed from the aerodynamic characteristics. Since the application of the simulator system to obtain the aerodynamic values is very expensive, the surrogate model is added to the system to reduce the computational cost of the optimization process. There are three different approaches to use the surrogate model to speed up the constrained objective function computation.

The first option is to generate a surrogate that directly models the objective function. This is the simplest and more direct method to apply the surrogate. The individual evaluation on this case is carried on by the surrogate model (i.e. the SVM).

$$fobj = SVM(P) \approx f(C_D, C_L, C_M, P) \tag{8}$$

The second option is to extract from the objective function the restrictions that are independent from the simulator output. In this case, the geometric constraints can be calculated independently:

$$\begin{aligned} fobj &= f(C_D, C_L, C_M, P) \\ &= f1(C_D, C_L, C_M) + f2(P) \\ SVM(P) &\approx f1(C_D, C_L, C_M) \\ fobj &= SVM(P) + f2(P) \approx f(C_D, C_L, C_M, P) \end{aligned} \tag{9}$$

This division reduces the complexity of the surrogate model, because it does not have to model the geometric information (only aerodynamic features). On the other hand, the system must perform an additional computation since in each evaluation the system must compute both the SVM output and the value of f2.

The third option is another step to simplify the surrogate model. The penalties in the objective function can add additional complexity to the function like discontinuities around the restriction boundaries. This can reduce the quality that a surrogate can achieve with a fixed number of data points. To avoid this effect, a multi-surrogate model can be implemented, and this is the approach considered in this paper. Each simulator output (i.e. C_D, C_L, C_M), is modeled by a different SVM (SVM_Cd, SVM_Cl, SVM_Cm), and then applied to the f1 function that contains the aerodynamic restrictions.

$$\begin{aligned} fobj = &f1(SVM\_Cd(P), SVM\_Cl(P), SVM\_Cm(P)) \\ &+ f2(P) \approx f(C_D, C_L, C_M, P)) \end{aligned} \tag{10}$$

In this way, the penalties associated to f1, and their corresponding complexity, are added after building the surrogate, allowing achieving simpler SVM models, with higher quality and accuracy. On the other hand, the global system is more complex, since now it is necessary to train now three different surrogates.

4 Numerical Results

The proposed approach is applied to the constrained single point optimization of the DPW-W1 wing in both inviscid and viscous transonic flow conditions. In order to handle the geometric constraints previously defined, the parameterization is prepared by locating certain control points in specific locations, as displayed in Fig. 2.

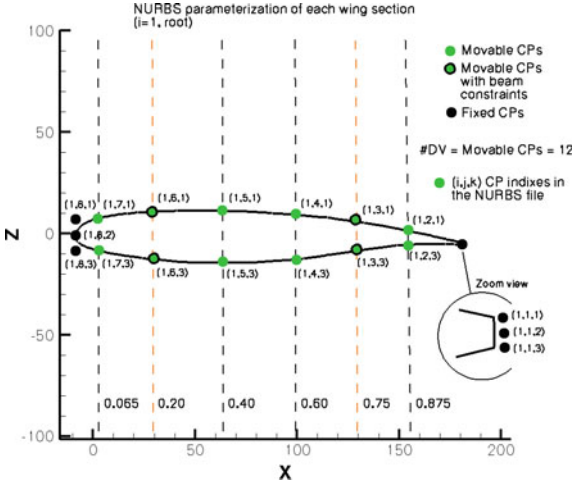


Fig. 2 Geometric constraints handling through the selected control box parameterization (wing section)

Table 3 Objective function and aerodynamic coefficients of baseline and optimized geometries

	fobj	Cd	Cl	Cm
DPW-W1	1	0.0307	0.5984	−0.02867
Optimized	0.77	0.0236	0.5981	−0.02653

4.1 *Inviscid Transonic Flow*

The following table shows the objective function of the original and optimized geometries. The results show that the objective function has been improved by 23% (after 192 iterations), while both aerodynamic and geometric the constraints have been satisfied (Table 3).

Figures 3, 4 and 5 show the Mach number distribution, shapes and Cp comparison between the baseline geometry and the optimized shape.

The computational time for the whole optimization of the Euler case using 8 processors on a Linux x86_64 computational cluster was about 40 h.

4.2 *Viscous Transonic Flow*

The following table shows the objective function of the original and optimized geometries. The results show that the objective function has been improved by 5%, while both aerodynamic and geometric the constraints have been satisfied (Table 4).

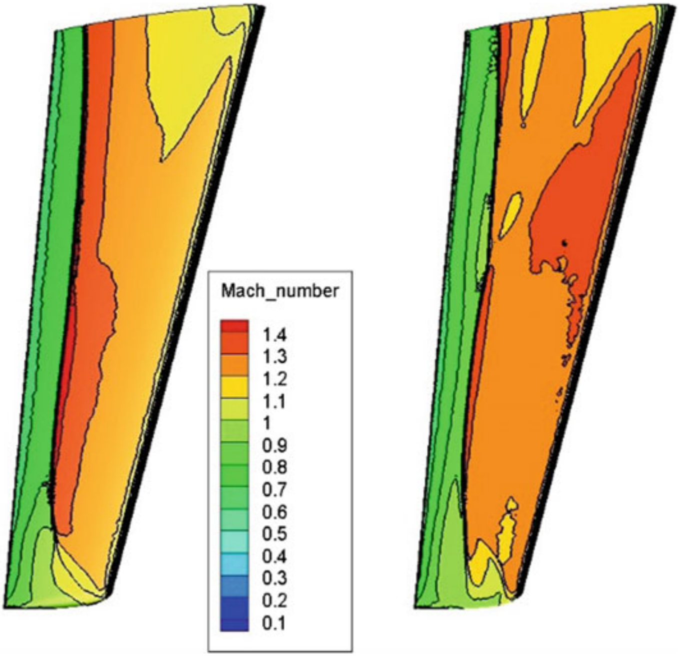


Fig. 3 Mach number distribution on the original (left) and optimized (right) geometries

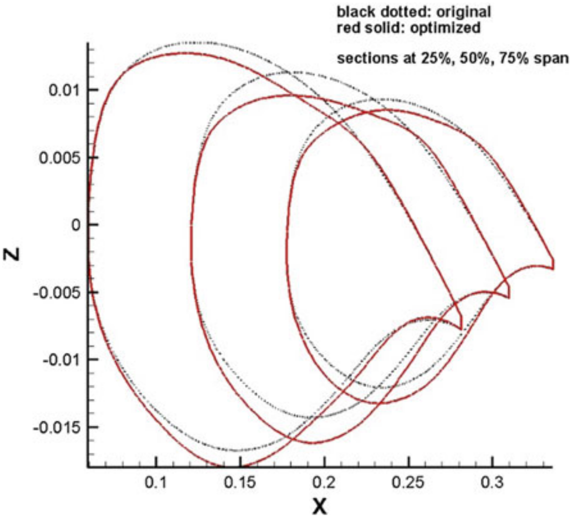


Fig. 4 Original versus optimized geometries

Figures 6, 7 and 8 show the shapes and Cp comparison between the baseline geometry and the optimized shape, which was obtained in the iteration number 175.

Table 4 Objective function and aerodynamic coefficients of baseline and optimized geometries

	fobj	Cd	Cl	Cm
DPW-W1	1	0.0257	0.3636	−0.0687
Optimized	0.95	0.0245	0.3658	−0.0684

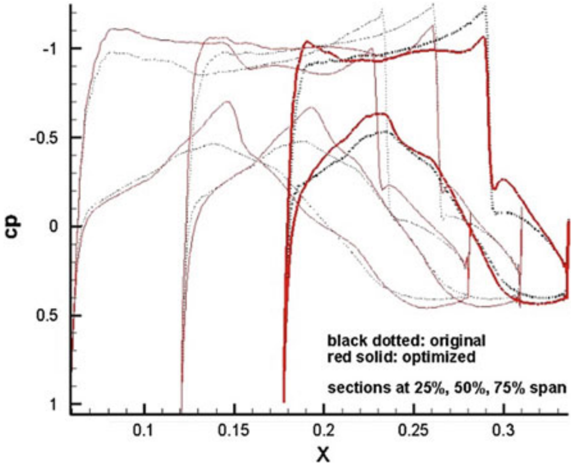


Fig. 5 Cp plots along wing span

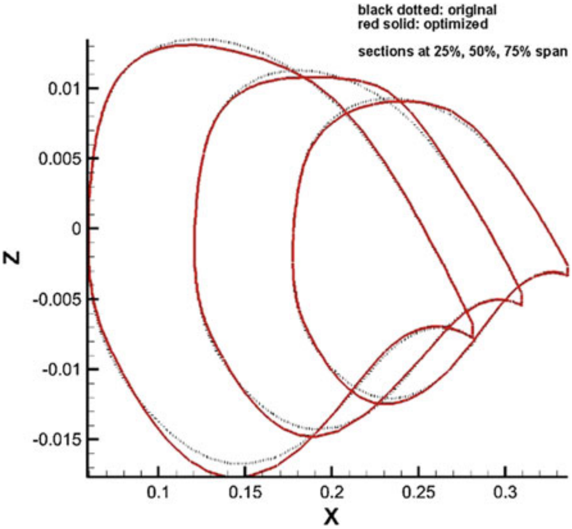


Fig. 6 Original versus optimized geometries

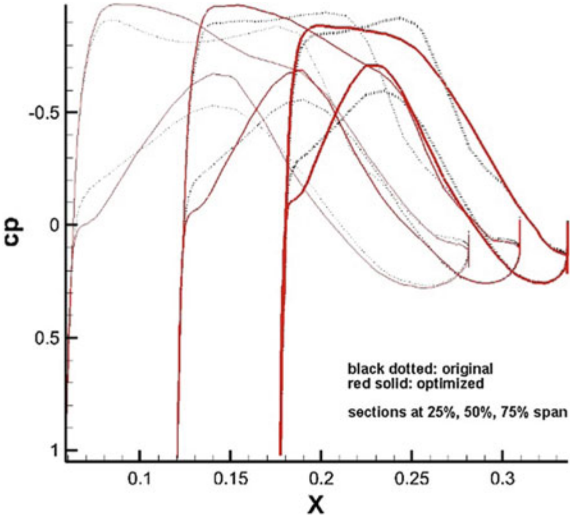


Fig. 7 Cp plots along wing span

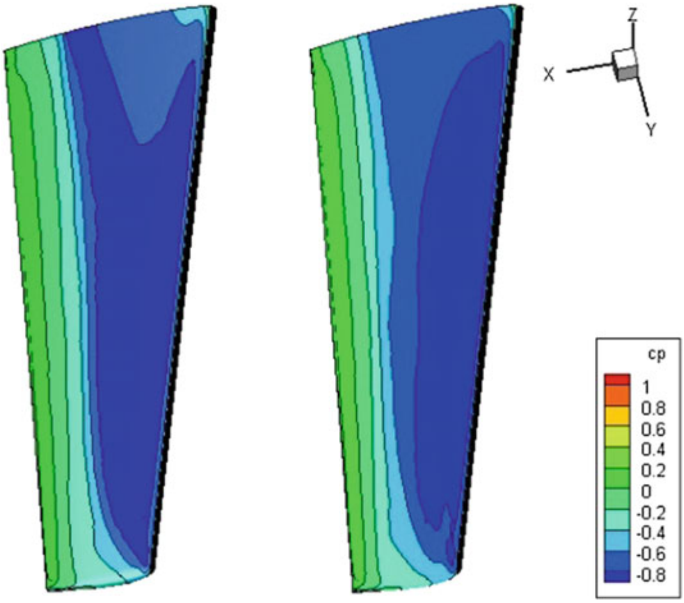


Fig. 8 Cp distribution on the original (left) and optimized (right) geometries

The computational time for 175 iterations of the RANS case using 36 processors on a Linux x86_64 computational cluster was about 170 h (7 days).

5 Conclusions

This paper presented the application of a global optimization strategy using the Intelligent Estimation Search with Sequential Learning (IES-SL) and the hybridization of EA and SVMr to the single-point constrained optimization of a three dimensional DPW wing in both inviscid and viscous transonic flow conditions, showing first promising results.

Future work will address the multi-point constrained optimization, for comparison with the results obtained by Epstein and Jameson in [10]. This extension will be performed within the GARTEUR AG52 group. In addition, research work on the parameterization sensitivity to the SBO process is also being performed.

Acknowledgements The research described in this paper made by INTA and UAH researchers has been supported under the INTA activity "*Termofluidodinámica*" (IGB99001), the R&T DREPANO project (RTC-2014-1593-4), the INTA "*Rafael Calvo Rodés*" grants and the Engineering Department of ISDEFE.

References

1. Leifsson, L., Koziel, S., Tesfahunegn, Y.: Aerodynamic design optimization: physics-based surrogate approaches for airfoil and wing design. In: AIAA SciTech 2014. AIAA 2014-0572
2. Koziel, S., Leifsson, L.: Multi-level surrogate-based airfoil shape optimization. In: 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Grapevine (Dallas/Ft. Worth Region), Texas. AIAA2013-0778, 07–10 Jan 2013
3. Jahangirian, A., Shahrokhi, A.: Aerodynamic shape optimization using efficient evolutionary algorithms and unstructured CFD solver. Comput. Fluids **46**, 270–276 (2011)
4. Iuliano, E., Quagliarella, D.: Aerodynamic shape optimization via non-intrusive POD-based surrogate modelling. In: 2013 IEEE Congress on Evolutionary Computation, Cancún, México, 20–23 June
5. Parr, J., Holden, C., Forrester A., Keane, A.: Review of efficient surrogate infill sampling criteria with constraint handling. In: 2nd International Conference on Engineering Optimization, Lisbon, Portugal, Sept 2010
6. Andres, E., Salcedo-Sanz, S., Monge, F., Perez-Bellido, A.M.: Efficient aerodynamic design through evolutionary programming and support vector regression algorithms. Expert Syst. Appl. **39**, 10700–10708 (2012)
7. Andres, E., Carro-Calvo, L., Salcedo-Sanz, S.: Fast aerodynamic coefficients prediction using SVMs for global shape optimization. In: ECCOMAS CFD, Barcelona, June 2014
8. Garteur AD/AG52 members 2013. Partial report on surrogate-based global optimization methods in aerodynamic design. https://ag52.blogspot.com
9. Web site of the 3rd AIAA Workshop on Drag Prediction: http://aaac.larc.nasa.gov/tsab/cfdlarc/aiaa-dpw/Workshop3/workshop3.html
10. Epstein, B., Jameson, A., Peigin, S., Roman, D., Harrison, N., Vassberg, J.: Comparative study of 3D wing drag minimization by different optimization techniques. In: 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada. AIAA2008-326, Jan 2008
11. Martin, M.J., Andres, E., Valero, E., Lozano, C.: Gradients calculation for arbitrary parameterizations via volumetric NURBS: the control box approach. In: EUCASS Conference (2013)
12. Ortiz-García, E.G., Salcedo Sanz, S., Pérez-Bellido, Á.M., Portilla-Figueras, J.A.: Improving the training time of support vector regression algorithms through novel hyper-parameters search space reductions. Neurocomputing (2009)

Enabling of Large Scale Aerodynamic Shape Optimization Through POD-Based Reduced-Order Modeling and Free Form Deformation

A. Scardigli, R. Arpa, A. Chiarini and H. Telib

Abstract We present an approach for shape optimization of large-scale aerodynamic problems, combining free-form deformation and POD based reduced-order modeling. An extension of the classical Free-Form Deformation techniques is derived in order to handle efficiently constraints regarding fixed and deformable portions of the geometry and to impose the smoothness at the interface between the two different regions. The second aspect concerns the development of a hybrid model, combining a POD-based reduced order model and a industrial CFD solver using a domain-decomposition approach. A method on how determining automatically a suitable domain decomposition is discussed. The effectiveness and drawback of the above techniques are highlighted on a large-scale aerodynamic shape optimization and control problem, i.e. the mainsail thrust optimization of a sailing boat.

1 Introduction

Besides being easily interfaceable with CFD solvers, evolutionary algorithms have the capability to find the global optimum, a feature that is often consider attractive in industrial application. On the other hand, they require a larger number of functional evaluations than deterministic methods, and this number grows super-linearly with

A. Scardigli (✉) · R. Arpa · H. Telib (✉)
OPTIMAD Engineering, Via G. Collegno 18, Turin, Italy
e-mail: angela.scardigli@optimad.it

H. Telib
e-mail: haysam.telib@optimad.it

R. Arpa
e-mail: rocco.arpa@optimad.it

A. Scardigli
Dipartimento di Scienze Matematiche, Politecnico di Torino, Turin, Italy

A. Chiarini
SCS Super Computing Solutions, Casalecchio di Reno, Italy
e-mail: a.chiarini@scsitaly.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_4

the number of design parameters. For this reason low-order approximation of the functionals, like kriging and response surfaces, represent a key-point of evolutionary optimization. Nevertheless for large-scale aerodynamics problems, the computational cost associated with high-order functional evaluations needed for the training of the response surface is usually prohibitive within an evolutionary design loop: for example, the driving force evaluation of a sailing boat requires typically a number of grid point of order O $(10 - 100 \times 10^6)$ and O $(1 - 10 \times 10^3)$ CPU hours for a standard RANS simulation. Such a cost is sustainable only for strategic aerodynamic optimization problems as found for example in the aeronautical industry. For both evolutionary and deterministic methods, another general burden is the interface to CAD and parameterization of complex geometries. The capacity of modifying the geometry resides within the CAD program and the designer, despite the fact that neither of them is available in an automated design loop. We address the problems discussed above as follows: in Sect. 2 we present our approach to parameterization of complex geometries, with focus on continuity constraints handling. In Sect. 3, the

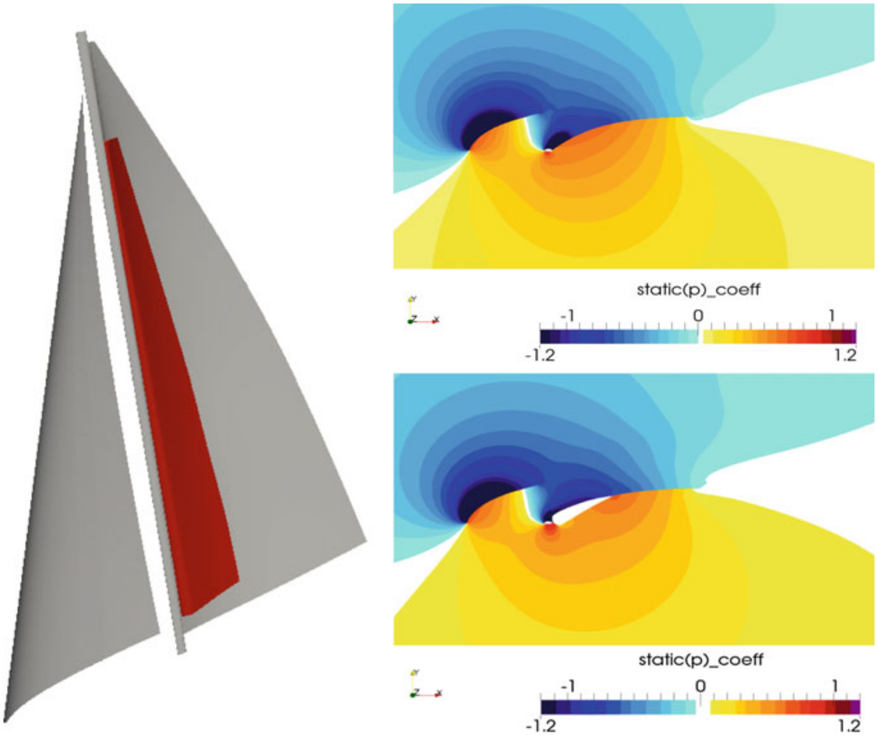


Fig. 1 Initial configuration of sails in the boat. The inflatable device is highlighted in red. Effect of the inflatable device on pressure field around sails is shown, in comparison with a classical sail configuration

implementation of a reduced order model (ROM) based on a domain-decomposition method to be employed in a multi-fidelity framework is described and discussed.

Finally, we demonstrate our results for a combined shape optimization and optimal control problem applied to a sailing boat with inflatable devices, which modify the shape of the mainsail profile (see Fig. 1). The final aim is to find the optimal shape of the inflatable device and the correct trim angles of the mainsail and the jib, under assumptions of rigid sails, close-hauled navigation and fixed wind conditions in order to maximize the thrust developed by the sails. The aim of introducing an inflatable device around the mainsail is twofold: (i) the inflatable chamber reduces the separation zone between the mast and allows the flow to recover faster. (ii) the thickness distribution introduces a new set of design variables which may be used in order to enhance the sail efficiency. It is crucial to introduce manufacturing constraints within the shape parameterization in order to avoid non-feasible designs. In particular we will consider that (i) the inflatable chambers must be symmetric on the port and starboard surface of the mainsail and that (ii) a fold is created where the inflatable part attaches to the mainsail.

2 Geometry Parametrization

Free Form Deformation (FFD) [1, 2] has been developed in the frame of computer graphics in order to morph 3D objects, and it is widely used in several CAD/CAE programs for quick modifications of the original geometry. More recently, Désidéri and co-workers [3] used FFD algorithms to manipulate geometry in the context of aerodynamic shape optimization for aeronautical applications. The basic idea is to wrap the 3D-object in an envelope/3D-grid of control nodes (lattice) and define a deformation field over the space embedded in it, by means of classical B-Spline/Bézier volume parameterization. Regardless of the geometrical description of the object, its shape can be deformed by manipulating those control nodes. Generally, the lattices are simple hexahedra, cylinders or spheres [2, 3] which can be easily mapped on their unit elemental primitives.

2.1 Continuity on Arbitrary Shaped Boundaries

Although this approach is direct and very flexible, practically it is only suited when global deformations need to be introduced or when no constraints are present (see Fig. 2). For real-life applications it is necessary to use arbitrary shaped lattices or to impose arbitrary shape constraints. In other words, an accurate control over deformable and undeformable portions of the surface is necessary, together with the possibility to prescribe the continuity condition between these regions.

We denote with Ω the deformable part of the entire surface and with Γ its boundary. $\mathbf{S}$ represents the tesselation of Ω, and a displacement field $\mathbf{Q}$ obtained through

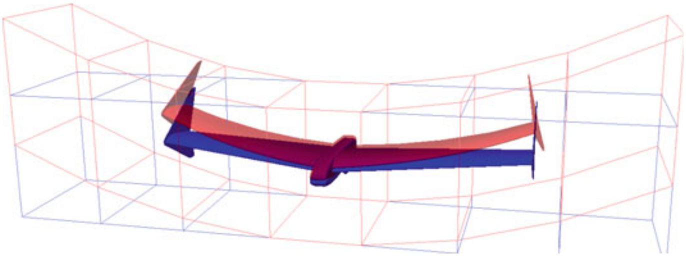


Fig. 2 Example of FFD global manipulation: original shape and lattice in blue, deformed ones in red

an arbitrary FFD technique. The new tesselation reads:

$$\mathbf{S}^* = \mathbf{S} + \mathbf{Q}(\mathbf{x}, \alpha^{(1\cdots J)}) \tag{1}$$

where $\mathbf{S}^*$ is the deformed surface.

We want to restrict the effect of the displacement field on a portion Ω, so that a continuity constraint (G^0, G^1, G^2) can be imposed between the deformable and undeformable parts of the surface. Our key idea is to introduce a filter scalar function $w(\phi(\mathbf{S}|\Gamma))$, with values between 0 and 1, so that:

$$1.\ w = 0 \text{ outside } \Omega$$
$$2.\ w \geq 0 \text{ inside } \Omega$$

and where $\phi(\mathbf{S}|\Gamma)$ is a function which represents the topological information of each point of $\mathbf{S}$ with respect to the boundary Γ. This implies that $\phi(\Gamma|\Gamma) = 0$ and that $\phi(\mathbf{x}^1|\Gamma) > \phi(\mathbf{x}^2|\Gamma)$ if $\mathbf{x}^2$ is closer to Γ than $\mathbf{x}^1$. Since we require that $\phi(\Gamma|\Gamma) = 0$ we denote this function the geodesic level set function.

The new constrained deformation $\mathbf{S}^{**}$ can be determined by modulating $\mathbf{Q}$ through w:

$$\mathbf{S}^{**} = \mathbf{S} + w\,(\phi(\mathbf{S}|\Gamma))\,\mathbf{Q}(\mathbf{x}, \alpha^{(1\cdots J)}) \tag{2}$$

By differentiating Eq. (2) twice, it can be seen that it is sufficient that in order to obtain

G^0 that $\mathbf{S}$, $\mathbf{Q}(\cdot,\cdot)$ are G^0, $w(\cdot)$, $\phi(\cdot)$ are C^0, and $w(\Gamma) = 0$
G^1 that $\mathbf{S}$, $\mathbf{Q}(\cdot,\cdot)$ are G^1, $w(\cdot)$, $\phi(\cdot)$ are C^1, and $w(\Gamma) = 0$, $w'(\Gamma) = 0$
G^2 that $\mathbf{S}$, $\mathbf{Q}(\cdot,\cdot)$ are G^2, $w(\cdot)$, $\phi(\cdot)$ are C^2, and $w(\Gamma) = 0$, $w'(\Gamma) = 0$, $w''(\Gamma) = 0$

Some of these requirements are easily guaranteed:

- the final geometry cannot be smoother than the original geometry $\mathbf{S}$
- the deformation fields $\mathbf{Q}$ derive from a tensor-product of Bezier curves and are continuous to any order
- the weight functions $w(\cdot)$ are chosen to fulfill the requested requirements

A candidate for $\phi(\mathbf{S}|\Gamma)$ is the geodesic distance function. This function can be calculated for example by using a Fast Marching technique [4]. However, depending on the shape of Γ, discontinuities in the gradients of the distance function can develop. Consequently, the deformation field can be guaranteed at most to be G^0 continuous.

In order to assure smoothness, a more regular level set field must be determined. Crane et al. [5] proposed a new method to approximately evaluate the geodesic distance on a 3D surface mesh, based on the solution of the time-dependent heat equation. The method consists in the following steps:

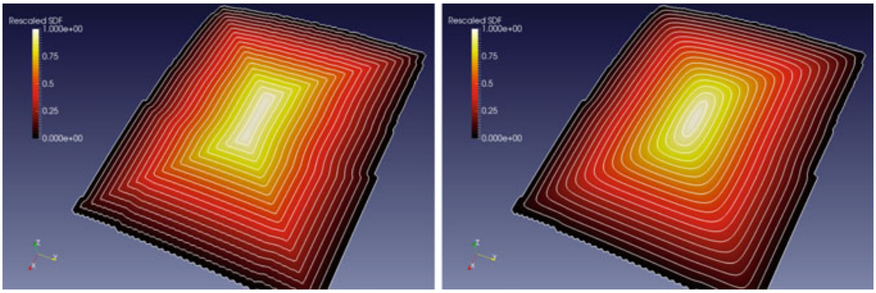
1. solve the heat equation $u_t = \Delta u$ on Ω, with the boundary condition $u = 1$ on Γ, up to some fixed time t_{end}
2. evaluate the vector field of the normalized gradient of the solution $\mathbf{X} = \frac{\nabla u}{\|\nabla u\|}$
3. solve the Poisson equation $\Delta\phi = -\nabla \cdot \mathbf{X}$, with Dirichlet boundary condition on Γ

In this case, the desired level of smoothness of the level set solution can be achieved tuning conveniently the heat propagation time t_{end} (see Fig. 3), without losing the topological information of the exact solution. In Fig. 4 an example of a resulting deformation is presented. In this case the weight function is such that an angle between the un-deformed and the deformed part is introduced, while the interior of the deformation is G^2 continuous.

2.2 *Direct Surface Manipulation*

Reinventing the wheel, the heat kernel level set representation of a 3D tessellated surface can be used to generate directly its parameterization without any extra efforts.

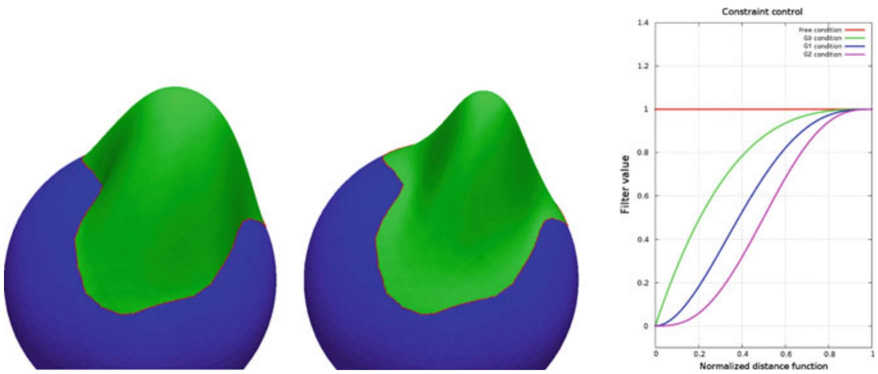
Lets have a deformable surface Ω, bounded by a generic contour Γ. Whenever it is possible to split Γ in four distinct, consecutive curves Γ_1, Γ_2, Γ_3 and Γ_4, a



(a) Exact Level-Set

(b) Smoothed Level-Set

Fig. 3 Exact geodesic distance function (left) and smoothed geodesic distance via heat kernel method (right) on a quadrilateral patch with irregular boundary contour



(a) G_0 continuity

(b) G_2 continuity

(c) Constraint control by means of filter functions

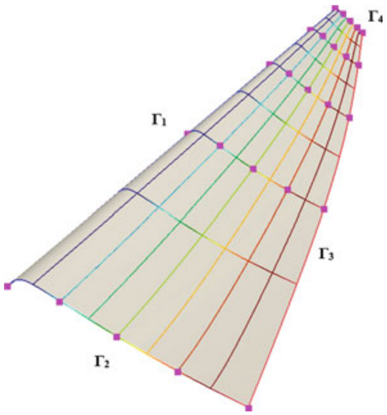
Fig. 4 Deformation of a sphere imposing different continuity constraints on the boundary of the green deformable patch

bi-variant map $(\xi, \eta)^t = \mathbf{M}(\mathbf{S}) \in [0, 1] \times [0, 1]$ of the original surface can be calculated, where:

- ξ is the rescaled heat kernel level set calculated on Ω, imposing Dirichlet conditions on Γ_1 and Γ_3, and natural conditions elsewhere
- η is the rescaled heat kernel level set calculated on Ω, imposing Dirichlet conditions on Γ_2 and Γ_4, and natural conditions elsewhere

This kind of approach allows to map the original 3D surface on the new bi-dimensional space (ξ,η) as a unit square, where the parameterization of the object becomes much simpler to define and easy to control. With respect to the inflatable device upper surface in Fig. 5, here we choose to parameterize the surface in the

Fig. 5 Direct Surface Parameterization, applied to the upper surface of the inflatable device: ξ is defined along the width of the wing, η along the height. A 5 × 5 grid of control points of a surface Bezier parameterization is defined in the mapped domain (ξ,η), and shown in the 3D original space



mapped domain (ξ,η), with a 5 × 5 grid of equi-spaced control nodes (in magenta), defining a Bezier surface. The new deformed surface can be written as:

$$\mathbf{S}^{**} = \mathbf{S} + w\,(\xi, \eta) \sum_{i=1}^{5} \sum_{j=1}^{5} \mathbf{P}_{i,j} B^{i,5}(\xi) B^{j,5}(\eta) \tag{3}$$

where $\mathbf{P}_{i,j}$ is the displacement, in the original 3D space, of the control node (i, j) of the grid and $B^{k,K}\,(\cdot)$ are the Bernstein polynomials defined along the mapped coordinates ξ and η. Each $\mathbf{P}_{i,j}$ defines 3 degrees of freedom of the parameterization, for a total of 75. In this case the weight function $w\,(\xi, \eta)$, which imposes the continuity condition at the boundaries, can be calculated explicitly in the (ξ, η) space.

In order to create a realistic inflation of the surface and to reduce the number of design parameters, the following assumptions were adopted:

1. nodal displacements $\mathbf{P}_{i,j}$ are allowed to move only along the local normal $\mathbf{N}_{i,j}$ of $\mathbf{S}$, where the control node is positioned;
2. nodal displacements variation is linear, along the η coordinate;
3. only displacements of control nodes in $i = 2, 3, 4$ are relevant for the deformation, leaving those on the borders Γ_1 and Γ_3 unmoved;
4. the weight function imposes G_0 continuity condition on Γ_1 and Γ_3 and free conditions on Γ_2 and Γ_4.

Putting all into practice, Eq. (3) becomes:

$$\begin{aligned} \mathbf{S}^{**} &= \mathbf{S} + w\,(\xi, \eta) \sum_{i=2}^{4} \sum_{j=1}^{5} \mathbf{P}_{i,j} B^{i,5}(\xi) B^{j,5}(\eta) \\ \mathbf{P}_{i,j} &= \alpha_{i,j} \mathbf{N}_{i,j}(\mathbf{S}) \\ \alpha_{i,j} &= \alpha_{i,0} + (C_1 - \alpha_{i,0})\eta_{i,j} \qquad\qquad 0 \le \eta \le 1 \end{aligned} \tag{4}$$

where C_1 is a constant and $\eta_{i,j}$ is the η coordinate value of the control node (i, j). Under this assumption, the total number of independent design parameters decrease to 3. The lower part of the inflatable surface is forced to deform symmetrically to the upper part.

2.3 CAMILO

The above features have been implemented in a software package called CAMILO [6] (Computer Aided ManIpulation by Level-set for Optimization), which is a Text Unit interface tool / C++ library to manipulate 3D object shapes, by means of FFD techniques.

3 POD Based Reduced-Order Modeling

We employ a hybrid low-order/high-order method based on domain decomposition, first presented by Buffoni et al. [7], that we generalized for unstructured grids. The main idea is to split the domain of interest in two subdomains and to use different approximation methods in each of the regions. Where it is possible we use a POD based ROM and where it is necessary we use the canonical CFD solver. In order to reduce efficiently the overall cost (cpu time and memory) of this hybrid method, we aim at decreasing the extent of the domain where we have to solve the high-order model (e.g. steady incompressible RANS equations for 3D flows) since the cost of the ROM and the overhead due to interfacing the two models can be considered negligible.

POD based ROMs suffer generally from two aspects. The first issue concerns the fact that the POD basis space is a linear combination of the solution space spanned by the empirical observations. Consequently, if POD is employed in order to predict a certain solution, it is required that this particular solution may be expressed as a linear combination of the previous solutions. This requirement is generally violated when simulating non-linear phenomena. The second issue derives from the fact that the POD basis vectors are defined in physical space. During shape optimization the domain boundaries change and so does the solution space. Hence it is necessary to define the reduced basis in a reference domain and to map each solution to this reference frame. This transformation is generally not available or very difficult to calculate a posteriori. An alternative approach consists in applying POD in a discrete vector space (degrees of freedom of the simulation). Also this approach is limited to very special applications where a one-to-one correspondence between two different computational grids has been assured.

In order to overcome the above mentioned difficulties we employ the canonical CFD solver within a crucial region where linear effects may be present and where geometry variations occur. Often these two criteria indicate the same zone.

3.1 Proper Orthogonal Decomposition

The POD [9] consists in an optimal representation of the solution space via a finite number M of shape modes $\phi^{(i)}(\mathbf{x})$. In our application, we compute the POD basis a priori, starting from an appropriate database of N flow snapshots $U^{(1\cdots N)}(\mathbf{x})$:

$$\phi^{(i)}(\mathbf{x}) = \sum_{k=1}^{N} b_{ik} U^{(k)}(\mathbf{x}) \tag{5}$$

with $1 \leq i \leq M$. In doing so, the solution can be represented by a small number of unknowns, that are the coefficients b_{ik} of the Galerkin expansion. The decomposition is performed individually for each primitive variable, i.e. pressure, flow velocity and

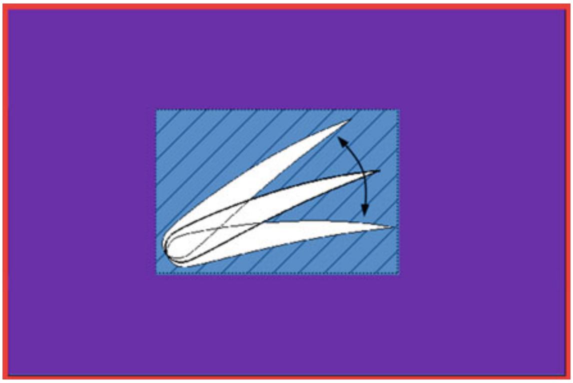


Fig. 6 Decomposed domain. The overlapping region is denoted by the color purple, whereas the blue region represents the reduced domain and the red regions identifies where the POD basis is evaluated

turbulent quantities, solving the $N \times N$ eigen-problem associated with the snapshots correlation matrix. Since we have to deal with unstructured meshes, we define a weighted inner product $\langle \cdot, \cdot \rangle_W$ using the cell volumes.

Alternatively, it is possible to compute the basis through the singular value decomposition (SVD) of the snapshots matrix $S = U \Sigma V^T$: this approach allows to efficiently update the POD basis during the optimization loop itself, as soon as a new snapshot evaluated by the high-fidelity model becomes available, e.g. using one of the methods suggested by Zimmermann [8] in his analysis of state-of-the-art algorithms.

3.2 Hybrid ROM Based on Domain Decomposition

The non-linear CFD model and the ROM are coupled through an overlapping region via a least-square problem which minimizes the L2 norm of the distance between the solutions, following the steps below (see Fig. 6 for reference):

1. integrate the governing equations in the reduced domain by the CFD solver in order to obtain U_{cfd}^{n+1} (blue and purple);
2. project the restriction to the overlapping region of the updated solution U_{cfd}^{n+1} on the subspace spanned by the POD modes Φ_i, and determine U_{pod}^{n+1} (purple);
3. recover the boundary conditions to be imposed at the next solver iteration as the trace of U_{pod}^{n+1} on the boundary (red);
4. go to 1. until convergence.

3.3 Automatic Detection of the HFM/ROM Interface

The principal parameter of the above presented algorithm is the interface which separates the ROM from the canonical CFD solver. We propose a strategy in order to identify in an automatic way the zones where the POD fails in representing non-linear effects.

3.3.1 By Using the Snapshot Database

The algorithm is based on a leave-one-out strategy. Within this strategy different POD bases are computed by leaving one snapshot out.

This snapshot is projected on the computed basis and the representation error map related to ith snapshot is calculated.

$$e^{(i)}(\mathbf{x}) = \frac{1}{U^{ref}}\left[U^{(i)}(\mathbf{x}) - \sum_{k=1}^{N}\langle U^{(i)}(\mathbf{x}), \phi^{(k)}(\mathbf{x})\rangle\phi^{(k)}(\mathbf{x})\right] \tag{6}$$

Let us denote with

$$\omega^{(i)} := \forall\mathbf{x}|e^{(i)}(\mathbf{x}) > tol \tag{7}$$

all points where the error map relative to one snapshot exceeds a user-defined tolerance. We define the region where to use the CFD solver by calculating the union of these domains:

$$\Omega_{CFD} = \bigcup_{i=1}^{N}\omega^{(i)} \tag{8}$$

This strategy uses the snapshots of the database in order to built the error indicator. Furthermore, the error measures the linear dependency between the different snapshots. Consequently, if the parameter space has not been sampled in an exhaustive manner, the method may fail and indicate small reconstruction errors. In other words, the database **must** contain significant information of non-linear effects all over the parameter range of interest.

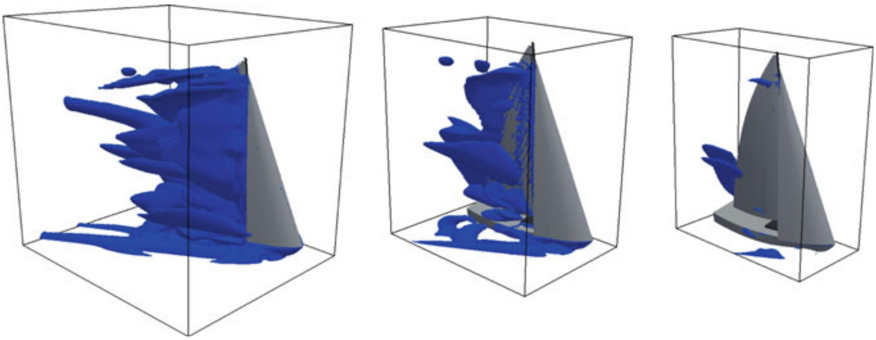

(a) $tol = 0.1$ (b) $tol = 0.2$ (c) $tol = 0.4$

Fig. 7 Ω_{CFD} for three different tolerance values. The resulting error maps envelope is represented by the blue isosurfaces, whereas the corresponding computational domain used in the hybrid runs is delimited by the black lines

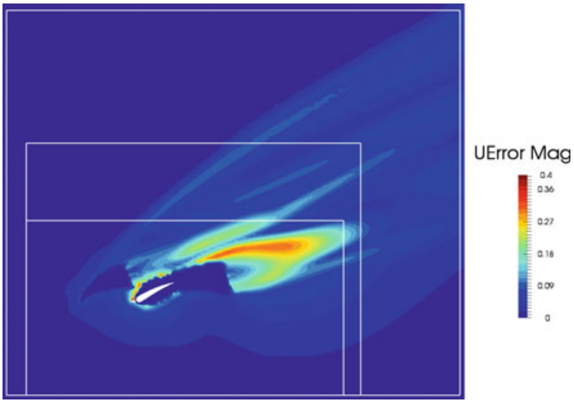


Fig. 8 $e^i(\mathbf{x})$ envelope of velocity using as U^{ref} the magnitude of velocity at the top of the mast: section at $z = 4$ m. The white lines delimit the reduced domains (a), (b) and (c). Close to the sails the error goes to zero since this region is excluded from the POD

Table 1 Performance of the hybrid scheme for reduced domain (a), (b) and (c). ϵ_T represents the relative error on the objective function, i.e. the sailing system thrust. It should be noted that the overall speed-up factor is given not only by the ratio between the number of cells in the computational domain, but it is enhanced by a faster convergence of the hybrid-ROM: the average number of iterations required to obtain convergence on the forces is at least three times smaller than the full-order case, thanks to a better flow initialization

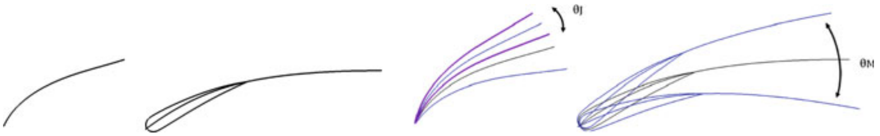
Domain	tol	ε_T	#cells [MIL]	iters	speed-up
full			25.7	3000	1
(a)	0.1	0.0066	5.3	1000	15
(b)	0.2	0.0093	4.8	1000	17
(c)	0.4	0.0099	4.1	1000	20

Figure 7 and Fig. 8 show respectively the extent of the Ω_{CFD} region for three different tolerance choices and a section of the global error map of the velocity field. Table 1, instead, summarizes the performance of the hybrid scheme for the corresponding reduced domains with respect to the full-order model.

4 Sail Optimization

With reference to Figs. 9 and 10, the following 5 design parameters are chosen to modify wing trimming and device inflating stage:

1. θ_M: is the main rotation, applied to both wing sails, around their respective axes.
2. θ_J: is the relative rotation of the jib around its relative axis.
3. p_1, p_2, p_3: displacement along surface normals of the interior control points of inflatable root contour Γ_2, as explained in Sect. 2.2. Inflating of upper and lower surfaces of the device is considered as symmetric.

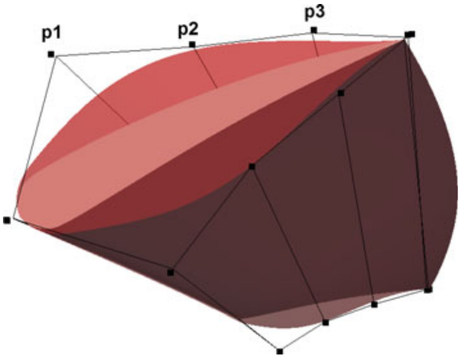


(a) Initial configuration of mainsail and jib

(b) Rotations of mainsail and jib

Fig. 9 Trimming of sails around mast and jib axes. Positive rotations in counterclockwise direction

Fig. 10 Parameterization and degrees of freedom of the inflatable device



Limits of parameters space are reported in Table 2. With the current set of p_i parameters, the maximum thickness on the chord of the inflatable device could vary from a minimum value of 10 cm up to a maximum of 30 cm. The maximum thickness of the baseline configuration is 14 cm.

The optimization process was carried out employing the open-source tool chain DAKOTA-OpenFOAM, with CAMILO handling the shape deformation of the inflatable device and the rigid rotation of the mainsail and the jib. The hybrid-ROM model was directly implemented in the OpenFOAM environment, as a plugin (podFoam). A surrogate-based model (kriging) was used, along with the hybrid-ROM as a high-fidelity model to update the response surface construction. The hybrid-ROM model required the definition of a database of 10 CFD simulations (RANS), on the full boat model to build up the POD modes. Reaching the optimum required 12940 evaluations of the surrogate model and 90 evaluation of the hybrid-ROM, for a total CPU time of computation of 3000 cpuh. The boost factor given by the hybrid-ROM model let us to save mostly the 87% of computational time, with respect to the case of a surrogate model with full CFD simulations as a high-fidelity corrector.

Table 2 Parameters space limits

DoF	θ_M	θ_J	p_1 (mm)	p_2 (mm)	p_3 (mm)
Inf.limit	$-10°$	$-5°$	-10	-5	-5
Sup.limit	10°	5°	100	50	50

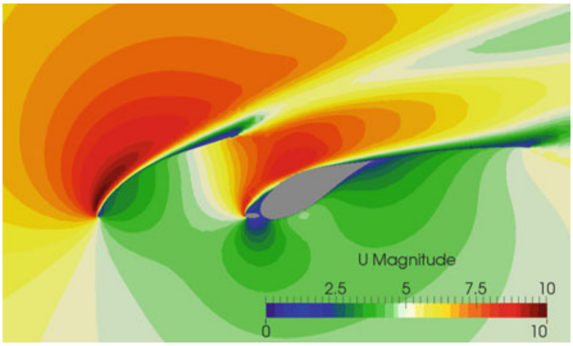


Fig. 11 Velocity magnitude field on a plane cut at z = 4 m, for the optimized wing configuration

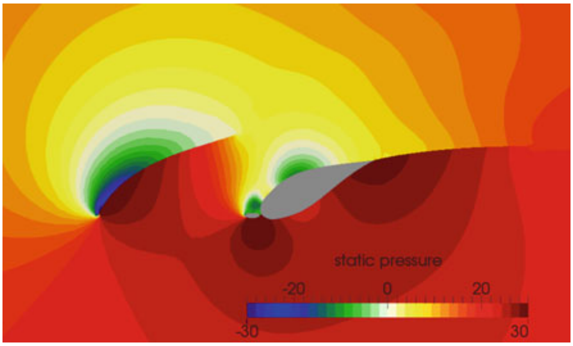


Fig. 12 Static pressure field on a plane cut at $z = 4$ m, for the optimized wing configuration

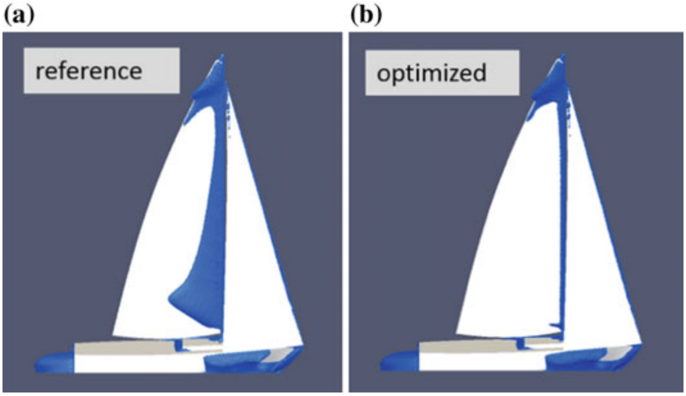


Fig. 13 Controlling flow separation near mast: comparison between reference (**a**) and optimized (**b**) configurations

Results of the optimization are collected in Figs. 11, 12 and 13. The optimum configuration was found for $\theta_M = -0.726°$, $\theta_J = 0.588°$, $p_1 = 93.8$, $p_2 = 17.81$, $p_3 = 24.26$ mm. The optimum thrust was 435.195 N, with a net gain of +3.82% w.r.t. the wing configuration without inflatable device.

5 Conclusions

The paper described an approach for shape optimization of large-scale aerodynamic problems. We focused on two aspects of the optimization process, that are crucial in any CFD industrial application: the geometry parameterization and the development of feasible Reduced-Order Models to speed up large CFD computations in optimization loops. In the first place, we presented an extension of the classical Free-Form Deformation techniques to parameterize complex geometries and enable local deformations on them, handling with geometry continuity constraints issues. Then, we developed a hybrid model, combining a POD-based reduced order model and an industrial CFD solver using a domain-decomposition approach. We discussed the overall accuracy of the model against different choices of the initial decomposed domain size. Its main advantages relies on the possibility to re-use and recycle avalaible data for CFD simulations, as well as in significant computational cost savings without too much loss of overall accuracy in CFD final results. In the end, the above tools were employed and tested, performing a global surrogate based optimization of the mainsail thrust of a sailing boat.

Acknowledgements The authors would like to thank SCS SupercomputingSolutions and CINECA SCAI for providing the necessary HPC resources for this project.

References

1. Sederberg, T.W., Parry, S.R.: Free-form Deformation of solid geometric models. SIGGRAPH Comput. Graph. (ACM) **20**(4), 151–160 (1986)
2. Coquillart, S.: Extended free-form Deformation: a sculpturing tool for 3D geometric modeling. SIGGRAPH Comput. Graph. (ACM) **24**(4), 187–196 (1990)
3. Désidéri, T.A., El Majd, B.A., Janka, A.: Nested and self-adaptive Bézier parameterization for shape optimization. J. Comput. Phys. **224**(1), 117–131 (2007)
4. Sethian, J.: Level Set Methods and Fast Marching Methods: evolving interfaces in Computational geometry, Fluid Mechanics, Computer Vision and Material Sciences. 2nd edn. Cambridge University Press (1999)
5. Crane, K., Weischdel, C., Wardetzky, M.: Geodesics in Heat: a new approach to computing distance based on heat flow. ACM Trans. Graph. **32**(5) (2013)
6. CAMILO: Computer Aided ManIpulation by Level-set for Optimization. http://www.optimad.it/products/camilo
7. Buffoni, M., Telib, H., Iollo, A.: Iterative methods for model reduction by domain decomposition. Comput. Fluids **38**(6), 1160–1167 (2009)

8. Zimmermann, R.: A Comprehensive Comparison of Various Algorithms for Efficiently Updating Singular Value Decomposition Based Reduced Order Models. DLR-Interner Bericht (2011)
9. Lumley, J.L.: The structure of inhomogeneous turbulent flows. In: Yaglow, A.M., Tatarski, V.L. (eds.) Atmospheric Turbulence and Radio wave Propagation, Moscow (1967)

Application of Surrogate-Based Optimization Techniques to Aerodynamic Design Cases

Emiliano Iuliano and Domenico Quagliarella

Abstract The paper proposes the application of evolutionary-based optimization coupled with physics-based and adaptively-trained surrogate model to the solution of both two- and three-dimensional aerodynamic optimization problems. The shape parameterization approach consists of the Class-Shape Transformation (CST) method with a sufficient degree of Bernstein polynomials to cover a wide range of shapes. The in-house ZEN flow solver is used for RANS aerodynamic solution. Results show that, thanks to the combined usage of surrogate models and smart training, optimal candidates may be located in the design space even with limited computational resources with respect to standard global optimization approaches.

1 Introduction

In the context of modern and innovative air vehicle design, the development and assessment of new theoretical methodologies represents a cornerstone for reducing the experimental load, exploring trade-offs and proposing alternatives along the design path. The fidelity of such methods is essential to reproduce "real-life" phenomena with a significant degree of accuracy and to take them into account since the very beginning of the design process. However, due to the high computational effort of high-fidelity methods, a big issue rises when hundreds or thousands of analysis evaluations, like in parametric or optimization studies, have to be performed. In order to speed up the computation while keeping a high level of fidelity, the scientific community is increasingly focusing on surrogate methodologies like meta-models, multi-fidelity models or reduced order models, which can provide a compact, accurate and computationally efficient representation of the aircraft design performance.

E. Iuliano (✉) · D. Quagliarella
Fluid Mechanics Department, Multidisciplinary Analysis and Optimization Group,
CIRA—Italian Aerospace Research Center, Via Maiorise, 81043 Capua, Italy
e-mail: e.iuliano@cira.it

D. Quagliarella
e-mail: d.quagliarella@cira.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_5

Nevertheless, the usage of such models is not straightforward as the amount and quality of information the user has to provide in the learning phase is not known a priori; furthermore, the efficient exploitation of learning data may be hampered by the inherent complexity of the design problem, e.g. non-linearities in the physical model, constraints handling, curse of dimensionality, multi-modal fitness landscape, accuracy *versus* computational effort trade-off. Hence, no general rule exists on the optimal choice of the type of surrogate model, the training and validation strategy, the combination of surrogate model and optimization algorithm.

In order to reduce the computational effort in training accurate surrogate models for aerodynamic shape design problems, this paper proposes the adoption of generic (i.e. not specific to a single surrogate model) in-fill criteria to adaptively and intelligently drive the training process. The adaptive criteria are formulated by explicitly taking into account the goal function with the help of auxiliary functions which have to be maximized. The aim is to find new "optimal" design space points which, once added to the training dataset, provide a "better" surrogate approximation for the optimization purpose. Two surrogate models will be investigated, namely a Kriging model and a Proper Orthogonal Decomposition (POD) model coupled with Radial Basis Function Networks for global interpolation of the modal coefficients. Moreover, two choices of the in-fill criteria are presented in the paper and compared to already published adaptive sampling techniques, like Expected Improvement maximization for Kriging and in-fill criteria for POD model machinery. Two aerodynamic optimization case studies are proposed to test different combinations of surrogate models and adaptive sampling approaches once fixed the computational budget in terms of number of high-fidelity simulations (i.e., CFD analyses). This allows to measure the performances of the presented strategies in a real-world environment and to draw some conclusions about the suitability of in-fill criteria to a specific surrogate model for such a class of problems.

2 Literature Review

Jones et al. [11], among the first, proposed a response surface methodology based on modelling the objective and constraint functions with stochastic processes (Kriging). The so-called Design and Analysis of Computed Experiments (DACE) stochastic process model was built as a sum of regression terms and normally distributed error terms. The main conceptual assumption was that the lack of fit associated only to the regression terms can be considered as entirely due to modelling error, not measurement error or noise, because the training data are derived from a deterministic simulation. Hence, by assuming that the errors at different points in the design space are not independent and the correlation between them is related to the distance between the computed points, the authors came up with an interpolating surrogate model able to provide not only the prediction of objectives/constraints at a desired sample point, but also an estimation of the approximation error. After the construction of such a surrogate model, this last powerful property is exploited to build an

Efficient Global Optimization (EGO), which can be considered as the progenitor of a long and still in development chain of surrogate-based optimization (SBO) methods. Indeed, they found a proper balancing between the need to exploit the approximation surface (by sampling where it is minimized) with the need to improve the approximation (by sampling where prediction error may be high). This was done by introducing the Expected Improvement (EI) concept, already proposed by Schonlau et al. [14], that is an auxiliary function to be maximized instead of the original objective. Sampling at a point where this auxiliary function is maximized improves both the local (exploitation) and global (exploration) search.

An overview of SBO techniques was presented also by Queipo et al. [13] and Simpson et al. [15]. They covered some of the most popular methods in design space sampling, surrogate model construction, model selection and validation, sensitivity analysis, and surrogate-based optimization. Forrester and Keane [4] recently proposed a review of some advances in surrogate-based optimization. An important lesson learned is that only calling the true function can confirm the results coming from the surrogate model. Indeed, the path towards the global optimum is made of iterative steps where, even exploiting some surrogate model, only the best results coming from the true function evaluations are taken as optimal or sub−optimal design. The true function evaluation has to be also invoked to improve the surrogate model. With the term "in-fill criteria" it is usually meant some principles which allow to intelligently place new points (in-fill points) at which the true function should be called. The selection of in-fill points, also referred to as adaptive sampling or model updating, represent the core of a surrogate-based optimization method and helps to improve the surrogate prediction in promising areas of the objective space.

The right choice of the number of points which the initial sampling plan would comprise and the ratio between initial/in-fill points has been the focus of several recent studies. However, it must be underlined that no universal rules exist, as each choice should be carefully evaluated according to the design problem (e.g., number of variables, computational budget, type of surrogate). Forrester and Keane assumed that there is a maximum budget of function evaluations, so as to define the number of points as a fraction of this budget. They identified three main cases according to the aim of the surrogate construction: pure visualization and design space comprehension, model exploitation and balanced exploration/exploitation. In the first case, the sampling plan should contain all of budgeted points as no further refinement of the model is foreseen. In the exploitation case, the surrogate can be used as the basis for an in-fill criterion, that means some computational budget must be saved for adding points to improve the model. They also proposed to reserve less than one half points to the exploitation phase as a small amount of surrogate enhancement is possible during the in-fill process. In the third case, that is two-stage balanced exploitation/exploration in-fill criterion, as also shown by Sóbester [16], they suggested to employ one third of the points in the initial sample while saving the remaining for the in-fill stage. Indeed, such balanced methods rely less on the initial prediction and so fewer points are required. Concerning the choice of the surrogate, the authors observed that it should depend on the problem size, i.e. the dimensionality of the design space, the expected complexity, the cost of the true analyses and the in-fill strategy to be adopted. However, for a given problem, there is not a general rule. The proper choice could come up past various model selection and validation criteria. The accuracy of a number of surrogates could be compared by assessing their ability to predict a validation data set. Therefore, part of the true computed data should be used for validation purposes only and not for model training. This approach can be infeasible when the true evaluations is computationally expensive.

Forrester also underlined that some in-fill criteria and certain surrogate models are somewhat intimately connected. For a surrogate model to be considered suitable for a give in-fill criterion, the mathematical machinery of the surrogate should exhibit the capability to adapt to unexpected, local non-linear behavior of the true function to be mimicked. From this point of view, polynomials can be immediately excluded since a very high order would be required to match this capability, implying a high number of sampling points. In general, a global search would require a surrogate model able to provide an estimate of the error it commits when predicting. Thus, the authors suggested to use Gaussian process based methods like Kriging, although citing the work of Gutmann et al. [5] as an example of one−stage goal seeking approach employing various radial basis functions. Finally, some interesting suitable convergence criterion to stop the surrogate in-fill process were proposed. In an exploitation case, i.e. when minimizing the surrogate prediction, one can rather obviously choose to stop when no further significant improvement is detected. On the other hand, when an exploration method is employed, one is interested in obtaining a satisfying prediction everywhere, so that he can decide to stop the in-fill process when some generalization error metrics, e.g. cross−validation, falls below a certain threshold. When using the probability or expectation of improvement, a natural choice is to consider the algorithm converged when the probability is very low or the expected improvement drops below a percentage of the range of observed objective function values. However, the authors also observed that discussing on convergence criterion may be interesting and fruitful, but "in many real engineering problems we actually stop when we run out of available time or resources, dictated by design cycle scheduling or costs". This is what typically happens in aerodynamic design, where the high-dimensionality of the design space and expensive computer simulations often do not allow to reach the global optimum of the design problem but suggest to consider even a premature, sub-optimal solution as a converged point.

3 Surrogate Model

The surrogate model consists of the Proper Orthogonal Decomposition (POD) of known CFD flow fields coupled with Radial Basis Function (RBF) Networks to realize the pseudo-continuous representation throughout the design space. The Singular Value Decomposition (SVD) solution of the POD basis vectors and coefficients for steady-state problems is described in references [6, 8–10]. This approach is normally preferred to the eigenvalue/eigenvector solution as it is faster and easier to implement. The discussion will unfold with specific reference to compressible aerodynamic problems, hence the space domain will be the discretized volume occupied by the flowing air and the snapshot vectors will be defined from computed flow fields.

The POD/RBF surrogate models is built from the vectors $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_M$ representing the CFD flow fields and obtained by expensive simulations at representative set of design sites $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$. Finding a Proper Orthogonal Decomposition means to compute a linear basis of vectors to express any other $\mathbf{s}_j \in \mathbb{R}^N$ with the condition that this basis is optimal in some sense. To compute the optimal basis, we first define the snapshot deviation matrix

$$\mathbf{P} = \left(\mathbf{s}_1 - \bar{\mathbf{s}}\ \mathbf{s}_2 - \bar{\mathbf{s}} \cdots \mathbf{s}_\mathrm{M} - \bar{\mathbf{s}}\right)$$

where the ensemble mean vector is computed as

$$\bar{\mathbf{s}} = \frac{1}{M}\sum_{j=1}^{M}\mathbf{s}_j$$

The POD decomposition is obtained by taking the singular value decomposition (SVD) of $\mathbf{P}$

$$\mathbf{P} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \mathbf{U}\begin{pmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_M \\ 0 & \cdots & 0 \end{pmatrix}\mathbf{V}^T \tag{1}$$

with $\mathbf{U} \in \mathbb{R}^{N\times N}$, $\mathbf{V} \in \mathbb{R}^{M\times M}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times M}$ and the singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_M \geq 0$. The POD basis vectors, also called POD modes, are the first M column vectors of the matrix $\mathbf{U}$, while the POD coefficients $\alpha_i(\mathbf{x}_j)$ are obtained by projecting the snapshots onto the POD modes:

$$\alpha_i(\mathbf{x}_j) = (\mathbf{s}_j - \bar{\mathbf{s}}, \boldsymbol{\phi}_i) \tag{2}$$

If a fluid dynamics problem is approximated with a suitable number of snapshots from which a rich set of basis vectors is available, the singular values become small rapidly and a limited number of basis vectors are adequate to reconstruct and approximate the snapshots as they preserve the most significant ensemble energy contribution. In this way, POD provides an efficient mean of capturing the dominant features of a multi-degree of freedom system and representing it to the desired precision by using the relevant set of modes. The reduced order model is derived by projecting the CFD model onto a reduced space spanned by only some of the proper orthogonal modes or POD eigenfunctions. This process realizes a kind of lossy data compression through the following approximation

$$\mathbf{s}_j \simeq \bar{\mathbf{s}} + \sum_{i=1}^{\hat{M}}\alpha_i(\mathbf{x}_j)\boldsymbol{\phi}_i \tag{3}$$

where

$$\hat{M} \leq M \implies \frac{\sum_{i=1}^{\hat{M}} \sigma_i^2}{\sum_{i=1}^{M} \sigma_i^2} \geq \varepsilon \tag{4}$$

and ε is a pre-defined energy level. In fact, the truncated singular values fulfils the relation

$$\sum_{i=\hat{M}+1}^{M} \sigma_i^2 = \varepsilon_{\hat{M}}$$

If the energy threshold is high, say over 99% of the total energy, then $\hat{M}$ modes are adequate to capture the principal features and approximately reconstruct the dataset. Thus, a reduced subspace is formed which is only spanned by $\hat{M}$ modes.

3.1 Pseudo-continuous Global Representation

Equation 3 allows to get a POD approximation of any snapshot $\mathbf{s}_j$ belonging to the ensemble set. Indeed, the model does not provide an approximation of the state vector at design sites which are not included in the original training dataset. In other words, the POD model by itself does not have a global predictive feature, i.e. over the whole design space. As the aim is to exactly reproduce the sample data used for training and to consistently catch the local data trends, a Radial Basis Function (RBF) network answers to these criteria and has been chosen for POD coefficients interpolation. Gaussian, multi-quadric and inverse quadratic functions are used. The RBF parameters are found by imposing the interpolation condition on the training set for any modal coefficient $i \leq \hat{M}$.

The RBF width parameters have a big influence both on the accuracy of the RBF model and on the conditioning of the solution matrix. In particular, it has been found (Refs. [3, 5]) that interpolation errors become high for very small and very large values of the width parameter θ, while the condition number of the coefficient matrix increases with increasing values of θ. Therefore, they have to be “optimal” in the sense that a tuning of the width parameters is needed to find the right trade-off between interpolation errors and solution stability (Ref. [3] for a discussion about how to properly select the best set of parameters). The pseudo-continuous prediction of the flow field at a generic design site w is then expressed as:

$$\mathbf{s}(\mathbf{x}) = \bar{\mathbf{s}} + \sum_{i=1}^{\hat{M}} \alpha_i(\mathbf{x}) \boldsymbol{\phi}_i \tag{5}$$

This provides a useful surrogate model which combines design of experiments for sampling, CFD for training, POD for model reduction and RBF network for global approximation. In conclusion, an explicit, global, low-order and physics-based model linking the design vector and the state vector has been derived and will be used as surrogate model.

4 Surrogate Model Sequential In-fill

The strategy for training a surrogate model is referred to as the process of selecting a proper set $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M$ on which the surrogate model is built. The training strategy is heavily dependent on the type and scope of the surrogate model and, in principle, has to be tailored on it. Indeed, the addition of new samples, if not completely random, must follow some specific criteria that may be very different depending on the purpose of the training process. For instance, Latin Hypercube Sampling (LHS) has been designed to satisfy space-filling requirements and obtain a good coverage of the design space. Here, the emphasis is given on sampling strategies which are able to "adapt" to the response within an optimization process: in particular, they can aim at improving the quality of the model prediction (error-driven strategies) or minimizing the objective function (objective-driven strategies). Most of the adaptive sampling approaches pursue the exploration/exploitation trade-off, where exploration means sampling away from available data, where the prediction error is supposedly higher, while exploitation means trusting the model prediction, thus sampling where the surrogate provides global minima. It is clear that a trade-off between the two behaviors is needed: indeed, exploration is useful for global searching, but it may lead to unveil uninteresting regions of the design space; on the other hand, exploitation helps to improve the local accuracy around the predicted optima, but it may result in local minima entrapment. Figure 1 provides a simple example of adding a new training point by using respectively exploitation, exploration and balanced approaches. Given a set of training points (black circle points) evaluated on the true function (solid black line), a surrogate model (dashed black line) is built: if a new sample has to be added, a pure exploitation approach would place it where the global minimum of the surrogate is detected, i.e. very close to one of the training point (triangle point); a pure exploration approach, instead, would lead to sample where the maximum uncertainty in the model prediction is found, i.e. far from available training points (circle point); a balanced exploration/exploitation approach combines the two aspects, thus providing a new sample which significantly improves the surrogate prediction (square point).

Here, we are interested in designing balanced in-fill criteria for a generic surrogate model. Such criteria are formulated in terms of an auxiliary function referred to as v function hereinafter and called potential of improvement. Given the generic location in the design space $\mathbf{x}$, the objective function $f(\mathbf{x})$ to minimize, a set of n available sampling points $\{X_n\}$ and the corresponding set of true objective function values $\{F_{X_n}\}$

$$X_n = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \qquad F_{X_n} = \{f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)\}$$

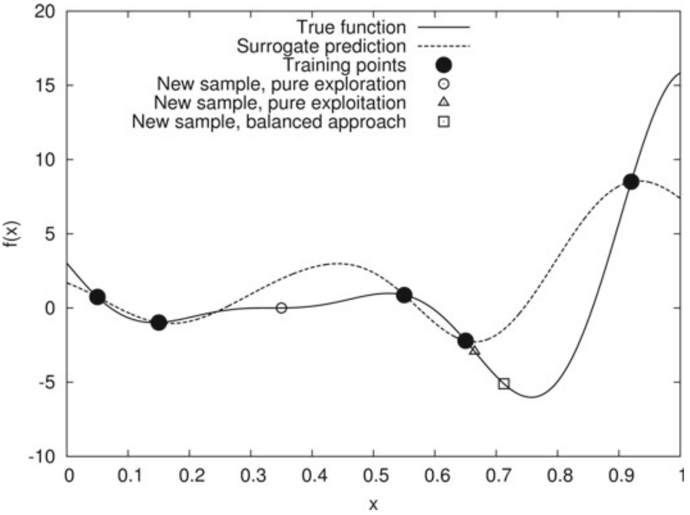


Fig. 1 Exploitation versus exploration, 1D example

the update of the surrogate model can be realized by finding a new sample $\mathbf{x}_{n+1}$ which maximizes the potential of improvement:

$$\mathbf{x}_{n+1} = \underset{\mathbf{x}}{\operatorname{argmax}}\, \upsilon(\mathbf{x}, \hat{f}(\mathbf{x}), X_n, F_{X_n})$$

where $\mathbf{x}$ is the generic design space location and $\hat{f}(\mathbf{x})$ is the surrogate prediction at $\mathbf{x}$.

Hereinafter, the maximization of the auxiliary function is achieved in the following way: a huge Latin Hypercube Sampling dataset (e.g., five hundred times the dimension of the design space) is computed and the values of the auxiliary functions are computed at each point (this requires limited computational effort as the auxiliary function only depends on the surrogate prediction, which is fast to obtain, and on the true objective function values at already collected points); hence, the new sample is located where the maximum value of the auxiliary function is met. In order to avoid the duplication of the updating samples when iterating the in-fill process, the seed of the Latin Hypercube is changed at each iteration.

As concerns the type and nature of the potential of improvement function, previous investigations [7] showed that error-driven in-fill criteria may lead to intensively explore the design space in order to reduce the prediction error, but, conversely, this resulted in a lack of efficiency of the whole optimization process when fixing the total computational budget. Hence, in the following section the discussion will focus on objective-driven approaches which proved to be more suitable to global optimization. In particular, two criteria will be proposed: the first is based on the factorization of the potential of improvement in order to explicitly realize the trade-off between exploration and exploitation; the second is defined according to the Expected Improvement concept.

4.1 Factorization Criterion

The first proposal of adaptive in-fill criterion aims at combining exploration and exploitation by means of a generic factorization as follows:

$$v(\mathbf{x}, \hat{f}(\mathbf{x}), X_n, F_{X_n}) = g(\mathbf{x}, X_n) h(\hat{f}(\mathbf{x}), F_{X_n}) \tag{6}$$

The functions g and h measure the exploration and model trust contribution respectively. In particular, the exploration function g should estimate how strong is the influence of the set of already collected samples X_n on a generic candidate $\mathbf{x}$. One of the preferred approaches is to make the g function dependent on the Euclidean distance $d(\mathbf{x}, \mathbf{x}_i)$ between the generic design space location $\mathbf{x}$ and the i-th element of the training set X_n:

$$g(\mathbf{x}, X_n) = g(d(\mathbf{x}, \mathbf{x}_1), d(\mathbf{x}, \mathbf{x}_2), \ldots, d(\mathbf{x}, \mathbf{x}_n))$$

On the other hand, the exploitation function h should take into account how the surrogate prediction $\hat{f}$ compares with the available set of true objective function values F_{X_n}. In particular, this contribution should put emphasis on trusting the model prediction, hence the h function should exhibit its maxima in correspondance to the minima of $\hat{f}$.

Of course, different in-fill criteria can be selected by properly designing the functions g and h. In the present context, the following solution is adopted:

$$g(\mathbf{x}, X_n) = \frac{\min_{\mathbf{x}_i \in X_n} d(\mathbf{x}, \mathbf{x}_i)}{\max_{\mathbf{x}_i, \mathbf{x}_j \in X_n} d(\mathbf{x}_i, \mathbf{x}_j)} \tag{7}$$

$$h(\hat{f}(\mathbf{x}), F_{X_n}) = \exp\Big(-\sigma \frac{\hat{f}(\mathbf{x}) - f_{min}}{f_{max} - f_{min}}\Big) \tag{8}$$

where σ is a tuning parameter, $f_{min} = \min\{f_{\mathbf{x}_1}, \ldots, f_{\mathbf{x}_n}\}$ and $f_{max} = \max\{f_{\mathbf{x}_1}, \ldots, f_{\mathbf{x}_n}\}$. This choice of the h function provides two main features:

1. the value of h approaches the unity when $\hat{f}(\mathbf{x})$ approaches f_{min};
2. for $\hat{f}(\mathbf{x}) < f_{min} \longrightarrow h(\hat{f}(\mathbf{x}), F_{X_n}) > 1$;

As a consequence, “bad” candidates (from the surrogate model point of view) will be filtered out, while “good” candidates (i.e., candidates with predicted objective function values lower than the current minimum of the true objective function) will be recognized and rewarded with higher rank. However, if they are too close to samples stored in X_n, they will be penalized by the g function. Hence, a trade-off is realized

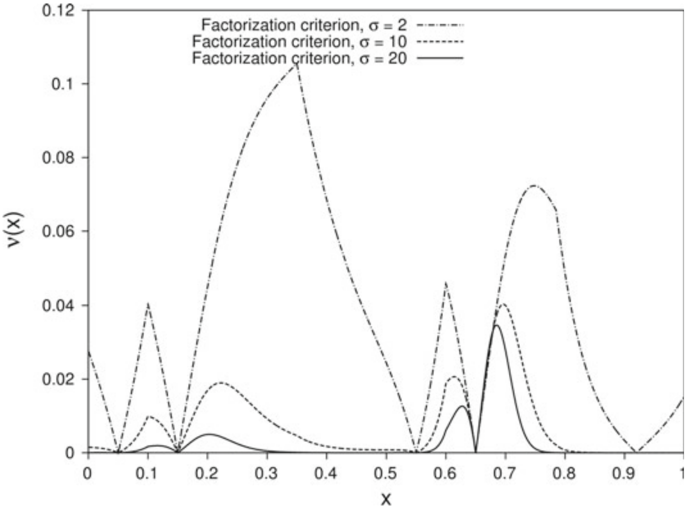


Fig. 2 Potential of improvement based on factorization criterion, 1D example

between surrogate prediction and location in the design space. Figure 2 shows the potential of improvement function for three values of the σ parameter. The case is the one depicted in Fig. 1. The effect of the σ parameter is clearly observable: indeed, for $\sigma = 2$ the peak of the potential of improvement is located around $x \simeq 0.35$, i.e. in a region which is uninteresting to discover the global optimum of the true objective function. This occurs because, for $\sigma = 2$, the filtering power of the h function is relatively small and so it is its weight within the factorization: as a consequence, the g function dominates, the exploration-exploitation trade-off is not realized (the exploration contribution is much higher) and new candidates will be chosen according to their distance from collected samples. The landscape changes for $\sigma = 10$ and $\sigma = 20$, as the levels of v are globally flattened and the peak moves to $x \simeq 0.7$, i.e. very close to the position of the true objective function optimum.

4.2 *Expected Improvement-Like Criterion*

This criterion has been designed trying to mimic the same rationale of the Expected Improvement criterion, usually coupled to a Kriging-based surrogate, as highlighted in Sect. 2. The present approach, named "EI-like" hereinafter, represents a generalization of that method: indeed, for a generic surrogate model, the information about the uncertainty of the surrogate is not available, while a Kriging model, being a Gaussian process, provides an estimate of the prediction variance together with the prediction itself. The potential of improvement is designed to have the same form of the Expected Improvement function, that is:

$$v(\mathbf{x}, \hat{f}(\mathbf{x}), X_n, F_{X_n}) = (f_{min} - \hat{f}(\mathbf{x}))\Phi\Big(\frac{f_{min} - \hat{f}(\mathbf{x})}{\hat{s}(\mathbf{x})}\Big) + \hat{s}(\mathbf{x})\phi\Big(\frac{f_{min} - \hat{f}(\mathbf{x})}{\hat{s}(\mathbf{x})}\Big) \tag{9}$$

where $\hat{s}(\mathbf{x})$ is an estimate of the prediction error and $\Phi(\mathbf{x})$ and $\phi(x)$ are respectively the cumulative distribution and probability density functions of a standard normal distribution. The prediction error is estimated as follows:

$$\hat{s}(\mathbf{x}) = \frac{1}{2}|f_{max} - f_{min}| \exp\Big(-\gamma \frac{\max_{\mathbf{x}_i, \mathbf{x}_j \in X_n} d(\mathbf{x}_i, \mathbf{x}_j)}{\min_{\mathbf{x}_i \in X_n} d(\mathbf{x}, \mathbf{x}_i)}\Big) \tag{10}$$

where γ is a tuning parameter.

The $\hat{s}$ function has been designed in order to quickly increase with increasing distance from an available sample and to have an order of magnitude related to the actual values of the objective function. Figure 3 shows the potential of improvement function for three values of the γ parameter. The case is the one reported in Fig. 1. The γ parameter strongly alters the potential of improvement profile in terms of potential levels and location of the peak. Indeed, for $\gamma = 0.001$ and 0.01, the peak of the potential of improvement is around $x \simeq 0.7$, i.e. in the vicinity of the true minimum, thus providing a balanced prediction for a new sample. For $\gamma = 0.1$, instead, the peak moves to $x \simeq 0.3$ and the balance would significantly shift in favor of pure exploration: in fact, injecting a new sample at $x \simeq 0.3$ would not improve the surrogate prediction of the true objective function in the vicinity of the global optimum at the next iterate.

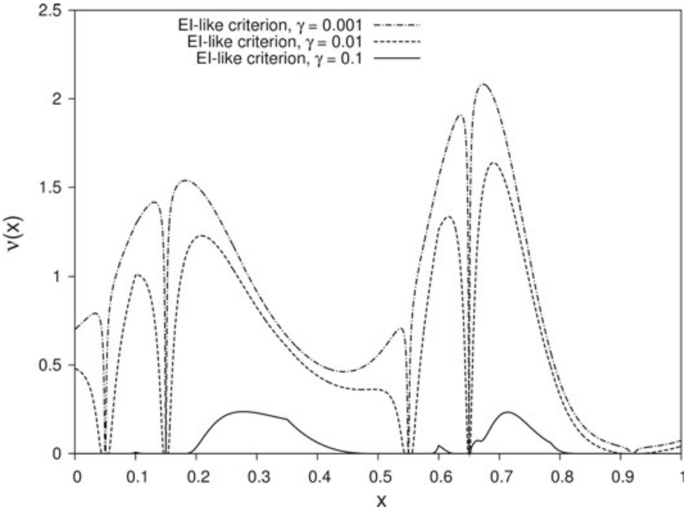


Fig. 3 Potential of improvement based on EI-like criterion, 1D example

5 Surrogate-Based Optimization

The workflow of the surrogate-based shape optimization (SBSO) is depicted in Fig. 4. Basically, it starts with an *a-priori* design of experiment (a Latin Hypercube sampler) whose aim is to initialize the database population: typically, based on literature results and author's experience, the dimension of the initial sampling should not exceed one-third of the total computational budget. A parameterization module transforms the design vectors into geometrical shapes, for each shape a volume mesh is computed by launching an in-house developed automatic mesh generator and a set of CFD computations are executed in parallel with the in-house ZEN CFD flow solver [2]. Once the converged flow field variables are available, the POD/RBF surrogate model is built as described in Sects. 3 and 3.1. After that, the workflow in Fig. 4 shows two internal cycles, namely the sequential in-fill (also called adaptive sampling) and the optimization update. These iterative phases reflect two different needs: first of all, providing an improved and reliable model to the optimizer; then, iterating the optimizer to refine the optimum search.

The first cycle (database updating by in-fill criteria) is based on the techniques described in Sect. 4 and is aimed at improving the surrogate model prior to the optimization phase by providing new design candidates $\mathbf{x}_{new}$ to be added to the ensemble database. The condition to exit from this internal loop is based either on pre-defined levels of improvement or on computational budget considerations.

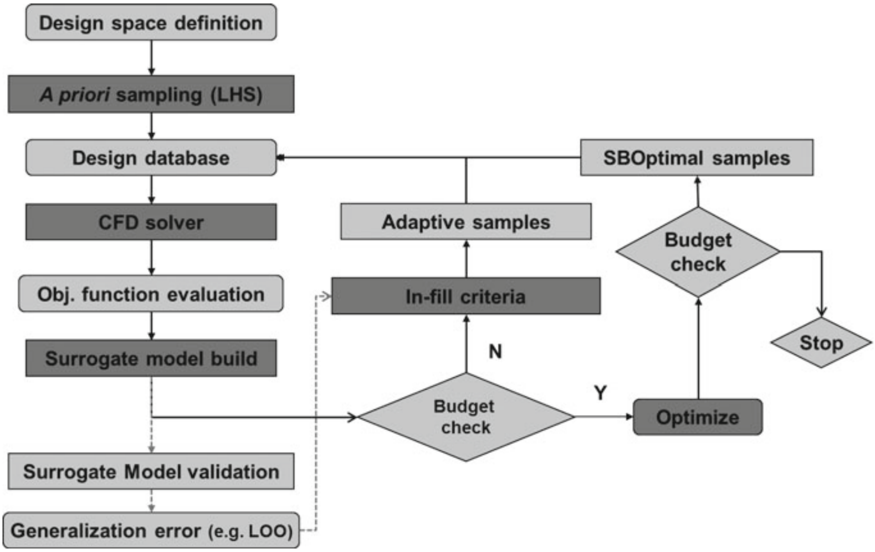


Fig. 4 Workflow of surrogate-based evolutionary optimization: design database updating by adaptive in-fill and surrogate optima addition

The second cycle (database updating by optimization) consists of a series of sequential optimizations where, at i-th iteration, the surrogate model is updated with the high-fidelity evaluation of the optimum candidate found at iteration $(i-1)$-th. This phase allows for including optimal or sub−optimal design sites $\mathbf{x}_{opt}$, provided by the surrogate-based optimization, into the POD ensemble database and should lead to refine the search process in the design space region where the "true" optimum resides. The loop terminates either when the residual of the objective function of the predicted optima falls below a pre-defined threshold or when the computational budget limit has been reached. The optimizer consists of an evolutionary algorithm implemented within the in-house ADGLIB optimization library [17].

6 Application to Aerodynamic Design Cases

In the next sections, two shape optimization cases are considered, i.e. the RAE 2822 airfoil and the isolated wing proposed within the 3rd AIAA Drag Prediction Workshop. Both cases are set in transonic viscous flow conditions. The results of global optimization by means of evolutionary algorithms and CFD evaluation of the objective function will be taken as reference for surrogate-based studies.

6.1 RAE 2822 Airfoil Case

The shape optimization problem is formulated as follows:

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & C_d(\mathbf{x}) \\ \text{subject to} \quad & C_l(\mathbf{x}) = C_{l,base} = 0.824 \\ & C_m(\mathbf{x}) \geq C_{m,base} = -0.092 \\ & A(\mathbf{x}) \geq A_{base} = 0.7787\,\mathrm{m}^2 \end{aligned}$$

where $\mathbf{x}$ is the generic design vector, $A(\mathbf{x})$ is the total area enclosed by the generic airfoil and A_{base} is the corresponding value for the baseline RAE 2822 airfoil. The lift constraint is explicitly satisfied by performing the flow simulation at fixed lift. The pitching moment and the geometric constraint are treated by using a penalization approach. A unit airfoil chord is assumed, the pitching moment is evaluated at the quarter-chord, the Mach number is 0.734 and the Reynolds number is 6.5×10^6.

6.1.1 Parameterization

The CST (Class-Shape Transformation) approach [12] describes an airfoil shape as a function of the surface parameter $\psi = \frac{x}{c}$, where x is the x-coordinate along the chord line and c is the chord length. The y-coordinate are obtained for upper and lower side as follows:

$$\frac{y_u}{c}(\psi) = C_{1.0}^{0.5}(\psi) S_u(\psi) + \psi \frac{\Delta z_{TE}}{2c} \tag{11}$$

$$\frac{y_l}{c}(\psi) = C_{1.0}^{0.5}(\psi) S_l(\psi) - \psi \frac{\Delta z_{TE}}{2c} \tag{12}$$

where $C_{1.0}^{0.5}(\psi)$ is the class function for rounded leading edge and pointed trailing edge airfoils and $S_u(\psi)$ and $S_l(\psi)$ are the so-called shape functions for upper and lower sides:

$$C_{1.0}^{0.5}(\psi) = \psi \sqrt{(1-\psi)} \tag{13}$$

$$S_u(\psi) = \sum_{i=0}^{n} A_{u_i} K_{i,n} \psi^i (1-\psi)^{n-i} \tag{14}$$

$$S_l(\psi) = \sum_{i=0}^{n} A_{l_i} K_{i,n} \psi^i (1-\psi)^{n-i} \tag{15}$$

Shape functions are Bernstein polynomials of order n, Δz_{TE} is the trailing edge thickness, $K_{i,n}$ are binomial coefficients

$$K_{i,n} = \binom{n}{i} = \frac{n!}{i!(n-i)!}$$

and the Bézier coefficients A_{u_i} and A_{l_i} are design weights which can be either defined a-priori in a design optimization process or computed with a least-squares fit to match a specified geometry. The first and last design parameters, i.e. A_{u,l_0} and A_{u,l_n}, are directly linked to well known airfoil shape parameters like leading edge radius R_{le} and trailing edge angle β, being $A_{u,l_0} = \sqrt{\frac{2R_{le}}{c}}$ and $A_{u,l_n} = \tan\beta$. In the present context, 6th-order Bernstein polynomials are considered, hence each airfoil side (upper and lower) is described by 7 design variables for a total number of 14 design variables. The corresponding design weights, which define the RAE 2822 profile according to the chosen parameterization, have been obtained by least-squares fit and are reported in Table 1. The design space bounds are defined by taking ±50% of the baseline design weights.

Table 1 Design weights representing the RAE 2822 airfoil shape with 6th-order Bernstein polynomials

Design weights	$\pm\sqrt{\frac{2R_{le}}{c}}$	A_1	A_2	A_3	A_4	A_5	tan (β)
Upper side	0.12658	0.13786	0.16165	0.16418	0.22047	0.17191	0.21321
Lower side	−0.12892	−0.14264	−0.13111	−0.26147	−0.02525	−0.11451	0.07275

6.1.2 Optimization Studies

Five optimization studies have been performed employing different methods and computational load. Details are reported in Table 2.

Three SBOSA (Surrogate Based Optimization with Sequential Adaptation) runs have been launched, sharing the same setup but exploiting respectively the factorization (SBOSA-FC) and the EI-like (SBOSA-EIL1 and SBOSA-EIL2) criteria. In the latter case, two simulations have been launched with two different values of the EI-like criterion tuning parameter γ in order to explore its effect on the search process. All SBOSA simulations consist of three stages: a a-priori LHS sampling of 42 samples, a model updating stage with adaptive sampling (208 sequential calls of the chosen in-fill criterion) and a surrogate-assisted evolutionary optimization stage consisting of 50 iterative genetic algorithm calls with re-injection of the computed surrogate-based optima in the POD/RBF model database. The total computational budget comprises 300 CFD computations, a rather low effort if considering that the dimension of the design space is 14 and the convergence of classical evolutionary algorithms may require a number of evaluations tens of times higher. No criterion for setting the convergence of the surrogate-assisted optimization is provided here: indeed, the aim is to verify that the attained level of improvement, once run out of the limited computational budget, is significant and possibly “close” to the solution obtained with more expensive algorithms. This would demonstrate that the surrogate-

Table 2 Summary of optimization studies for RAE 2822 case

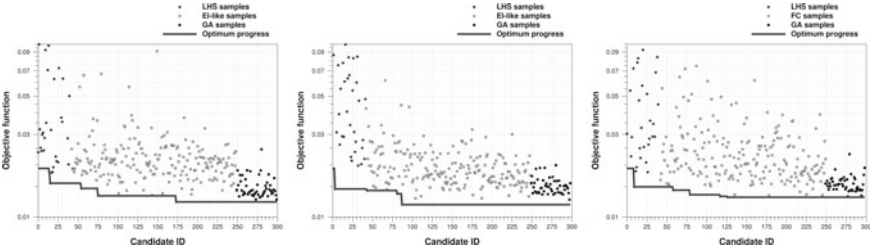
Opt. run ID	SBOSA-EIL1	SBOSA-EIL2	SBOSA-FC	EGO	PGA
Opt. method	EI-like + GA	EI-like + GA	FC + GA	EI	GA
Tuning parameter	$\gamma = 0.001$	$\gamma = 0.005$	$\sigma = 10$	–	–
Obj. func. eval.	POD/RBF	POD/RBF	POD/RBF	Kriging	CFD
Total no. of evaluations	300	300	300	224	6400

assisted search method is considerably efficient in exploring high-dimensional design spaces.

EGO (Efficient Global Optimization) run is performed by using the DAKOTA package [1]. As already stated in Sect. 2, the EGO algorithm searches for the design space location which maximizes the Expected Improvement function and updates the Kriging model database accordingly. The total number of iterations is set to 224.

PGA (Plain Genetic Algorithm) is a pure evolutionary optimization, it is not assisted by any surrogate model and the objective function evaluation is carried out by means of the CFD flow solver (true evaluation of the objective function). A population size of 64 individuals is let evolve for 100 generations with a crossover probability activation of 100% and a mutation rate of 2%.

The true objective function history of the SBOSA runs are reported in Fig. 5. Candidate samples are clearly distinguished according to the criterion used to select them (LHS, EI-like or FC and GA). The solid dark grey line marks the advancement of the minimum value of the true objective function. It is clearly observable how the initial LHS stage (dark grey diamonds) provides only for the initialization of the surrogate, while the objective-driven in-fill points (light grey squares) contribute to progressively drop the objective function levels. In all cases, the overall minimum is not found within the GA-assisted final stage, but rather within the adaptive in-fill phase: this is not surprising as the in-fill criteria are objective-driven, i.e. they rely on the minimization of the surrogate objective function in a balanced exploration-exploitation approach. Figure 6 shows the progress of the true objective function minimum for each of the five optimization studies. The x-axis (progressive number of candidates) is reported in logarithmic scale for the sake of clarity as different scales are involved. It is clearly observable how the surrogate-based optimizations manage to achieve significant improvement with a limited number of CFD evaluations with respect to the plain GA. In particular, Table 3 summarizes the results of each optimization study in terms of minimum objective function value attained and number of effective CFD evaluations needed to capture it. SBOSA provides an interesting compromise solution between the greater performance of PGA at much higher computational cost and the reduced performance of EGO at increased speed (best candidate found with only 81 evaluations).



(a) SBOSA-EIL1 ($\gamma = 0.001$) (b) SBOSA-EIL2 ($\gamma = 0.005$) (c) SBOSA-FC

Fig. 5 Objective function history of SBOSA optimization runs

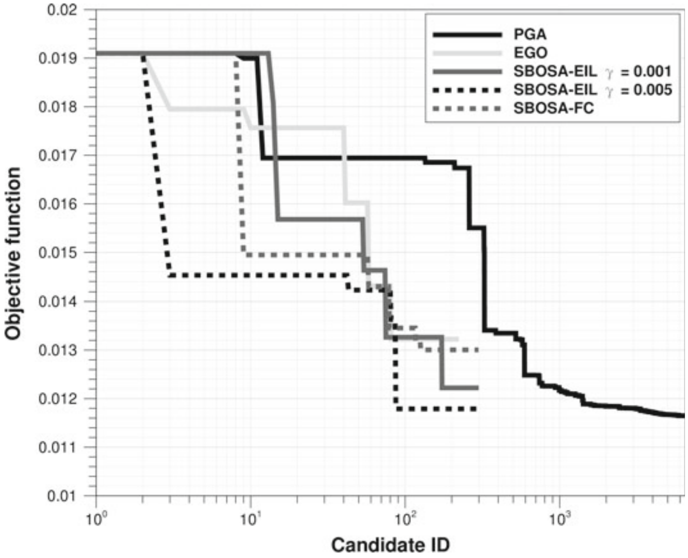


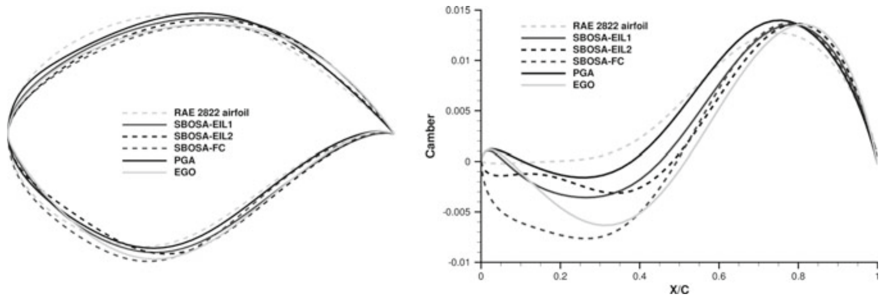
Fig. 6 Progress of minimum values of the objective function for RAE 2822 airfoil optimization studies

Table 3 RAE 2822 optimization, comparison of best candidates

Run	CFD evaluations	Objective value	Improvement
RAE 2822 baseline	–	0.0194	–
PGA best	4200	0.0116	−40.2%
SBOSA-EIL1 best	174	0.0122	−37.1%
SBOSA-EIL2 best	88	0.0118	−39.2%
SBOSA-FC best	126	0.0130	−33.0%
EGO best	81	0.0132	−32.0%

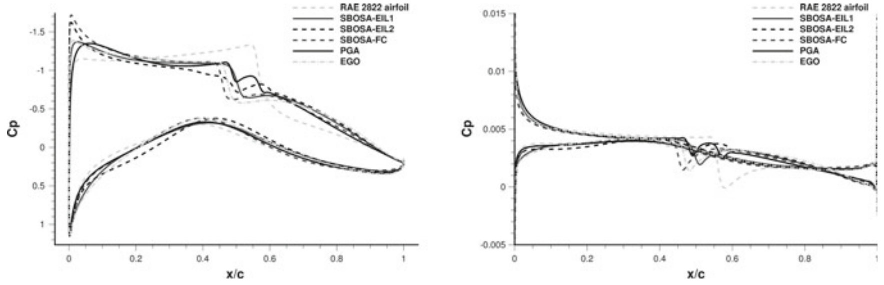
Figure 7a depicts a comparison of the RAE 2822 and the optimized airfoils contour: it can be observed that, for all optimal shapes, there has been a significant reduction of the leading edge radius and an important de-cambering in the fore region (up to 50% airfoil chord). Figure 7b better highlights this feature as the camber distributions of baseline and optimized airfoil shapes are shown. Reducing the leading edge radius is essential to increase the pressure peak and to dampen the shock wave strength, while the negative camber helps to fit within the pitching moment constraint.

Figure 8 shows a comparison of pressure coefficient and skin friction distribution between the RAE 2822 and the optimized airfoils. The shape modification shifted the shock location forward and reduced the local Mach number upstream the shock, thus reducing its intensity and preventing the incipient shock-induced separation which is observed on the RAE 2822 skin friction profile.



(a)Shape (b) Camber distribution

Fig. 7 Geometry comparison between RAE 2822 and optimized airfoils



(a) C_p distribution (b) C_f distribution

Fig. 8 Aerodynamic comparison between RAE 2822 and optimized airfoils

6.2 Drag Prediction Workshop Wing Case

The optimization case is inspired to one of the test cases issued within the 3rd AIAA Drag Prediction Workshop (DPW). The viscous flow around an isolated wing at Mach number of 0.78, Reynolds number of 5 millions and angle of attack (AOA) of 1° is considered. Fully turbulent flow is assumed. The constrained minimization problem is defined as follows:

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & -\frac{C_L(\mathbf{x})}{C_D(\mathbf{x})} \\ \text{subject to} \quad & \max_s \left(\frac{t(\mathbf{x}, s)}{c} \right) = 0.136 \\ & C_L(\mathbf{x}) \geq 0.5 \\ & C_M(\mathbf{x}) \geq -0.05 \end{aligned}$$

where c is the local wing chord value and $t(\mathbf{x}, s)$ is the wing thickness distribution along the chordwise abscissa s for a generic design vector $\mathbf{x}$.

6.2.1 Parameterization

As an extension of the CST airfoil parameterization, a wing shape can be obtained by distributing the airfoil shape function along another surface parameter η, representing the wing span, with the desired law. Once selected N_x, the order of the Bernstein polynomial which represents the wing airfoils, the complete wing shape can be obtained by transforming the coefficients of the Bernstein polynomial into a distribution along the spanwise direction using any appropriate numerical technique. In this way, the entire wing surface is defined through the same basic component airfoils of the root airfoil, but the magnitude of each of them varies across the wing span according to the expansion technique. For example, using the Bernstein polynomial as expansion technique, the spanwise variation of each coefficients A_{u_i} in Eqs. 14 and 15 can be stated in the form

$$A_{u_i} \equiv A_{u_i}(\eta) = \sum_{j=0}^{N_y} B_{u_{i,j}} S_{j,N_y}(\eta) \tag{16}$$

where

$$S_{j,N_y}(\eta) = K_{j,N_y} \eta^j (1-\eta)^{N_y - j}$$

N_y is the order of the expansion Bernstein polynomial and K_{j,N_y} is the usual binomial coefficient.

By inserting Eq. 16 into Eq. 15 and adopting the same technique for the lower surface, the bi-variate Bernstein polynomial shape function for the entire wing is derived

$$S_u(\psi, \eta) = \sum_{i=0}^{N_x} A_{u_i}(\eta) S_{i,N_x}(\psi) = \tag{17}$$
$$\sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \left[B_{u_{i,j}} K_{j,N_y} \eta^j (1-\eta)^{N_y - j} \right] K_{i,N_x} \psi^i (1-\psi)^{N_y - i}$$

$$S_l(\psi, \eta) = \sum_{i=0}^{N_x} A_{l_i}(\eta) S_{i,N_x}(\psi) = \tag{18}$$
$$\sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \left[B_{l_{i,j}} K_{j,N_y} \eta^j (1-\eta)^{N_y - j} \right] K_{i,N_x} \psi^i (1-\psi)^{N_y - i}$$

The wing shape will be then represented by $2 \times (N_x + 1) \times (N_y + 1)$ design parameters, namely $B_{u_{i,j}}$ and $B_{l_{i,j}}$, $i = 0, \ldots, N_x$, $j = 0, \ldots, N_y$. By multiplying the wing shape function by the airfoil-like class function, the overall shape of the wing can be computed. However, a wing is generally and naturally conceived with spanwise distributions for twist angle, dihedral angle, sweep angle and taper ratio:

this additional parameters have to be introduced in the CST model in order to cover realistic wing concepts. The actual wing surface cartesian coordinates can be obtained from the equations

$$x = \psi L(\eta) + x_{LE_{root}} + \int_0^\eta [\sin \Lambda(\eta)] d\eta \tag{19}$$

$$y = \frac{b}{2}\eta \tag{20}$$

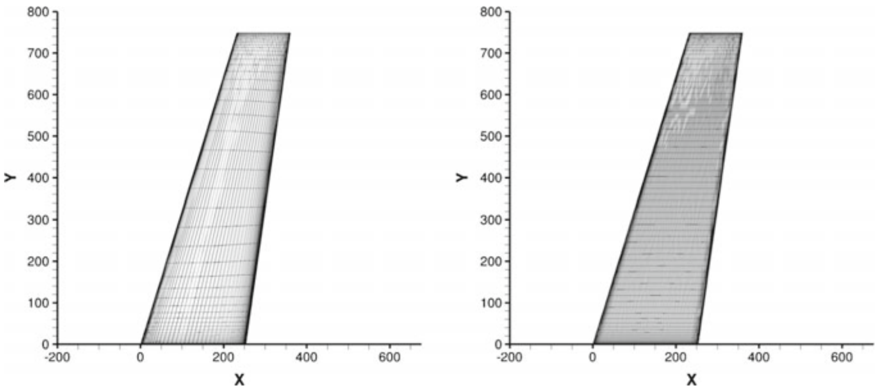
$$\begin{aligned} z_{u,l} = {} & L(\eta)C^{0.5}_{1.0}(\psi)S_{u,l}(\psi, \eta) + L(\eta)\psi[\Delta z_{TE}(\eta) - \tan \alpha_T(\eta)] + \\ & L(\eta)\int_0^\eta [\sin \delta(\eta)] d\eta \end{aligned} \tag{21}$$

where the following spanwise distributions have been introduced: $L(\eta)$ chord length, $\Lambda(\eta)$ sweep angle, $\Delta z_{TE}(\eta)$ trailing edge thickness, $\alpha_T(\eta)$ twist angle, $\delta(\eta)$ dihedral angle. b is the wing span length and $x_{LE_{root}}$ is the x-position of the leading edge of the root airfoil. Several laws can be defined for spanwise distributions, but the standard approach which assures the manufacturing feasibility of the wing shape is to assume constant (e.g. sweep angle), piecewise constant (e.g. dihedral angle), linear (e.g. twist angle) or piecewise linear (e.g. chord length/tapering) variations.

In the present case, one design variable is assigned to control the wing tip twist angle while the twist angle at the root section is kept fixed. A linear twist distribution is adopted along the wing span. The shape design variables are 16, as N_x and N_y are chosen to be respectively 3 and 1. Hence, the total number of design variables is 17.

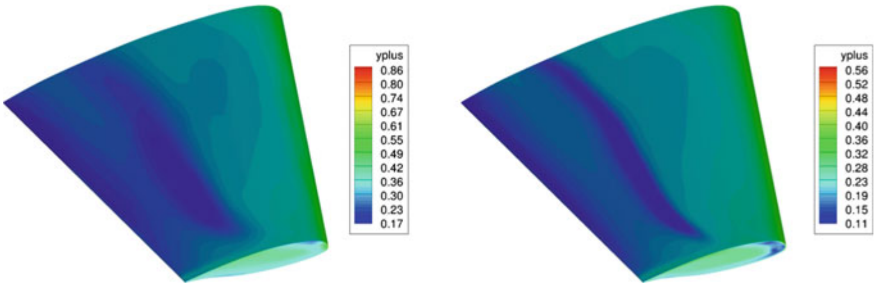
6.2.2 Mesh Generation

The computational mesh is generated by using the ICEM CFD commercial package. Once defined the wing shape from a specific design vector by using the aforementioned CST approach, three wing sections, namely root, mid and tip sections, are imported within ICEM CFD and employed to generate support curves and surfaces. A replay script file is used to fully parametrize the blocking arrangement, the structured grid generation and the mesh export in a file format suitable for the CFD flow solver. Eight blocks are designed around the wing shape and a family of two grids is defined: the coarse and fine mesh consist respectively of 712,448 cells and 2,959,872 cells. A sketch of the surface mesh distribution is shown in Fig. 9. Both meshes are conceived to respect the $y^+ = \mathrm{O}(1)$ condition, as also shown in Fig. 10 where the contour map of y^+ distribution on the wing surface is depicted. The coarse mesh will be used for optimization studies, while the fine mesh will provide more accurate comparisons of the aerodynamic flow for optimized shapes at the end of the optimization process.



(a) Coarse mesh (b) Fine mesh

Fig. 9 Computational mesh on DPW wing surface



(a) Coarse mesh (b) Fine mesh

Fig. 10 y^{+} distribution on DPW wing surface

6.2.3 Optimization Studies

Similarly to the previous case, three optimization approaches are compared: a plain evolutionary-based optimization and two surrogate-assisted approaches. Table 4 provides details of each study. In particular, PGA (Plain Genetic Algorithm, also referred to as DGA, Direct Genetic Algorithm) features a single-objective, genetic algorithm run (40 generations with a population size of 96 candidates), calling the CFD flow solver as fitness evaluator. As in the previous case, the EGO method is launched through the Dakota interface. SBOSA-EIL approach is the same as for the RAE 2822 case, but here the total budget of true function evaluations is set to 102 and the database breakdown is as follows:

- 16 samples are suggested by LHS and the POD/RBF surrogate model is initialized;
- 56 samples are iteratively provided by applying the EI-like in-fill criterion;
- 30 samples are finally suggested by optimizing on the surrogate with repeated GA calls.

Table 4 Summary of optimization studies for DPW wing

ID	Obj. function eval.	Opt. method	Total no. evaluations
PGA	CFD	GA	3936
SBOSA-EIL	POD/RBF	SBO (EI-like + GA)	102
EGO	Kriging	Dakota EGO	502

The baseline shape is the wing geometry DPW-W1 proposed within the 3rd Drag Prediction Workshop. With respect to the original design point (Mach $= 0.76$, AOA $= 0.5°$), some changes are introduced in order to make the optimization problem harder. Indeed, the primary interest is to verify the capability of surrogate-assisted techniques to recover optimal or sub-optimal design solutions by using limited computational resources. To this aim, a convincing test is to force the algorithm to start from poor aerodynamic solutions and observe how quick it is to fall back in promising regions. Hence, a new design point suitable for optimization purposes is identified at a higher Mach number ($= 0.78$) and a higher angle of attack ($= 1.0°$). Table 5 reports the aerodynamic coefficients and objective function values for three design points, namely the original one, the optimization one and an intermediate one. The latter has been reported in order to highlight that, by increasing only the Mach number at fixed angle of attack, the lift coefficient and hence the induced drag would not be altered significantly: as a matter of fact, moving the AOA to 1° would force the optimizer to heavily work on twist and shape design variables in order to decrease the lift coefficient and the induced drag accordingly.

The true objective function history of the SBOSA-EIL run is reported in Fig. 11. For the sake of clarity, a constant is added to the objective function in order to allow for using a logarithmic scale. Candidate samples are clearly distinguished according to the criterion used to select them (LHS, EI-like and GA). The solid dark grey line marks the advancement of the minimum value of the true objective function. As in the airfoil case, the initial LHS stage (dark grey diamonds) provides only for the initialization of the surrogate, as the aerodynamic performance is very poor. The objective-driven in-fill points (light grey squares) contribute to progressively drop the objective function levels prior to the final, steepest downhill during the GA-assisted stage (black circles). Unlike the previous case, here the best candidate is found during the GA-assisted search.

Figure 12 shows the progress of the true objective function minimum for all optimization studies. The x-axis (progressive number of candidates) is reported in

Table 5 Comparison of aerodynamic coefficients of the baseline wing

Mach	AOA (°)	C_L	C_D	C_M	Obj. function
0.76	0.5	0.504	0.0237	−0.0722	−16.323
0.78	0.5	0.510	0.0283	−0.0785	−9.927
0.78	1.0	0.563	0.0337	−0.0781	−8.776

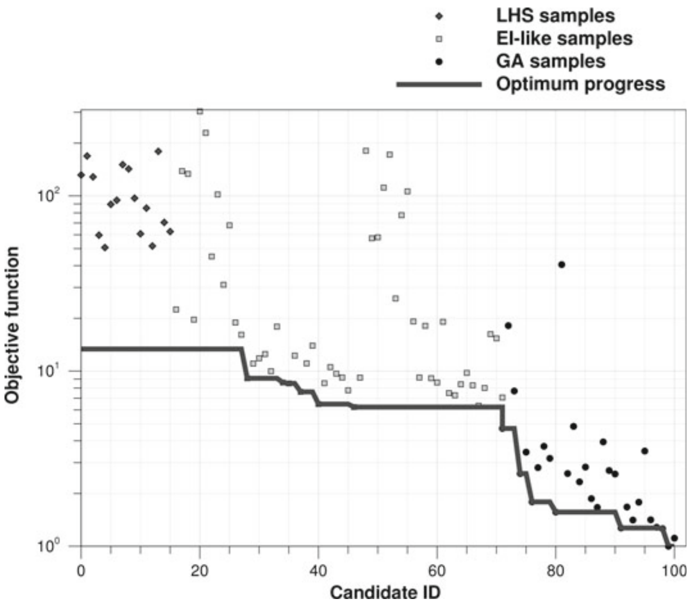


Fig. 11 Objective function history of DPW wing SBOSA-EIL optimization run

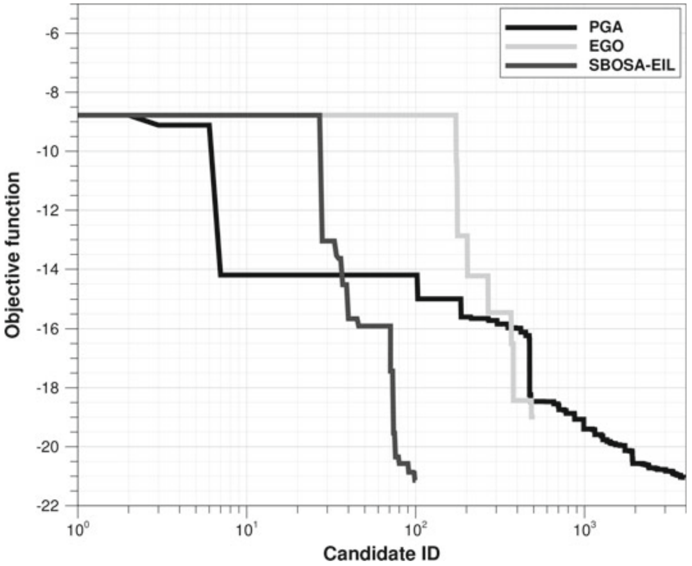


Fig. 12 Progress of minimum values of the objective function for DPW wing optimization studies

logarithmic scale for the sake of clarity as different scales are involved. The SBOSA-EIL optimization, despite the very limited number of CFD evaluations, manages to achieve the same performance level with respect to the plain GA. The EGO algorithm once again turns out to stay somewhat in the middle in terms of computational budget and performance. To better compare the algorithms, Table 6 summarizes the results of each optimization study in terms of aerodynamic coefficients, minimum objective function value attained, number of effective CFD evaluations needed to capture it and a measure of CFD evaluations saving with respect to PGA run. It can be observed that PGA and SBOSA-EIL achieve the same objective function minimum value, although the two optima candidates are clearly different as they exhibit different aerodynamic coefficients. This suggests that the optimization case features non-unique optima solutions. However, the valuable point of such a comparison lies in the fact that, by employing surrogate-assisted procedures coupled to adaptive objective-driven training, the global optimum can be detected by exploiting only 3% of the computational budget of a genetic algorithm optimization. Table 7 summarizes the aerodynamic coefficients and objective function values for DPW-W1, SBOSA-EIL and PGA candidates as computed on the fine mesh: small deviations can be observed with respect to the coarse mesh, a slight loss in the lift coefficient for PGA optimum deteriorates the objective performance due to the triggering of the corresponding penalty. On the other hand, the goal function of SBOSA-EIL optimum is even better due to lower drag contribution and lower penalization of the pitching moment coefficient.

In order to evaluate the difference between optimal candidates, aerodynamic comparisons are proposed in the following figures. Figure 13 shows the contour map of the pressure coefficient on the upper surface of each optimal candidate. A general reduction of the wing loading can be observed which allowed to reduce the shock

Table 6 Optimal candidates comparison

ID	C_L	C_D	C_M	Obj. function	CFD evaluations	CFD saving
Baseline	0.563	0.0337	−0.0781	−8.776	–	–
PGA	0.500	0.0236	−0.0522	−21.14	3700	0.0
EGO	0.502	0.0260	−0.0554	−18.99	485	−87%
SBOSA-EIL	0.514	0.0242	−0.0529	−21.14	100	−97%

Table 7 Optimal candidates comparison, fine mesh

ID	C_L	C_D	C_M	Obj. function
Baseline	0.569	0.0340	−0.0796	−7.970
PGA	0.492	0.0228	−0.0489	−21.013
SBOSA-EIL	0.507	0.0234	−0.0505	−21.638

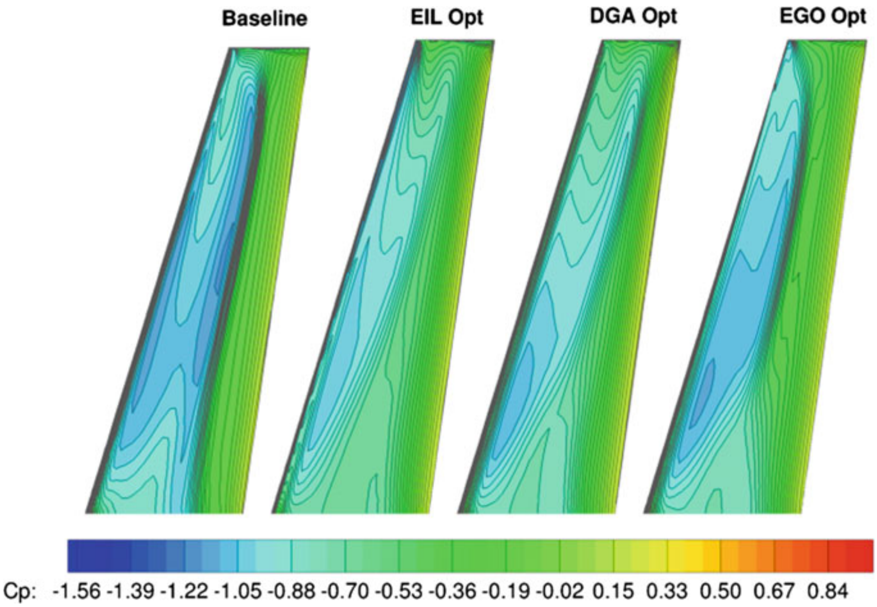


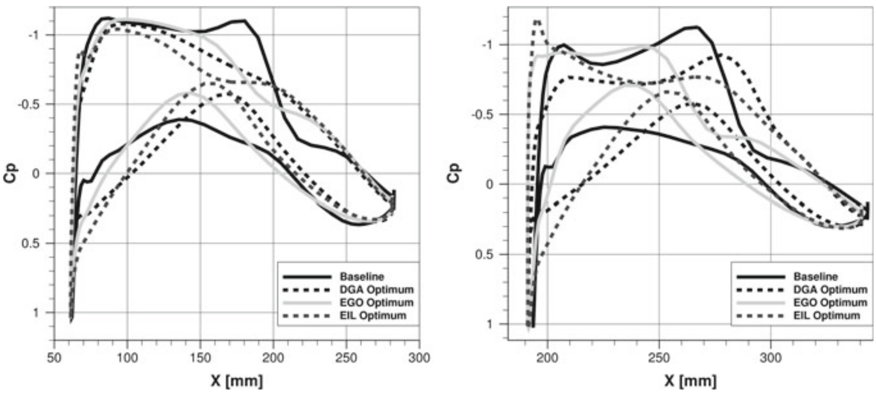
Fig. 13 DPW wing optimal candidates comparison, pressure coefficient contour map

wave strength. This is also confirmed by the fact that the lift coefficient of all candidates are lower than the baseline value and close to the constraint threshold. The SBOSA-EIL solution is very similar to the PGA one, some slight differences are observed in the outboard wing isobars.

Moreover, the shape modification worked to push the wing loading forward in order to satisfy the pitching moment constraint. This feature is more clear in Fig. 14, where the sectional pressure distribution at two spanwise sections (26 and 82% of the span length) is shown. The PGA and SBOSA-EIL solutions present different design choices on the outboard wing, the former being more flat and less loaded, the latter showing a peak near the leading edge followed by a gentle compression. For the sake of completeness, the section shape comparison is also provided in Fig. 15. The most evident geometry modifications can be summarized as follows:

- increase of the outboard twist angle for the SBOSA-EIL optimum candidate;
- airfoil de-cambering for SBOSA-EIL optimum to compensate for twist angle increase;
- reduction of leading edge radius for all the optimized shapes.

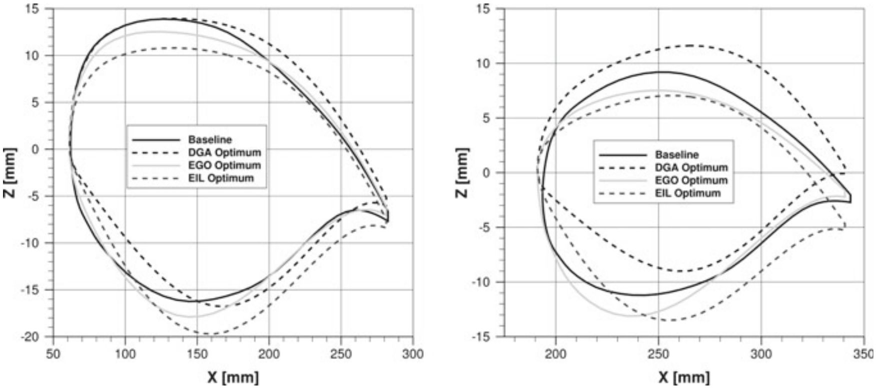
Of course, each design choice is strictly related to other design features, but only their combination affects the objective function/constraint evaluation. For instance, PGA and SBOSA-EIL represent two different design examples, however they share the same level of performance in light of the chosen objective function. Indeed, in the former case the twist angle distribution in slightly higher than the DPW wing



(a) Inboard section, $\frac{y}{b} = 0.26$

(b) Outboard section, $\frac{y}{b} = 0.82$

Fig. 14 DPW wing optimal candidates comparison, sectional pressure distribution

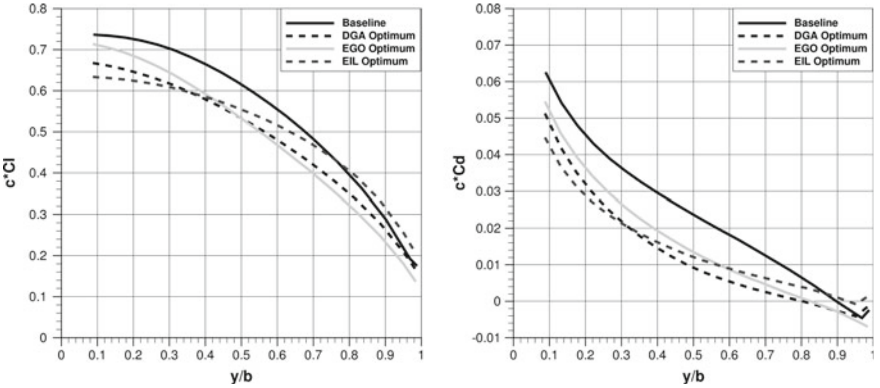


(a) Inboard section, $\frac{y}{b} = 0.26$

(b) Outboard section, $\frac{y}{b} = 0.82$

Fig. 15 DPW wing optimal candidates comparison, section shape

but the wing airfoils are shaped to lower the wing loading along the wing span; in the latter case, the section shape in the inboard wing region is designed to minimize the wing loading in order to reduce the main source of wing pressure drag as much as possible and, to compensate this effect, the outboard wing twist is increased. Figure 16 provides a clarifying overview of such a concept as it reports the wing lift and (pressure) drag loading along the wing spanwise direction. It can be observed how the drag levels on the inboard region are reduced for the SBOSA-EIL candidate by reducing the sectional lift accordingly, while the PGA optimum exhibits very low drag contribution on the mid-outboard wing.



(a) Lift loading $c\,C_l$

(b) Drag loading $c\,C_d$

Fig. 16 DPW wing optimal candidates comparison, wing span loading

7 Conclusions

The paper proposed a surrogate-assisted methodology suitable to aerodynamic shape optimization. A physics-based surrogate model coupling Proper Orthogonal Decomposition and Radial Basis Functions interpolation has been exploited to predict approximate values of the objective functions throughout the optimization process. The surrogate model database has been split in three stages, namely a space-filling Latin Hypercube stage to initialize the surrogate, an adaptive sampling stage in which the model is gradually improved and a final optimization stage in which optimal candidates predicted by the surrogate model are re-injected in the database and the model is updated accordingly. The adaptive sampling phase consists in applying two *ad hoc* in-fill criteria which have been purposely designed to enrich the surrogate model database towards the realization of the exploration/exploitation trade-off. The first in-fill criterion is designed to mimic the Expected Improvement Function maximization, the second is based on a sort of factorization of the exploration and exploitation effects.

Two aerodynamic cases have been proposed to test the methodology: the shape optimization of the well-known RAE 2822 airfoil and of an isolated wing from the AIAA CFD Drag Prediction Workshops. In the first case, the results obtained by applying three in-fill strategies compare very well with classical evolutionary-based optimization and surrogate-based EGO algorithm, taken as references. In particular, the same aerodynamic performance level (-40% in terms of objective function) of the computationally intensive genetic optimization can be reached by tuning the σ parameter of the Expected Improvement-like criterion. The second test case involves the solution of the flow field around an isolated wing in transonic viscous flow, hence the number of CFD degrees of freedom is quite larger with respect to two-dimensional cases. Moreover, the design optimization problem has been intentionally made more difficult in order to stress the proposed methodology. Despite the increased intrinsic complexity of the case, results are even better as the aerodynamic performance of the global optimum (-140% in terms of objective function), as predicted by the high-fidelity optimization, is achieved by the surrogate-assisted approach at very limited computational cost (only 3% of the high-fidelity optimization).

Such results support the conclusion that surrogate models alone may not provide the right answer within an aerodynamic shape optimization context, especially if transonic viscous flow is considered. However, when coupled to smart adaptive sampling techniques, they allow to catch the basic trends of the objective function without penalizing the design space exploration: indeed, in complex design cases with high non-linearities and multi-modal landscapes, the latter has to be carefully balanced as it may result in unveiling promising regions as well as leading the optimizer to waste time in searching poor solutions.

References

1. Adams, B., Bauman, L., Bohnhoff, W., Dalbey, K., Ebeida, M., Eddy, J., Eldred, M., Hough, P., Hu, K., Jakeman, J., Swiler, L., Vigil, D.: Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.4 user's manual. Technical report SAND2010-2183, Sandia, Apr 2013
2. Amato, M., Catalano, P.: Non linear κ ε turbulence modeling for industrial applications. In: ICAS 2000 Congress. IOS Press, Harrogate, UK (2000)
3. Chandrashekarappa, P., Duvigneau, R.: Radial basis functions and kriging metamodels for aerodynamic optimization. Rapport de recherche RR-6151, INRIA (2007). http://hal.inria.fr/inria-00137602/en/
4. Forrester, A.I.J., Keane, A.J.: Recent advances in surrogate-based optimization. Prog. Aerosp. Sci. **45**(1-3), 50–79 (2009). http://dx.doi.org/10.1016/j.paerosci.2008.11.001
5. Gutmann, H.M.: A radial basis function method for global optimization. J. Glob. Optim. **19**, 201–227 (2001). http://dl.acm.org/citation.cfm?id=596093.596381. https://doi.org/10.1023/A:1011255519438
6. Iuliano, E.: Towards a pod-based surrogate model for CFD optimization. In: Proceedings of the Eccomas CFD & Optimization Conference, Antalya, Turkey (2011)
7. Iuliano, E.: Adaptive Sampling Strategies for Surrogate-based Aerodynamic Optimization, Springer Tracts in Mechanical Engineering, vol. Springer, Application of Surrogate-based Global Optimization to Aerodynamic Design (2015)
8. Iuliano, E., Quagliarella, D.: Surrogate-based aerodynamic optimization via a zonal pod model. In: Proceedings of the EUROGEN 2011 Conference, Capua, Italy (2011)
9. Iuliano, E., Quagliarella, D.: Aerodynamic shape optimization via non-intrusive pod-based surrogate modelling. In: Proceedings of 2013 IEEE CEC Congress on Evolutionary Computation (2013)
10. Iuliano, E., Quagliarella, D.: Proper orthogonal decomposition, surrogate modelling and evolutionary optimization in aerodynamic design. Comput. Fluids **84**, 327–350 (2013)
11. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. J. Glob. Optim. **13**, 455–492 (1998). https://doi.org/10.1023/A:1008306431147. http://dx.doi.org/10.1023/A:1008306431147
12. Kulfan, B.M.: Universal parametric geometry representation method. J. Aircr. **45**(1), 142–158 (2008)

13. Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanathan, R., Kevintucker, P.: Surrogate-based analysis and optimization. Prog. Aerosp. Sci. **41**(1), 1–28 (2005). http://linkinghub.elsevier.com/retrieve/pii/S0376042105000102
14. Schonlau, M., Welch, W.J., Jones, D.R.: Global Versus Local Search in Constrained Optimization of Computer Models. Lecture Notes-Monograph Series, vol. 34 (1998). https://doi.org/10.2307/4356058. http://dx.doi.org/10.2307/4356058
15. Simpson, T.W., Toropov, V.V., Balabanov, V., Viana, F.A.C.: Design and analysis of computer experiments in multidisciplinary design optimization: a review of how far we have come—or not. In: Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA 2008–5802, pp. 1–22. American Institue of Aeronautics and Astronautics (2008)
16. Sóbester, A., Leary, S., Keane, A.: A parallel updating scheme for approximating and optimizing high fidelity computer simulations. Struct. Multidiscip. Optim. **27**, 371–383 (2004). https://doi.org/10.1007/s00158-004-0397-9. http://dx.doi.org/10.1007/s00158-004-0397-9
17. Vitagliano, P.L., Quagliarella, D.: A hybrid genetic algorithm for constrained design of wing and wing-body configurations. In: Bugeda, G., Désidéri, J.A., Périaux, J., Schoenauer, M., Winter G. (eds.) Evolutionary Methods for Design, Optimization and Control Applications to Industrial and Societal Problems. International Center for Numerical Methods in Engineering (CIMNE), Barcelona, Spain (2003)

Efficient Global Optimization Method for Multipoint Airfoil Design

Davide Cinquegrana and Emiliano Iuliano

Abstract In the frame of an investigation about surrogate models employed in aerodynamic optimization problems, this work aims at illustrating the suitability of adapted design space sampling to evolutionary optimization. The adaptive sampling algorithm is based on the Weighted Expected Improvement idea applied to a Kriging-based meta-model. A multipoint airfoil optimization is set as test case. A deep investigation is devoted to the tuning of the weights of Expected Improvement function to enhance the performance of the optimization process. A comparison between a pure genetic optimization and a Weighted Expected Improvement approach is proposed. Efficiency and quality of the obtained results are discussed.

1 Introduction

The present paper focuses on investigating smart strategies for guiding the training of a stochastic-based meta-model to be used as fitness valuator in an aerodynamic optimization context. The main aim is to design an in-fill criteria able to enhance the global optimum search in a surrogate-based optimization. Efficient Global Optimization (EGO), introduced by Jones et al. [5], can be considered as an alternative to gradient-based and stochastic methods. Based on Expected Improvement Function (EIF) estimation, it can be considered as a 'smart' sampler of the design space, since it represents a good compromise between exploration and exploitation: indeed, the search of global optimum can be theoretically achieved with a reduced number of objective function evaluations. This adaptive sampling technique picks additional points where a proper balance between high probability of improvement and high

D. Cinquegrana (✉)
Fluid Mechanics Department, Computational Fluid Dynamics Group, Capua, Italy
e-mail: d.cinquegrana@cira.it

E. Iuliano
Fluid Mechanics Department, Multidisciplinary Analysis and Optimization Group, Capua, Italy
e-mail: e.iuliano@cira.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_6

uncertainty in the surrogate prediction is realized. In fact, two different contributions can be identified in the EIF definition: the first is related to exploitation, which consists in trusting the meta-model approximation and, hence, sampling close to the predicted global optimum; the second is called exploration, which instead aims at discovering unveiled portions of the design space, thus reducing the surrogate prediction uncertainty. In the original version of the EGO algorithm, as reported by Jones, these two contributions are perfectly balanced: however, a weighting approach can be followed to give more emphasis to exploitation or exploration depending on the evolution and the status of the search process. This topic will be extensively investigated in Sect. 7 where different criteria for weighting the EIF will be proposed.

Several papers can be found dealing with optimization by EGO. Sasena et al. [10] introduced the Generalized Expected Improvement and applied it to the design of a hybrid electric vehicle, aiming at maximizing the fuel economy. In the context of structural optimization problems, Sóbester [11] has showed a variable global-local bias scheme via the Weighted Expected Improvement Function (WEIF) based on Radial Basis Function (RBF) meta-models. Glaz et al. [4] applied the EGO algorithm to noise reduction of rotor blades with a design space composed of 4 design variables, representing the blade structural thicknesses and non-structural masses.

The main effort of the present paper is focused on assessing the effects of weighting the EIF within the context of an aerodynamic shape optimization problem. As a first approach, a stepwise change of the weighting parameter is performed underlining the significant role that it can play even when no prior knowledge of the problem at hand is available. Furthermore, an adaptive change of the weighting parameter has been designed: it is based on the correlation metrics computed through the meta-model cross-validation. In fact, during the updating phase of the meta-model, this approach aims at increasing the contribution of the exploitation term when the model is deemed to be more reliable. A similar attempt was also proposed recently by Xiao et al. [12], consisting in adaptively changing the weight parameter by means of an artificial intelligence algorithm: the method was applied to the design of an electromagnetic device, but the author deferred the global conclusions to testing the algorithm on more challenging cases.

The paper is organized as follows: the main topic and the optimization algorithms are introduced in the first section, followed by a short section devoted to the theoretical background of Kriging model. Cross-Validation (CV) and the corresponding metrics employed to evaluate the reliability of the surrogate model are presented in a dedicated section. The main section is the one that illustrate the different updating strategies adopted in this work. Then a section is devoted to the main test case description and also the results will be shown. A deep investigation on the effects of the turbulence model are also reported in Sect. 7.2.3. Finally, discussion and future work are described in the conclusions.

2 Methods

This section is devoted to the description of the step defining the EGO chain, that can be summarized as follow:

1. Generate an initial experimental design dataset.
2. Do the high-fidelity evaluations at the points generated in Step 1.
3. Fit a meta-model to the data generated in Steps 1 and 2.
4. Use the meta-model to evaluate the Expected Improvement at un-sampled points in the design space to choose the location to the next expensive function evaluation.
5. Perform the high-fidelity evaluation at the point(s) selected in Step 4.
6. Use the new data point(s) to update the meta-model.
7. Iterate through Steps 4 to 6 until the maximum number of f_{obj} evaluations is reached.

The initial dataset of samples is defined by means of Latin Hypercube Sampling, LHS. The geometries that comes out from the design vectors so defined are successively analyzed by the high-fidelity aerodynamic model, i.e. a Computational Fluid Dynamics (CFD), Reynolds-Averaged Navier-Stokes (RANS) code to evaluate the objective function, f_{obj}, for each sample. At this time, a first, approximated, meta-model can be addressed by Kriging, that mimic the behavior of the f_{obj} in the Design Space. Then, as described in the procedure, the further samples are chosen sequentially by various in-fill strategies tested for this work, all derived starting from EGO approach.

The exiting criteria of the loop is based on the fixed computational CFD budget devoted to evaluate the f_{obj}. Two different budget are defined: the first one is equal to five times the number of design variable, defining a Validation stage Data-Base (DB); the second one, which will follow the surrogate model validation, is devoted to the optimization stage. In details:

- Validation Stage
  - Generate an a-prioristic DB consisting of 25% of the validation CFD budget
  - Generate a DB of CFD simulation consisting of 5 times the Design Variable. This analysis was performed to test the predictive potential of the surrogate model.
- Optimization Stage
  - Sequential CFD evaluation dedicated to the f_{obj} optimization until to reach the final CFD budget.

Effort of the work is focused on the sequential updating strategy, based on the weighted Expected Improvement: several strategy, explained in a devoted Sect. 5 are investigated and the results compared with a pure-genetic optimization and and with a Surrogate-Based Optimization (SBO) approaches.

3 Metamodel Building: Gaussian Process Approach

The meta-model that mimic the behavior of objective function, f_{obj}, in the Design Space of Variable is based on Kriging interpolation technique. Furthermore, since that this method is able of evaluate an estimation of the error in its prediction, an improvement function can be defined. Based on this findings, an in-fill criteria can be built, as will be described in this section and in the next one.

Kriging is a Gaussian process [7, 8] based models, where the basis function are:

$$\psi^{(i)} = corr\left[Y\left(x^{(i)}\right), Y\left(x\right)\right] = \exp\left(-\sum \boldsymbol{\theta}_j |x_j^{(i)} - x_j|^{p_j}\right) \tag{1}$$

The variance of the basis function can be controlled in each of k dimensions by θ_j, and also the exponent p_j can be varied. The unknown parameters θ_j and p^j are chosen to maximize the Maximum Likelihood Estimation (MLE) function (details in McKay et al. [7] work):

$$ln(L) \simeq -\frac{n}{2}\ln\left(\hat{\sigma}^2\right) - \frac{1}{2}\ln(|\Psi|) \tag{2}$$

where

$$\widehat{\sigma}^2 = \frac{\left(y - \mathbf{1}\mu\right)^T \Psi^{-1} \left(y - \mathbf{1}\mu\right)}{n} \tag{3}$$

and ψ is an $n \times n$ matrix of correlations between the sample data with each element given by (1). After determined θ and p, function prediction can be made at unknown design space vector **x** using:

$$y(x) = \mu + \psi^T \psi^{-1}(y - 1\mu) \tag{4}$$

where, for Ordinary Kriging the mean term is defined as:

$$\mu = \frac{\mathbf{1}\Psi^{-1}y}{\mathbf{1}^T\Psi^{-1}\mathbf{1}} \tag{5}$$

In Universal Kriging, employed for this work, the mean term can be expressed as:

$$\mu = \mu(\mathbf{s}) = \sum \mu_i \nu_i(\mathbf{s}) \tag{6}$$

taking the form of a first order polynomial regression.

As explained before, EGO is founded on the provision of an estimated error by kriging in its prediction, the Mean Square Error, valued as:

$$s\left(x\right) = \sigma^2 \left[1 - \psi\Psi\psi + \frac{1 - \mathbf{1}^T\Psi^{-1}\psi}{\mathbf{1}^T\Psi^{-1}\mathbf{1}}\right] \tag{7}$$

See Saks et al. [9] for the full derivation.

4 Rating a Metamodel: Cross Validation and Metrics

Cross-Validation (CV) analysis provides an average error when the meta-model is applied to a sample not collected in the training DB. Here is adopted a K-fold CV, that randomly breaks the Dataset into K partitions removing temporarily the jth subset from the dataset. With this reduced set, the model is trained on the remaining $K - 1$ subset. Then, the error is evaluated when predicting the jth part of the data, and finally, this is combined with the K prediction error estimations. A sub case of K-fold happens when $K = N$: in this case each subset contains only a single pattern and is known as Leave-One-Out CV (LOO-CV). The information coming out from this analysis shows the points where the meta-model works well and the regions of the Design Space variables that need more data (i.e., additional CFD runs).

For prediction error estimation, different metrics are considered. The first is the Pearson's correlation coefficient, defined as:

$$\rho = \frac{N\sum_{i=1}^{N} f_i \hat{f}^{-k(i)}(x_i) - \sum_{i=1}^{N} f_i \sum_{i=1}^{N} \hat{f}^{-k(i)}(x_i)}{\sqrt{N\sum_{i=1}^{N} f_i^2 - \left(\sum_{i=1}^{N} f_i\right)^2}\sqrt{N\sum_{i=1}^{N}\left[\hat{f}^{-k(i)}(x_i)\right]^2 - \left[\sum_{i=1}^{N} \hat{f}^{-k(i)}(x_i)\right]^2}} \tag{8}$$

that ranges between -1 and 1 and provides the ratio between the covariance of parameter of the modeled function $\widehat{f}$ and true function f, and the product of their standard deviations. If it is close to zero, the functions are weakly correlated and, hence, we can expect that the prediction model badly reproduces the variation of the true function. On the other and, if approaches the unity value strong correlation between function is expected.

Other loss function adopted is the R^2 coefficient, defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (f_i - \hat{f}^{-k}(x_i))^2}{\sum_{i=1}^{n} (f_i - \overline{f})^2} \tag{9}$$

that express the ratio between the residual sum of squares and the total deviation. It ranges between 0 and 1: the higher the value, the better is the goodness of fit. Furthermore, the mean square error (MSE) metric:

$$MSE\left(\hat{f}\right) = \frac{1}{N}\sum_{i=1}^{N}\left(f_i - \hat{f}^{-k}(x_i)\right)^2 \tag{10}$$

that gives an estimate of the expected test error by using the squared error as loss function. It ranges between zero and plus infinity, and smaller values indicate smaller errors.

The latest parameter that is keep in account to rate the meta-model is its monotonicity property, valued with the G metric:

$$G = \sum_{i=1}^{N} \sum_{j=1}^{i} -min \left[0, \frac{\hat{f}(x_i) - \hat{f}(x_j)}{f(x_i) - f(x_j)} \right] \quad (11)$$

that is always positive, and the lower is G value, the better the behavior of the meta-model in preserve the true data relations, in particular, $G = 0$ indicates that global monotonicity is preserved.

5 Infill Strategy: Expected Improvement

The Expected Improvement (EI) is an auxiliary function based on actual minimum, surrogate predictions and uncertainties [5]. It is a probabilistic quantity that predicts the expected decrease in the current minimum value if a new function evaluation were performed at that point. Hence, in a sequential sampling of the design space, the next function evaluation in the optimization process should be performed at the point of maximum expected improvement to maximum the probability of encountering a new minimum value. In a minimization problem, the improvement over the current best design is written as:

$$I(x) = \max\left(f_{min} - f\left(x\right), 0\right) \quad (12)$$

where f_{min} is the best design out of all the sample points on which the surrogate is defined. Keep in mind the feature of a Gaussian process to model its error in prediction with Eq. 7, the expected improvement function can be derived from the expected value of the improvement function:

$$EI = E[I\left(x\right)] = \begin{cases} \left(f_{min} - f(x)^*\right) \Phi\left(\frac{\left(f_{min} - f(x)^*\right)}{s(x)}\right) + s\left(x\right) \phi\left(\frac{\left(f_{min} - f(x)^*\right)}{s(x)}\right) & if\ s\left(x\right) > 0 \\ 0 & if\ s\left(x\right) = 0 \end{cases} \quad (13)$$

where Φ () and ϕ () denote the cumulative distribution function (cdf) and the probability density function (pdf) of the standard normal distribution, respectively. Analyzing the terms, on the left the exploitation contribution, related to the minimum distance between the value predicted by meta-model and the actual minimum value, while on the right is the exploration term, that keep in account the errors of the meta-model itself. In the meanwhile the meta-model is in its training stage, especially when it deal with high dimensional problems, in the early stage it is an hazard to trust in a good prediction. Indeed, the choice of the next sample should be direct toward an exploration direction.

Introducing a scalar weight, w, it is possible to give different emphasis to the two terms composing the EI: the one related to the model uncertainties and the one related to the actual minimum value. Then the Weighted Expected Improvement (WEI) can be defined as follows:

$$WEI\,(x) = \begin{cases} w\left(f_{min} - f\,(x)^{*}\right)\Phi\left(\frac{(f_{min} - f(x)^{*})}{s(x)}\right) + (1-w)\,s\,(x)\,\phi\left(\frac{(f_{min} - f(x)^{*})}{s(x)}\right) & if\ s\,(x) > 0 \\ 0 & if\ s\,(x) = 0 \end{cases} \tag{14}$$

According to the value of w, ranging from [0, 1] the infill criteria can be biased towards exploitation, with high value of w (more emphasis to the predicted minimum value), or exploration (emphasis on regions of high uncertainty of the model) with low value of w.

In this paper, different strategies are applied to enhance in efficiency the EGO when deal with high-dimensional, multi-modal problem like the multi-point airfoil design. After designed a starting dataset composed by 18 elements (30% of the DB size considered for validation) by means of LHS, various EGO-based sequential infill strategies are implemented. In particular, a first investigation regards a parametric study in change the weight w to check the impacts on absolute value of the minimum OF reached, its variance, and metrics obtained in a CV to rate the different approaches. The second and other approaches are described in the next subsections. As a term of comparison a full LHS Dataset and SBO are performed. In summary:

- LHS(18) + EGO(52): for EGO are also included some attempt with w variations
    - $w = [0.25, 0.3, 0.5, 0.6, 0.75, 0.9]$
    - w assigned with a step law
    - $w = w(\rho)$
- LHS(18) + SBO(52): SBO is intended as Surrogate Based Optimization with Genetic Algorithm
- LHS(70)

in the bracket are reported the CFD evaluations of f_{obj}.

Since that the first (and common) part of Database of CFD runs was based on randomly generated LHS samples, in order to eliminate the effects of distorted performance figures caused by one or more points of the initial design landing near the global basin, five different LHS sampling was conducted, and then different surrogate model from sequential updating was statistically evaluated to estimate the incidence of the different starting points.

5.1 Weighted EI—A Global-Local Approach

This approach is based on the previous considerations. Keeping in mind that the larger values of weight w, the more local the scope of search will be, a small value at beginning of in-fill is chosen and the it grows up in a step-wise manner. This could help the model in the first stage when large uncertainty can characterize highly multi-modal function. In the next equation is showed the step-wise manner in which the weight grows up in the sequential infill of validation stage.

$$w = \left\lfloor \frac{i}{nCFD} \right\rfloor 0.1 + 0.5 \tag{15}$$

where $\lfloor\rfloor$ is the floor function, i is the current CFD evaluation, $nCFD$ is the step size, i.e. the number of CFD evaluation where the weight w is kept constant. Here, the range of w is $w \in [0.02, 0.6]$.

5.2 *Adaptive Approach*

The previous strategy is based on the assumption that in the early stage the meta-model can fail in predict the improvement function, especially when dealing with high-dimensional optimization problem. Then, it automatically grows up the weight starting from low value of w. The following strategy instead relate the changing weight of WEI according to a parameter that should rate the actual reliability of the training meta-model. The parameter is the Pearson's Coefficient, Eq. 8 of Sect. 4. The law is opportunely defined as follow;

$$w(\rho) = a + b \tanh(2.5\rho - c) \tag{16}$$

where the scalar parameter are chosen so that $w \in [0.02, 0.6]$. This strategy was also defined in order to achieve better metrics value in the Validation phase of the database, an trying to correlate a good rating with also optimum minimum value of f_{obj}.

6 Multipoint Airfoil Shape Optimization

The RAE 2822 airfoil [2, 3] has been selected as the baseline for a multi-point aerodynamic optimization. The airfoil shape is shown in Fig. 1.

The airfoil is a rear-loaded, sub-critical geometry, designed to exhibit a roof-top type pressure distribution at design conditions ($Mach = 0.66$, $Cl = 0.56$) [3]. The coordinates of the lower and upper surfaces and the wind tunnel flow conditions can be found [2]. The main geometric features of the airfoil are summarized in Table 1.

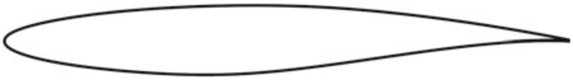


Fig. 1 Baseline Airfoil geometry of RAE 2822

Table 1 Baseline airfoil features

Geometrical feat	Value
Chord (m)	0.61
Maximum thickness-to-chord ratio	$0.121@x/c = 0.38$
Maximum camber-to-chord ratio	$0.0126@x/c = 0.76$
Leading edge radius (m)	0.00827
Airfoil area (m^2)	0.0776
Trailing edge angle (deg)	9

Table 2 Free-stream conditions for design points

DP_i	M	Re	AoA (deg)
DP_1 (Case9)	0.734	6.5×10^6	2.65
DP_2 (Case10)	0.754	6.2×10^6	2.9

6.1 Problem Description

The flow conditions, the aerodynamic and geometric constraints for the different design points, are the inputs for the optimization process. The flow conditions include prescribed angle of attack (AoA), Mach number (M), Reynolds number (Re), for each Design Point (DP_i), as it is shown in Table 2.

The angle of attack was chosen to reach, on the baseline, the experimental pressure coefficient distribution. As will be show in the next sections, those value are sensitive to the turbulence model employed in RANS computation. The f_{obj}, concerning both the Design Point, is defined in the following:

$$f_{obj} = \frac{1}{2}\left(\frac{C_d + C_{d,t} + C_{d,l}}{C_l}\frac{C_{d,0}}{C_{l,0}}\right)_{DP1} + \frac{1}{2}\left(\frac{C_d + C_{d,t} + C_{d,l}}{C_l}\frac{C_{d,0}}{C_{l,0}}\right)_{DP2} \tag{17}$$

where:

$$\begin{cases} \left(C_{d,t}\right)_{DP_i} = 0.01 \max\left(0, C_m^0 - C_m\right)_{DP_i}; \\ \left(C_{d,t}\right)_{DP_i} = 0.1 \max\left(0, (C_l^0)^2 - C_l^2\right)_{DP_i} \end{cases} \tag{18}$$

The optimal solutions should also satisfy the design constraints, which are usually classified as aerodynamic and geometric constraints. The first ones should ensure a predefined level of performance in terms of aerodynamic coefficients, the second ones are usually set to satisfy manufacturing and structural requirements. The aerodynamic constraints and penalties for the present design task are the following:

- Prescribed minimum lift coefficient $C_l^0|_k : C_l|_k \geq C_l^0|_k$
- Prescribed minimum pitching moment coefficient $C_m^0|_k : C_m|_k \geq C_m^0|_k$

Table 3 Aerodynamic constraints threshold

	$C_l^0\|_1$	$C_l^0\|_2$	$C_m^0\|_1$	$C_m^0\|_2$	$C_d^0\|_1$	$C_d^0\|_2$
Coarse grid	0.800	0.791	−0.0962	−0.112	0.0183	0.0285
Fine grid	0.801	0.785	−0.0956	−0.109	0.0177	0.0276
Experiment	0.803	0.743	−0.099	−0.106	0.0168	0.0242
Constraints	0.800	0.750	−0.100	−0.110	–	–

Table 4 Geometric constraints threshold

$\left(\frac{t}{c}\right)_{max}$	$\left(\frac{t}{c}\right)_{min}$ @80	$R_{min} l.e.$
0.121	0.04	0.004

where $C_l^0|_k$ and $C_m^0|_k$ are the lift and pitching moment coefficients, respectively, of the initial geometry, resumed in Table 3.

The Geometric constraints concerns a prescribed maximum and minimum thickness ratio, and a minimum leading edge nose radius: those value are resumed in Table 4. As explained in the next subsection, those constrain are implicitly satisfied.

6.1.1 Airfoil Parameterization

The RAE2822 is parameterized by a volumetric NURBS [6] by means of 14 control points that are considered the design variables, DV, of the design problem. A highly desirable feature of the parameterization approach is the capability of handling and satisfying the geometric constraints in order to generate feasible candidates at any design vector. The control points, showed in Fig. 2, have been distributed to be able to handle these constraints.

6.1.2 Computational Grid and Flow Solver

The grid generator employed to discretize the flow-field and the airfoil geometry is an single block hyperbolic mesh generator with C-grid topology. The size is defined with two level of refining: coarse, with 512 × 64 cells; and fine, with 1024 × 128 cells. For both the first cell height ensure an $y^+ = 1$.

The solver that has performed the flow-fields evaluations is the CIRA in-house code ZEN [1] (Zonal Euler-Navier Stokes), a multi-block structured, finite volume method, a 2nd order spatial discretization (Jameson-Schmidt-Turkel scheme), with 5-stages Runge-Kutta time integration. The flow is assumed as Fully turbulent and, as explained in previous section, the Spalart-Almaras was chosen for DP1 while for DP2 the k-w SST by Menter. Other details on turbulence effect will be illustrated in the last section.

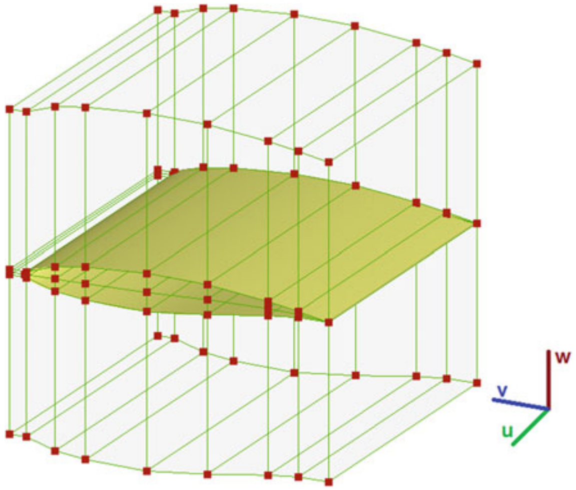


Fig. 2 NURBS control box

7 Results

The approach adopted to verify the efficiency of EGO algorithms is to fix the CFD budget, i.e. a maximum number of f_{obj} evaluation with CFD analysis. In details, three steps are identified to different stage of CFD evaluations:

- Latin Hypercube Sampling of Design Space, with 18 samples evaluations by means of CFD run.
- Sequential Updating:
  - DB validation: up to 70 samples, performed to test the predictive potential of the surrogate model
  - DB optimization: up to 120 samples, to achieve, eventually, further enhancement of objective function

The validations and the optimization stages were repeated five times for each strategies adopted, in order to evaluate the influence of the huge sampling of surrogate EI function, employed to find maximum EI and then selected the new candidate to be evaluated by means of CFD. In the Optimization paragraph are also included an off-design analysis of the optimum airfoil obtained that gave the chance to discuss about the proper choice of the turbulence model and its influence.

7.1 Validation Results

In this paragraph are compared the results of the in-fill strategies implemented, in terms of f_{obj} and of metrics, obtained with K-fold CV. The aim of this comparison is to

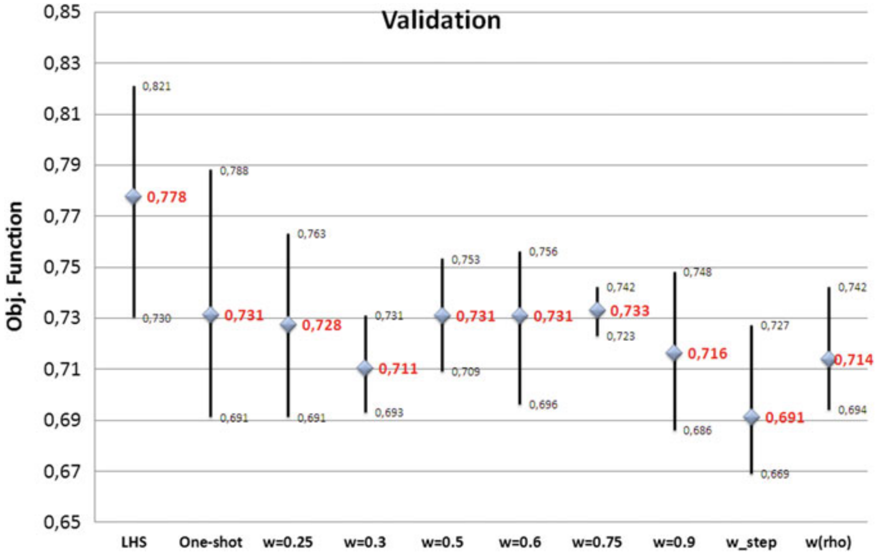


Fig. 3 Objective function values at end of validation stage

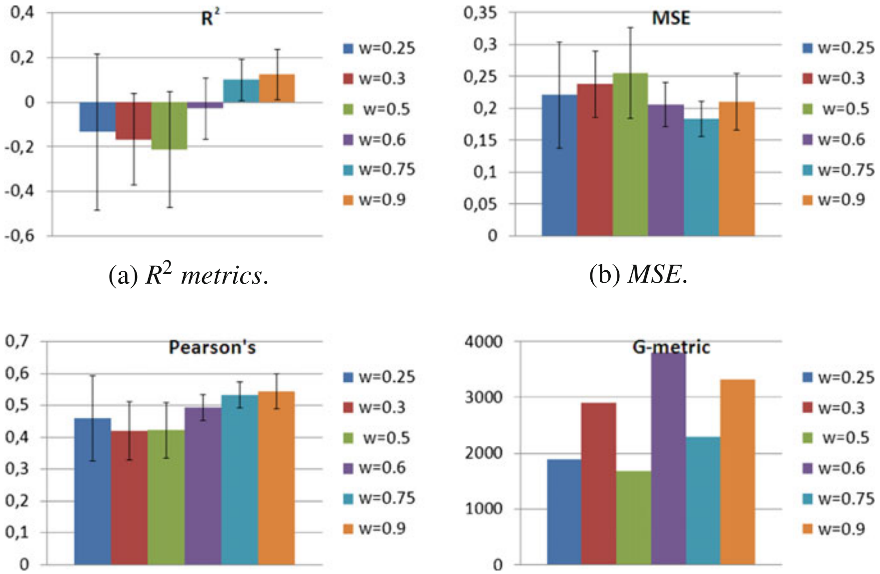
find a relation, if any, between a good metrics relative a specific dataset, and its success in the final global optimum research. Figure 3 resume the results of the minimum, maximum and average f_{obj} obtained by the various methodologies of Surrogate Models adopted in this work. The plot indicates that the better performances, both in average and absolute terms, comes from the EGO with step approach of changing w of Eq. 15.

As a terms of comparison are also included in this validation stage, a full LHS dataset and a SBO dataset. As expected, the former approach lead to worst results concerning the minimum f_{obj} within the validation dataset in the Fig. 3, but also a good metrics was evidenced in terms of Pearsons's coefficients of Eq. 8, MSE of Eq. 10 and R^2 of Eq. 9, not showed here. The latter, SBO, that has the label 'one-shot' in Fig. 3, showed a large variance of minimum f_{obj}, but with an average value f_{obj} that is comparable to the other in-fill criteria with fixed weight w.

Further details of the in-fill criteria based on WEI can be found in the next subparagraph.

7.1.1 Simple Weighted EI: Results

The numerical investigation conducted keeping constant the weight of the EI function has revealed, in the validation phase of DB in-fill, that for this particular problem, the larger the weight, the better are the metrics quality of MSE, R^2 and Pearson's coefficients of the final DB, as shown in Fig. 4(a–c). The G-metric shows a not clear trend and no conclusion can be argued at this stage (see Fig. 4d).



(a) R^2 *metrics.*

(b) *MSE.*

(c) *Pearson's Coefficient.*

(d) *G-metric.*

Fig. 4 Metrics for DB validation

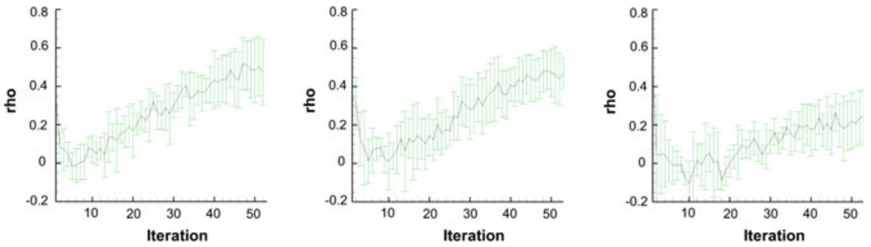
In terms of f_{obj}, in the validation stage is not detected a clear trend, while for the Optimization stage seems that the better performance in minimizing f_{obj} are achieved by infill with the lower value of w.

7.1.2 Step-Wise Weighted EI: Results

The strategies can be compared to a global-local approach: an exploring Strategy at beginning with ($w = 0.05$) and Exploiting in last phase with ($w = 0.65$). This has shown the better results in terms of absolute and average minimum f_{obj} value in the validation stage, as showed in Fig. 3.

7.1.3 Adaptive Weighted-Expected-Improvement: Results

This approach is adopted in the validation stage only. Referring to Fig. 3, the performance in terms of mean and variance of minimum f_{obj} are comparable to a constant low weight approach, with $w = 0.3$ that has reached good results. In Fig. 5 is plot the history of the Pearson's coefficient valued in each step with the its standard deviation. A quite difference can be noticed in the early stage of the in-fill process, where, if compared with the step law results, this method seems to perform better, also in terms of final values of the metric, with lightly broad deviation. The worst results



(a) *Pearson's history in the training with w =w(ρ).*

(b) *Pearson's history in the training with step-wise WEI.*

(c) *Pearson's history in the training SBO.*

Fig. 5 In-line CV results: Pearson's

are the one relative to the SBO training approach, where the in-fill process picks up the sample in the minimum OF predicted by surrogate model.

This approach should be further enriched by other investigation, changing the metric or taking into account a global value that sum up all the metrics here proposed.

7.2 Optimization Results

In this section are shown the results coming from the further infilling stage with its devoted CFD budget of 50 runs to evaluate the f_{obj}. The processes involved at this stage can be resumed as follow:

- LHS(18) + EGO(52)+ EGO(50):
  - $w = [0.25, 0.3, 0.5, 0.6, 0.75, 0.9]$
  - 'step-wise' w + EGO(50)
- LHS(18) + SBO(52) + SBO(50)
- LHS(70) + Standard EGO(50)
- LHS(70) + SBO(50)

In the next sub-paragraph, the comparison with a plain optimization conducted with a genetic algorithm is showed drawing some conclusions about efficiency and quality of results with the criteria here presented.

Figure 6 confirms that, also during the optimization stage the better performance of the step approach of step-wise changing weight of Eq. 15. The worst results in terms of minimum averaged OF value belongs to the Optimization based on the dataset designed with full LHS (with 70 samples), and then optimized with SBO. Starting from the same DB, an EGO optimization shows better results. The pure SBO approach ('one-shot' label in the Figure) shows results comparable with the constant WEI approach when the wight is kept low (from 0.3 to 0.6).

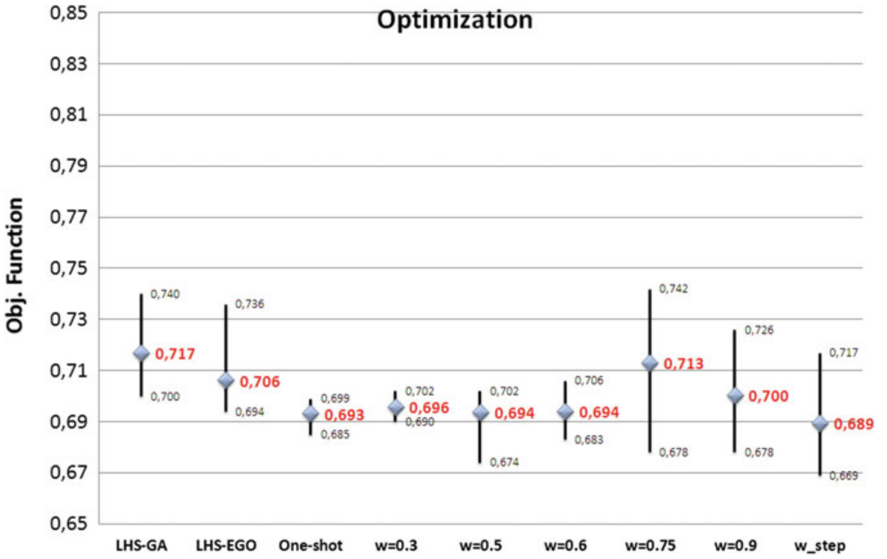
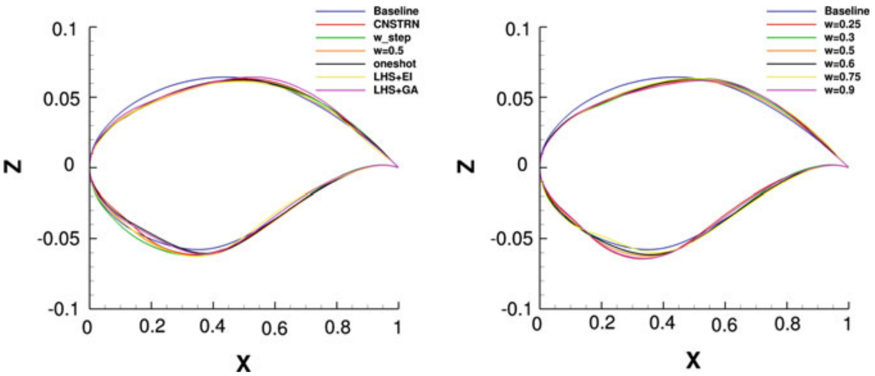


Fig. 6 EGO optimization. Of results

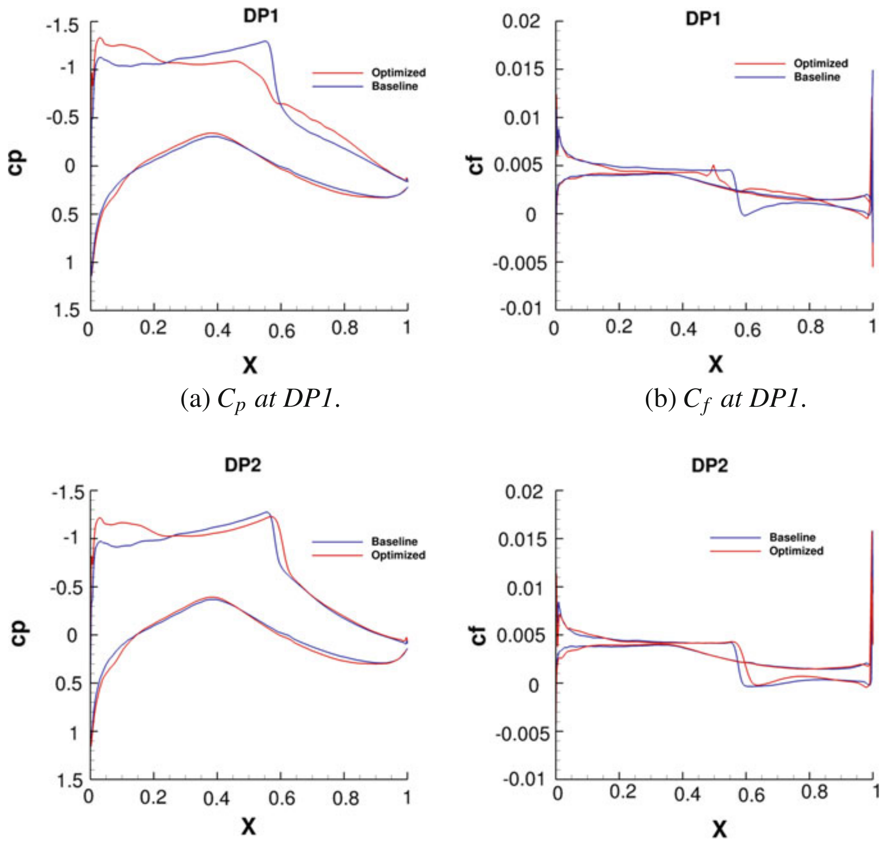


(a) *Baseline and Other Strategies.*

(b) *Baseline and constant weight approach.*

Fig. 7 Optimized aerodynamic shape with various EGO strategies

The Figure in 7 resume the best airfoils coming from the various investigations, compared with the baseline and the overall best (i.e. the w_step). The load distributions are showed in Fig. 8, that is lightly penalized in pitching moment.



(a) C_p *at DP1.*

(b) C_f *at DP1.*

(c) C_p *at DP2.*

(d) C_f *at DP2.*

Fig. 8 Coefficients for the optimized EGO airfoil

7.2.1 Comparison with Plain Optimization

The results of the surrogate optimization are compared with a classical optimization based on genetic algorithm. The strategy of optimization was based on a population of 64 elements and 100 generations, starting from a random population, for a total of 6400 CFD evaluation of the f_{obj}. The plain optimization led to an improved OF of 4%, but with a computational effort larger than EGO algorithm. In fact, to obtain the same performance, plain optimization employed a 1:10 of CFD evaluations ratio (1200 evaluations versus 100 of EGO). Furthermore, to obtain further 4% of improvement in terms of f_{obj}, the ratio of CFD evaluations grows up to 1:50. In the Fig. 9 are compared the evolution of the f_{obj} through the iterations.

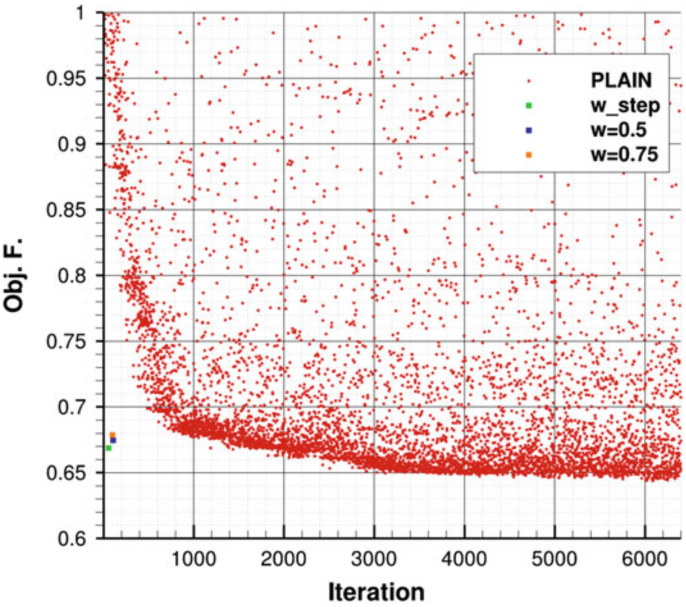
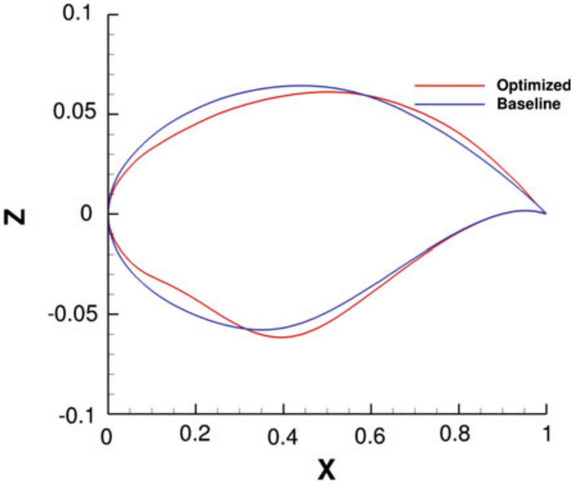


Fig. 9 OF evolution: comparison of EGO and plain strategies

Fig. 10 Optimized Airfoil shape with plain optimization



The improvements of aerodynamics performances obtained with the plain optimization are, in terms of drag coefficient, 4 drag count for DP_1 and 12 drag count for DP_2. In Fig. 10 are compared the baseline and the optimized geometries with plain optimization.

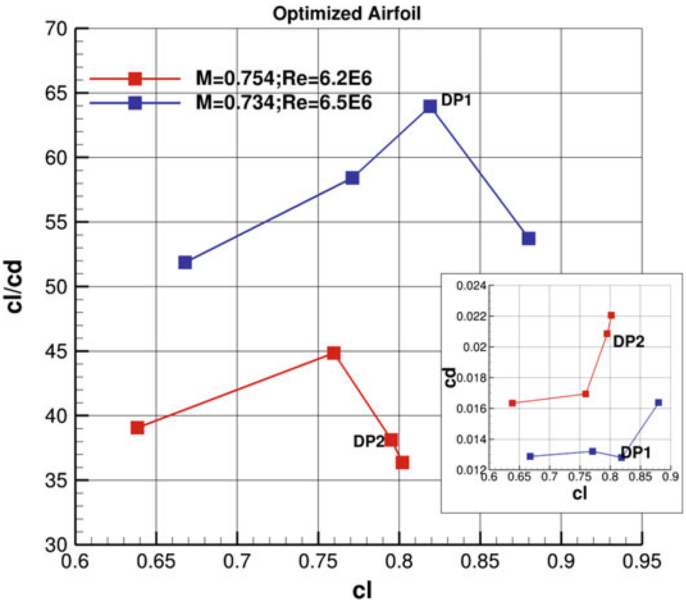


Fig. 11 Efficiency branch of best airfoil from EGO

7.2.2 Off-Design Performance

The best airfoil coming from the EGO strategy based on the step law of Eq. 15 is selected for an off-design analysis aimed to check its performance around the two DP's. Figure 11 shows two branches of Efficiency. For the DP_1 the shape guarantees a maximum value, while for DP_2 the solution is strongly influenced by fixing the angle of attack to a pretty high value, then the optimization results are negatively impacted in terms of drag coefficients.

7.2.3 Impact of the Turbulence Model

This analysis was started considering the efficiency curves showed before in this section that shows, for DP_2, a surplus in terms of lifting coefficient payed in terms of drag count, due to the high value of angle of attacks, that globally deteriorate the value of aerodynamic efficiency. As discussed in the description of the Optimization problem, for the DP_2 test case, the turbulence model involved is the two equation shear-stress transport (SST) $k - \omega$, of Menter. Based on this model, the angle of attack was chosen to reach on the baseline, the experimental pressure coefficient distribution. A study with a different turbulence model was conducted switching from SST to Kok's (turbulent/non turbulent interface, TNT) $k - \omega$ model, for the best airfoil obtained with the Step strategy. Furthermore, the lift coefficient was also

Table 5 Turbulence model influence on O.F

Model	AoA	C_l	C_d	C_m	E_{DP2}	f_{obj}
SST	2.90	0.795	0.02086	−0.1067	38.12	0.668
TNT	2.13	0.743	0.01618	−0.1224	45.92	0.656

fixed while the angle of attacks changes to achieve the target. The results shows that, with TNT, the angle of attacks to reach the lifting coefficient value was lower that SST, $AoA = 2.13$ (deg). The f_{obj} valued with the new airfoil performance shows a greater reward if compared with the one obtained with SST. In the Table 5 a comparisons with the SST model in terms of aerodynamic coefficient for DP_2.

8 Conclusions

The work has shown an investigation on the capability of an EGO method based on the Weighted EI auxiliary function facing with an aerodynamic optimization problem of a multi-point airfoil design. A tuning of the weight was conducted and compared with a global-local approach and strategies based on in-line Cross-Validating metrics. A good comparisons between the implemented in-fill strategy was difficult, masked from the particularly 'stiff' test-case. In future works, a single-point design problem should clarify better the hierarchy of the presented approaches. The final comparisons has shown the efficiency of the method if compared with a pure genetic optimization with a different within 5% in terms of efficiency.

Acknowledgements This research was conducted in the frame of GARTEUR AG-52 group: Surrogate-based global optimization methods in preliminary aerodynamic design.

References

1. Catalano, P., Amato, M.: An evaluation of rans turbulence modeling for aerodinamic applications. Aerosp. Sci. Technol. **7**(7), 493–590 (2003)
2. Cook, P., McDonald, M., Firmin, M.: Aerofoil RAE 2822—Pressure Distributions and Boundary Layer and Wake Measurements, vol. Experimental Data Base for Computer Program Assessment (1979)
3. Fluid Dynamics Panel, W.G..: AGARD Advisory Report No. 138, Experimental Data Base for Computer Program Assessment (1979). ftp://ftp.rta.nato.int//PubFullText/AGARD/AR/AGARD-AR-138/AGARD-AR-138.pdf
4. Glaz, B., Friedmann, P., Liu, L.: Efficient global optimization of helicopter rotor blades for vibration reduction in forward flight. In: Multidisciplinary Analysis Optimization Conferences, pp. –. American Institute of Aeronautics and Astronautics (2006). https://doi.org/10.2514/6.2006-6997. http://dx.doi.org/10.2514/6.2006-6997

5. Jones, D., Schonlau, M., Welch, W.: Efficient global optimization of expensive black-box functions. J. Glob. Optim. **13**(4), 455–492 (1998). https://doi.org/10.1023/A:1008306431147. http://dx.doi.org/10.1023/A3A1008306431147
6. Martin, M., Valero, E., Lozano, C., Andres, E.: Gradient calculation for arbitrary parameterizations via volumetric nurbs: the control box approach. In: 5th European Conference for Aeronautics and Space Sciences (EUCASS) (2013)
7. McKay, M., BeckMan, R., Conover, W.: A comparison of three methods for selecting values of input variables in the analisys of output from a computer code. Technometrics **21–2**, 239–245 (1979)
8. Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanathan, R., Tucker, P.: Surrogate-based analysis and optimization. Prog. Aerosp. Sci. **41**, 1–28 (2005)
9. Sacks, J., Welch, W., Mitchell, T., Wynn, H.: Design and analysis of a computer experiments. Stat. Sci. **4**, 409–435 (1989)
10. Sasena, M.J., Papalambros, P., Goovaerts, P.: Exploration of metamodeling sampling criteria for constrained global optimization. Eng. Optim. **34**, 263–278 (2002)
11. Sóbester, A., Leary, S., Keane, A.: On the design of optimization strategies based on global response surface approximation models. J. Glob. Optim. **33**(1), 31–59 (2005). https://doi.org/10.1007/s10898-004-6733-1. http://dx.doi.org/10.1007/s10898-004-6733-1
12. Xiao, S., Rotaru, M., Sykulski, J.: Adaptive weighted expected improvement with rewards approach in kriging assisted electromagnetic design. IEEE Trans. Magn. **49**(5), 2057–2060 (2013). https://doi.org/10.1109/TMAG.2013.2240662

Part III
Adjoint Methods for Steady and Unsteady Optimization

Checkpointing with Time Gaps for Unsteady Adjoint CFD

Jan Christian Hückelheim and Jens-Dominik Müller

Abstract Gradient-based optimisation using adjoints is an increasingly common approach for industrial flow applications. For cases where the flow is largely unsteady however, the adjoint method is still not widely used, in particular because of its prohibitive computational cost and memory footprint. Several methods have been proposed to reduce the peak memory usage, such as checkpointing schemes or checkpoint compression, at the price of increasing the computational cost even further. We investigate incomplete checkpointing as an alternative, which reduces memory usage at almost no extra computational cost, but instead offers a trade-off between memory footprint and the fidelity of the model. The method works by storing only selected physical time steps and using interpolation to reconstruct time steps that have not been stored. We show that this is enough to compute sufficiently accurate adjoint sensitivities for many relevant cases, and does not add significantly to the computational cost. The method works for general cases and does not require to identify periodic cycles in the flow.

1 Introduction

The adjoint method is commonly used in academia and industry to compute the derivative of a cost function with respect to its design variables. Its greatest appeal lies in the fact that the computational cost is constant in the number of design variables, in contrast to simpler approaches such as finite differences or tangent-linear derivatives. This makes the method feasible for industrial applications with rich design space [4].

Many real-world problems however still present a challenge for the adjoint method. A particular problem is severe unsteadiness, as can be found in turbines, including wind turbines, aircraft wings in high-lift configuration, car engines and

J. C. Hückelheim (✉) · J.-D. Müller
Queen Mary University of London, London, UK
e-mail: j.c.hueckelheim@qmul.ac.uk

J.-D. Müller
e-mail: j.mueller@qmul.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_7

many more [12]. The adjoint method has been formulated for this kind of problem in frequency [13] and temporal space [15], but requires the storage of the full flow history, resulting in prohibitive memory requirements in most cases.

A well-known way to mitigate this problem is the REVOLVE checkpointing algorithm [5]. It stores checkpoints only at carefully chosen time steps, and recomputes each time step when it is needed, starting from the time steps that have been stored. Another approach that has recently been proposed [17] is the compression of checkpoints. Both ideas have one thing in common: the memory requirements are relaxed at the cost of increased computational expense. Furthermore, lossless data compression does not offer large savings in storage space [14], and so the method becomes more useful if lossy compression is used, resulting in errors in the reconstruction of the primal flow field.

We investigate incomplete checkpointing as an alternative, which reduces memory usage at no extra computational cost, but instead offers a direct trade-off between memory footprint and the fidelity of the model. We use a dual time stepping scheme in which the inner iterations are fully converged, so that only physical time steps need to be stored and the adjoint field can be reconstructed based on the fully converged checkpoint, which preserves the accuracy of the result if the inner iteration was fully converged [3]. In addition, we store only selected physical time steps and use interpolation to reconstruct time steps that have not been stored.

The scheme comes at negligible cost for linear or other low order interpolation methods. In particular, the reconstruction from data available in memory is significantly faster than reading the checkpoint from disk, which would be another possible (but slow) way of addressing the memory limitations. Finally, our method does not require any assumptions about the flow such as periodicity.

Since the computational cost of interpolation is negligible in most cases, this work focuses on assessing the accuracy of the adjoint results obtained with this approach.

2 Background

We use an unsteady viscous flow solver for unstructured grids, BDF2 dual time stepping and an implicit solver to converge the inner iterations which was presented in [20]. The adjoint solver is generated using the automatic differentiation tool Tapenade [7] with some hand-coded optimisations for improved speed [2].

2.1 Solving the Flow and Adjoint Equations

The viscous unsteady flow equations can be written as

$$\frac{\partial U}{\partial t} + R(U) = 0$$

and can be discretised using a third-order accurate BDF2 time marching scheme as

$$\begin{aligned}\frac{\partial U}{\partial t} + R(U_t) &= \frac{U_{t-2} - 4 \cdot U_{t-1} + 3 \cdot U_t}{2\Delta t} + R(U_t) \\ &:= \hat{R}\left(U_{t-2}, U_{t-1}, U_t\right)\end{aligned}$$

The above system can be evolved in time by solving the linearised system for U_k and successively updating the converged flow solution U_t at time t

$$\begin{aligned}\left[\frac{\partial \hat{R}(U_{t-2}, U_{t-1}, U_k)}{\partial U_k}\right] \delta U_k &= -\hat{R}(U_{t-2}, U_{t-1}, U_k) \\ U_t &= U_{t-1} + \delta U_k\end{aligned}$$

The unsteady adjoint system can be written as

$$-\frac{\partial v}{\partial t} + \underbrace{\left(\frac{\partial R}{\partial U}\right)^T v - \left(\frac{\partial J}{\partial U}\right)^T}_{:=R_v} = 0$$

and can, like the primal equation, be discretised using BDF2 as

$$\begin{aligned}-\frac{v^{t-2} - 4 \cdot v^{t-1} + 3 \cdot v^t}{2\Delta t} &+ R_v(v^t) \\ &:= \hat{R}_v\left(U_{t-2}, U_{t-1}, U_t\right)\end{aligned}$$

and solved using the same method as the primal equation.

The solution of the adjoint equation requires the history of the flow solution U_t at each time step for the calculation of R_v and the preconditioning matrix P^T. This flow field can be stored during the flow solution and loaded during the adjoint solution, recomputed by running the flow solver again (e.g. following the REVOLVE algorithm), restored approximately from a compressed data, or reconstructed using interpolation, following our new approach.

2.2 *Physical Checkpointing*

We use an approach in which only the physical time steps are stored during the primal computation and restored during the adjoint computation, as presented in [9] and Algorithm 1. The memory requirements are orders of magnitude smaller compared to the brute-force method of storing every iteration.

Aside from the memory requirements that arise from storing the flow trajectory, one major challenge in this approach lies in the need to fully converge the inner loop so that an efficient adjoint method for fixed-point loops can be used [3]. To address

Algorithm 1: Dual timestep with physical checkpointing

```
$n \leftarrow 0$;
$U_0, V_0 \leftarrow$ initial guess;
while $t < t_{final}$ do // primal loop
    $n \leftarrow 0$;
    while $R(U_{t,n}) > cutoff$ do // primal loop
        $U_{t,n+1} \leftarrow$ flow_pseudostep($U_{t,n}$);
        $n \leftarrow n + 1$
    end
    $t \leftarrow t + \Delta t$;
    $U_{t+1,0} \leftarrow U_{t,n}$; // init for next step
    store($U_t$);
end
while $t > t_{init}$ do // adjoint loop
    load($U_t$);
    $V_{t,0} \leftarrow V_{t+1,n}$; // init for next step
    $t \leftarrow t - \Delta t$;
    while $R(V_t) > cutoff$ do // primal loop
        $V_{t,n+1} \leftarrow$ adjoint_pseudostep($V_{t,n}, U_t$);
        $n \leftarrow n + 1$
    end
end
```

this, we use an implicit solver with geometric multigrid and ILU preconditioning to converge in an acceptable time [20].

For our proposed method, the calls to `store()` and `load()` in this algorithm are replaced by calls to augmented routines `gappyStore()` and `gappyLoad()` as described below.

3 Checkpointing with Gaps

The routine `gappyStore()` contains a logic that selects certain snapshots worth storing, which are denoted by the set of stored time steps T_s which are a subset of all time steps T.

In the simplest incarnation of this method, T_s would contain only every n-th time step, for some fixed n. In some special cases it might be beneficial to vary the checkpoint density over time, e.g. to capture a particular phenomenon with a higher accuracy. This was not investigated in this work.

If our method is regarded as a very simple form of data compression, then the data compression ratio is $\|T\|/\|T_s\|$. Obviously we get better compression ratios if we store fewer time steps. For evenly spaced snapshots as suggested above, we obtain a compression ratio $\|T\|/\|T_s\| = n$.

Checkpoints that have not been stored need to be reconstructed. If linear interpolation is used, we can formalise this method as follows. Let t denote the time for which a checkpoint needs to be reconstructed. Also, let t^+ and t^- denote the unique time steps for which all of the following conditions hold:

$$\begin{aligned} t^+, t^- &\in T_s \\ \nexists t^* \in T_s &: t < t^* < t^+ \\ \nexists t^* \in T_s &: t^- < t^* < t \end{aligned}$$

In other words, t^+ and t^- are the closest stored time steps just after and before t, respectively.

We can then implement the `gappyLoad()` routine that can perform linear interpolation or constant interpolation as follows:

Function gappyLoad(t)

```
if $t \in T_s$ then // t was stored
    return load($U_t$);
else
    $U^- \leftarrow$ load($U_{t^-}$);
    $U^+ \leftarrow$ load($U_{t^+}$);
    return $U^- + \frac{t-t^-}{t^+-t^-} \cdot (U^+ - U^-)$;
end
```

If the gap $t^+ - t^-$ is larger than two, this method can be implemented much more efficiently by storing most of the intermediate results. The routine could be implemented for higher orders of interpolation, taking into account more of the surrounding stored time steps.

4 Test Case

4.1 Primal Solver Setup

To test our method we use a RAE2822 aerofoil with a trailing edge that ia truncated at 10% cord length and a 30° angle of attack to provoke a high amount of shedding. The freestream velocity is 0.2 Ma, we use viscous flow. The mesh has around 25000 cells, the solver is node-centred and uses 4 levels of geometric multigrid for faster convergence. The setup and the primal and adjoint flow field are shown in Fig. 1.

For the reference setup, we use a time step size of $t_{ref} = 1$ ms which corresponds to ca. 70 checkpoints for each flow period. While this flow does exhibit periodic cycles, our solver does not exploit this periodicity. Our findings should therefore apply to non-periodic flow as well.

To validate that the chosen time step is fine enough to resolve the primal flow sufficiently we perform another simulation with a time step size of $0.5 \cdot t_{ref}$.

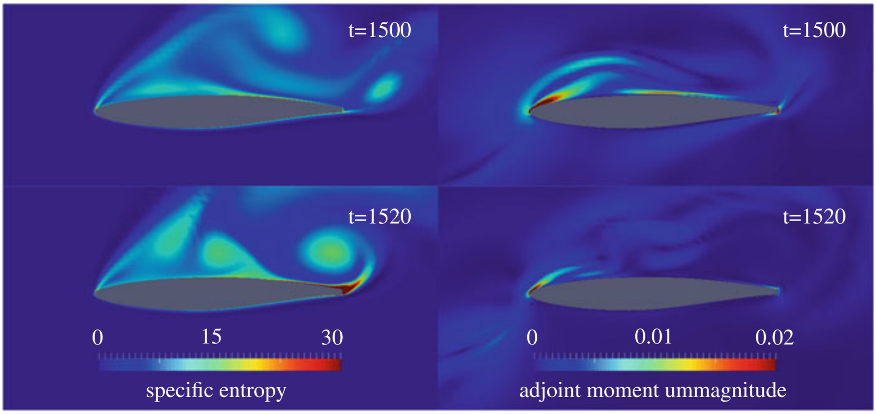


Fig. 1 Snapshots after 1.5 s (1500 time steps) and 1.52 s (1520 time steps). **Left**: Specific entropy, showing strong vortex shedding above the airfoil with a frequency of roughly 70 time steps per period (0.07 s). **Right**: Adjoint momentum at the same time steps. A high sensitivity close to the top surface in step 1500 appears where a vortex is about to form, which can be seen in the primal flow at step 1520. This adjoint peak stems from the small adjoint momentum peaks that are above the airfoil in step 1520 and are propagated back to the airfoil surface during the reverse sweep

Furthermore, to obtain benchmark results for the incomplete checkpointing method we run a series of 6 additional simulations with time step sizes 2, 4, 8, 16, 64 and 256 times t_{ref}. We will refer to these setups as $t_{0.5}, t_2 \dots t_{256}$.

4.2 Incomplete Checkpointing Setup

We use the primal results generated with t_{ref} to initialise the adjoint solver. To investigate if this temporal resolution is also sufficient to get accurate adjoint results, we perform another adjoint simulation with time step size $0.5 \cdot t_{ref}$ for comparison.

Finally, we create incomplete checkpoint trajectories from the reference primal result t_{ref} as follows: For the setup that we will refer to as a_2, we discard all primal states at even time step numbers, and reconstruct them using linear interpolation from the nearest odd time steps. We replace all but every 4th time step by linear interpolation for the a_4 setup and proceed likewise to obtain a_8, a_{16}, a_{64} and a_{256}.

4.3 Adjoint Solver Setup

The adjoint solver is set to the same time step size as the primal solver for the $t_{0.5}, t_2 \dots t_{256}$ setups. For the $a_2 \dots a_{256}$ setups the numerical time step size is t_{ref}.

The cost function is total drag of the airfoil averaged over a time window with a weight function ω to progressively switch the averaging process on and off. Given the total drag at each time step $J(t)$, the average drag J_{avg} can be formulated as

$$J_{avg} = \frac{\sum_{t=1}^{t_\infty} J(t) \cdot \omega(t)}{\sum_{t=1}^{t_\infty} \omega(t)} \tag{1}$$

We use a weight function that ramps up linearly for 0.05 s until it reaches its peak at which it remains for 0.175 s, then ramps down to deactivate the averaging after another 0.05 s. We study three design parameters that are shown in previous works:

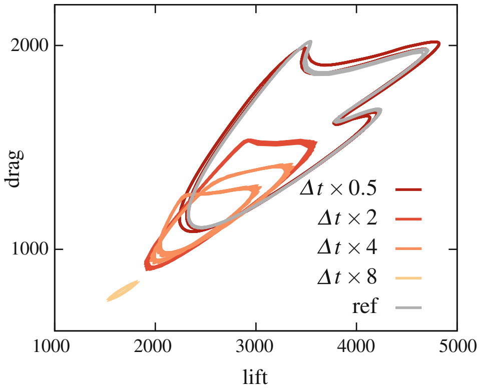
1. **Flow control**: (e.g. [1, 6, 8]) We consider a valve that can inject or remove tangential momentum on the airfoil top just behind the leading edge. This design parameter can vary in time and thus allows us to study transient behaviour of the adjoint field. We compute this sensor for each time step by integrating the adjoint momentum field over a small circular area around the valve location.
2. **Surface node displacement**: (e.g. [10, 19]) We consider the surface nodes' displacement in normal direction as the design vector, which is common for shape optimisation applications. The design parameter does not allow variations in time and is thus based on the time-averaged adjoint field. We use this to study the spatial behaviour of the adjoint field. The spring analogy model is used to project volume sensitivities onto the surface.
3. **Angle of attack**: (e.g. [11]) We consider the shape fixed and only allow an adjustment of the angle of attack. Since this will hide oscillatory errors in space and time, it can be used to study the overall trend of the adjoint field. This sensor is computed based on the cross product of surface sensitivity vectors and point vectors of surface nodes, integrated over the entire airfoil surface.

For the surface sensitivity and angle of attack sensors we require a time-averaged adjoint field. The average is taken over a time window given by the window function ω_a. The adjoint averaging window is twice as long as the cost averaging window.

5 Results

The reference and $t_{0.5}$ solution show very similar primal flow features and lift/drag values match well, suggesting that the reference step size is small enough to resolve the primal flow. Using coarser time steps $t_2 \ldots t_8$, the primal flow is no longer correctly resolved, see Fig. 2.

We observe strong unsteadiness for the adjoint field, see Fig. 1, with peaks of sensitivity close to the leading and trailing edges, and a reverse wake propagated from the leading edge towards the incoming flow. Just like in the primal solution, the unsteadiness is strongest downstream of the airfoil.



t_{ref} ×	lift	drag
0.5	3530 ± 1291	1552 ± 465
1	3517 ± 1192	1561 ± 458
2	2747 ± 846	1219 ± 318
4	2659 ± 700	1175 ± 242
8	1680 ± 134	797 ± 42
16	1446 ± 10^{-12}	702 ± 10^{-12}
64	1446 ± 10^{-2}	702 ± 10^{-2}
256	1445 ± 0.3	701 ± 0.15

Fig. 2 Lift/drag history of fully developed flow for various time step sizes. The reference solution and the twice refined solution agree qualitatively. With coarser time steps, we are unable to resolve the transient behaviour correctly. The table shows that results for t_{ref} and $t_{0.5}$ differ by 0.4%, which is acceptable for many applications. With larger time steps, the unsteadiness vanishes and mean values for lift and drag differ from the reference by more than 60%

5.1 Overall Accuracy: Angle of Attack

We first consider $\frac{\mathrm{d}J}{\mathrm{d}\alpha}$, which is the sensitivity of drag J with respect to changes in angle of attack α. This is the least sensitive sensor that we investigate, in the sense that it regards a time- and space-averaged result which allows temporal and spatial error modes to cancel out to some extent, see Table 1 for results.

Coarsening the primal temporal solution has a strong effect on the sensitivity results: relative errors rise above 60% for setup $t_8 \dots t_{256}$. In contrast to this, we observe that the sensitivity results produced with incomplete checkpointing are

Table 1 **Centre column**: $\frac{\mathrm{d}J}{\mathrm{d}\alpha}$ for different time step sizes. **Right column**: $\frac{\mathrm{d}J}{\mathrm{d}\alpha}$ for different gap sizes using incomplete checkpointing. The sensitivity is more dependent on the time step size than the primal result: The reference and $t_{0.5}$ sensitivities differ more than 6%, an order of magnitude more than the primal drag

step/gap	$\frac{\mathrm{d}J}{\mathrm{d}\alpha}$ for step	$\frac{\mathrm{d}J}{\mathrm{d}\alpha}$ for gap
0.5	**3**844.451056	*N/A*
1.0	**3612.203084**	**3612.203084**
2.0	2541.615522	**3612.2**35254
4.0	2227.642980	**3612**.317687
8.0	1504.326992	**361**5.459025
16.0	1288.293729	**3612**.068629
64.0	1262.006153	**36**06.972954
256.0	1228.272627	**3**364.129272

acceptable. To give an example from the table: the a_{64} setup only keeps every 64th time step and results in a sensitivity that differs from the reference by less than 0.2%. This is achieved with the same memory requirements as for the t_{64} setup, which has an error of over 60%. The a_4 result agrees with the reference result within 0.003%, but requires 75% less memory than the reference computation.

5.2 Spatial Accuracy: Surface Sensitivity

We investigate the spatial accuracy of the adjoint field by studying the sensitivity $\frac{\mathrm{d}J}{\mathrm{d}(\mathbf{x}\cdot\mathbf{n})}$ of the cost function with respect to normal displacements of airfoil surface nodes. This is based on the time-averaged adjoint field.

This sensor also shows that the temporal primal accuracy is crucial. There is a significant difference in surface sensitivities between the t_{ref} and $t_{0.5}$ setup, and an even larger one for coarser time steps. For setup t_8 and beyond, the sensitivity on

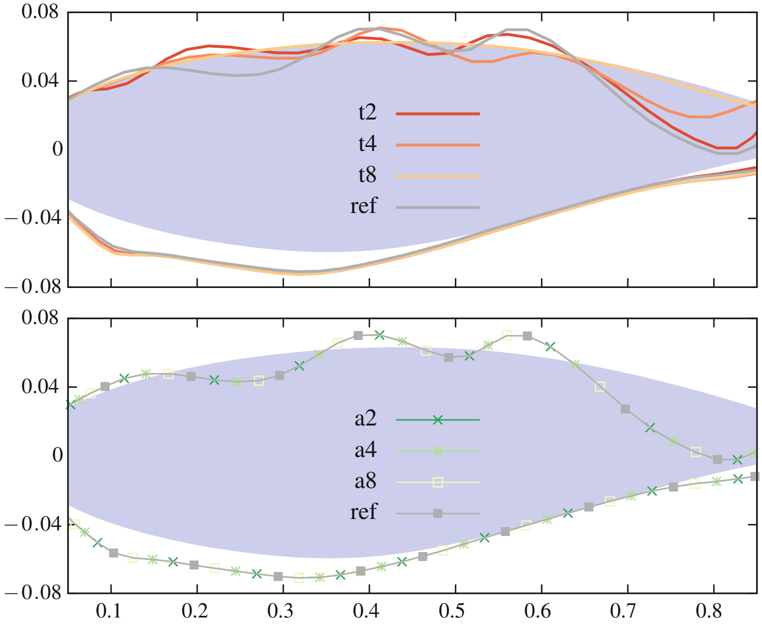


Fig. 3 Surface sensitivity scaled by a factor of $2e-4$ for plotting, superimposed on the airfoil surface. **Top**: Results for different primal time step sizes. The results agree qualitatively, although the sensitivity modes are shifted along the top surface. Reducing the temporal accuracy leads to a decrease in sensitivity. For t_8 and above, the surface sensitivity on the airfoil top is almost zero (i.e. is aligned with the current shape). The airfoil bottom shows a strong sensitivity which is resolved correctly regardless of time step size. **Bottom**: Different gap sizes for incomplete checkpointing. The sensitivities match that of the reference solution to plotting accuracy

the airfoil top vanishes completely, which is an indicator that the unsteadiness is no longer resolved. The surface sensitivity on the airfoil bottom is resolved correctly for large time step sizes and is the only contributor to the angle of attack sensitivity above $8 \cdot t_{ref}$.

Incomplete checkpointing results in relatively small errors for this case, see Fig. 3. The error grows somewhat with the gap size. Surprisingly, the a_{16} setup is more accurate than the a_8 setup in this case, see Fig. 4. Looking at the spatial distribution of errors, we find that errors are highest on the airfoil top and several orders of magnitude smaller at the airfoil bottom, see Fig. 4. This is due to the more intense unsteadiness above the airfoil (Fig. 1).

5.3 Temporal Accuracy: Flow Control

Finally, we study the sensitivity of average drag with respect to momentum injection close to the leading edge by means of a flow control valve, the location of which is shown in Fig. 5. The time window in which the drag is averaged is illustrated in Fig. 6. We observe that the adjoint momentum field for all time step and gap sizes is zero after the end of the cost function window, which is to be expected: An injection of momentum at any given time will not affect the drag in the past.

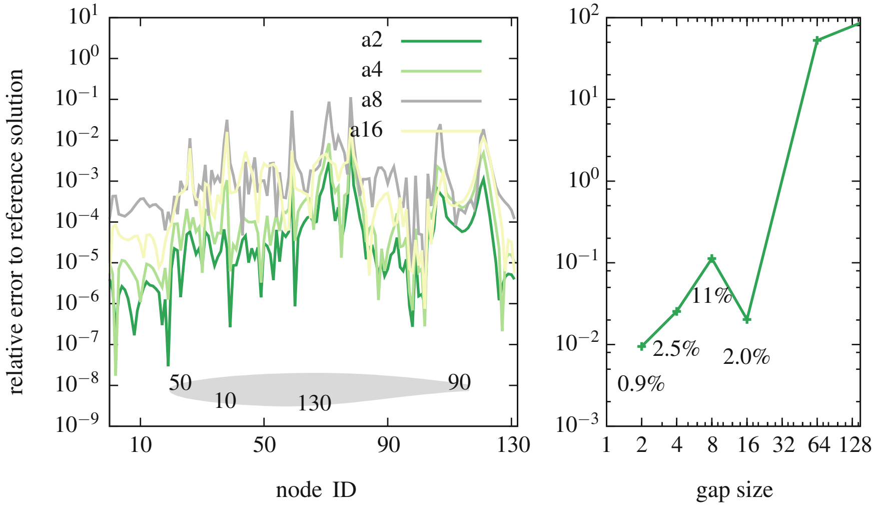


Fig. 4 **Left**: Surface sensitivity error along the airfoil surface. Node 1 is at the bottom centre, all other nodes are numbered continuously and clockwise as shown in the bottom of the left plot. Errors are larger on the airfoil top (nodes 50 to 95) than at the bottom (nodes 105 to 132 and 1 to 20). **Right**: Maximum relative error of surface sensitivity for different gap sizes. The relative error for $a_2 \ldots a_{16}$ is labeled and surprisingly smaller for a_{16} than for a_8

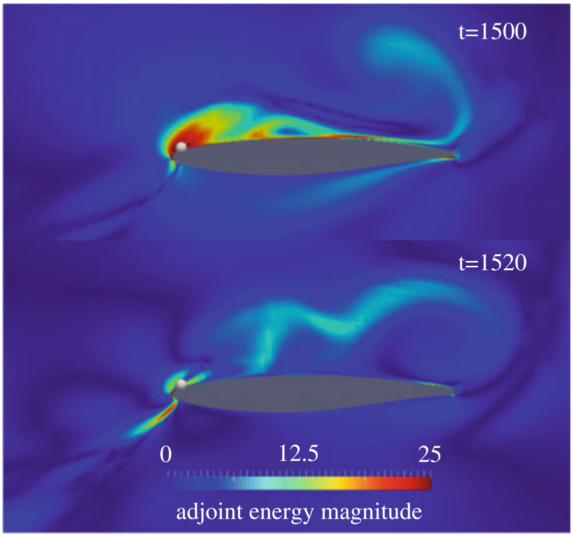


Fig. 5 Adjoint energy magnitude field after 1.5 and 1.52 s. The flow control valve is marked with a sphere above the leading edge. The flow control sensitivity results are based on a design variable that is the momentum injection rate in airfoil surface tangential direction at this location

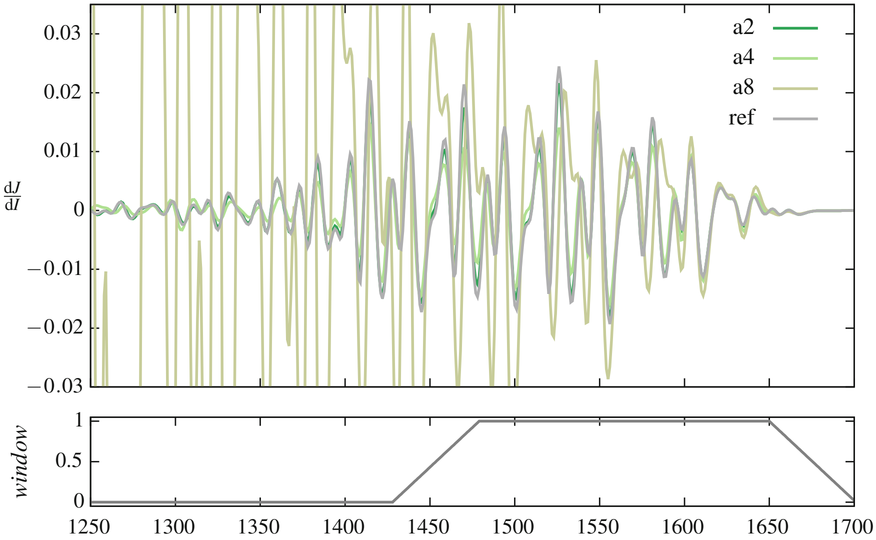


Fig. 6 **Bottom**: cost function averaging time window ω. **Top**: Sensitivity of drag with respect to momentum injection above the leading edge, plotted over time for a series of incomplete checkpointing gap sizes. We observe a zero adjoint field for time step 1700 and above, as any change happening after the averaging window can not affect the cost function. The sensitivity peaks for a time frame that is slightly longer than the cost averaging window and shifted towards lower time step numbers. The a_8 setup is oscillatory with a growing amplitude towards earlier time steps

If the flow is chaotic [18] or almost chaotic [16] the sensitivities can start to diverge as the adjoint field is computed backwards in time. For some simpler test cases we can expect the adjoint field to converge to zero as we proceed backwards in time from the beginning of the cost averaging window. This behaviour can be observed for our test case and is shown in Fig. 6. The sensitivity is largest for all points in time from which an injection of momentum could be propagated to some point of the airfoil surface during the cost function averaging time.

For gap sizes 8 and above, we find that the sensitivity does not settle down as we proceed backwards in time, and instead starts to oscillate with an exponentially growing magnitude. Surprisingly, this growing error mode canceled out in the time-averaging process that we used for the angle of attack and surface sensitivity studies.

In contrast to this, a coarsening of primal temporal resolution leads to a decrease in transient oscillations. Due to the highly nonlinear nature of the primal flow, it is impractical to compare the results of different time step or gap sizes directly for a given point in time, since even a minor change in the frequency of oscillations introduced by a change in time step size can accumulate over time, see Fig. 7. Hence, we compare the range in which the sensitivity oscillates over time.

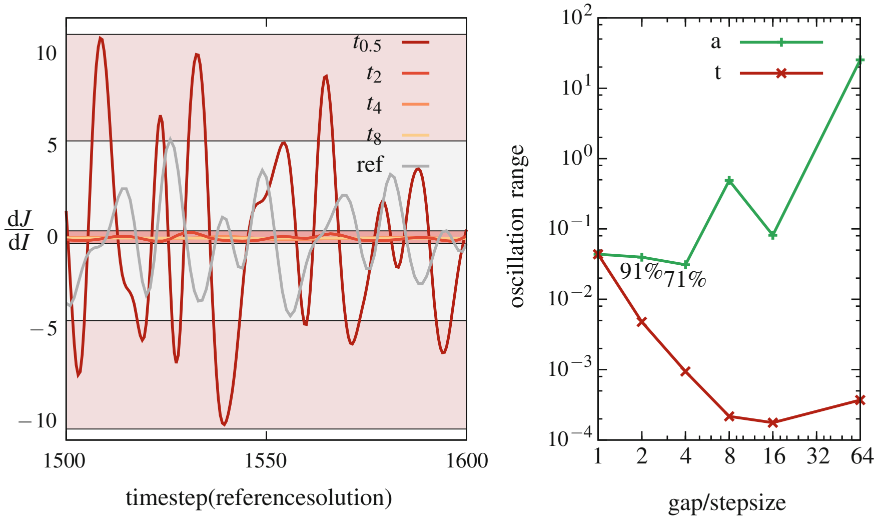


Fig. 7 **Left**: Time history of flow control sensitivities for the reference, $t_{0.5}$ and $t_2 \ldots t_8$ setups. The oscillation range for the finest 3 setups are highlighted with background colour. Due to the nonlinear behaviour of the primal flow, sensitivity values at any particular time are highly dependent on the temporal resolution. **Right**: Range of oscillation (difference between minimum and maximum value over time) for a range of time step and gap sizes. Coarsening the time steps removes temporal oscillations in the adjoint field, while incomplete checkpointing with increasing gap sizes leads to stronger oscillations as the gap size becomes very large. The a_2 and a_4 setups reduce the oscillation range to 91% and 71% of the original range, respectively

6 Conclusion, Possible Extensions

We show that gaps in the stored time trajectory are an easy and effective way of reducing the memory footprint of unsteady adjoint calculations. The effect on the sensitivity accuracy is acceptable for many industrial cases even for relatively large gap sizes, making our approach worth considering as an alternative to lossy checkpoint compression, with a significantly smaller implementation effort and computational cost.

In particular, the error introduced by storing an incomplete trajectory is much smaller than the error introduced by under-resolving the physical time during the primal flow computation. It is therefore preferable to perform the primal simulation with a fine temporal resolution and to use incomplete checkpointing, compared to the alternative of reducing the number of primal time steps to a number that fits into memory.

An error estimation strategy for incomplete checkpointing would be useful to choose the gap size and interpolation order. This could be used to adapt the checkpoint storing interval during the primal simulation and the restoration order during the adjoint simulation dynamically, e.g. to capture some flow features with a higher accuracy.

Finally, a similar method could be implemented for spatial coarsening, using interpolation or coarse grid multigrid solutions if available.

Acknowledgements This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no [317006].
This research utilised Queen Mary's MidPlus computational facilities, supported by QMUL Research-IT and funded by EPSRC grant EP/K000128/1.

References

1. Berggren, M.: Numerical solution of a flow-control problem: vorticity reduction by dynamic boundary action. SIAM J. Sci. Comput. **19**(3), 829–860 (1998)
2. Christakopoulos, F., Jones, D., Müller, J.D.: Pseudo-timestepping and verification for automatic differentiation derived CFD codes. Comput. Fluids **46**(1), 174 – 179 (2011). https://doi.org/10.1016/j.compfluid.2011.01.039
3. Christianson, B.: Reverse accumulation and implicit functions. Optim. Methods Softw. **9**(4), 307–322 (1998). https://doi.org/10.1080/10556789808805697
4. Giles, M., Pierce, N.: Adjoint equations in CFD—duality, boundary conditions and solution behaviour. Am. Inst. Aeronaut. Astronaut. (1997). https://doi.org/10.2514/6.1997-1850
5. Griewank, A., Walther, A.: Algorithm 799: Revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation. ACM Trans. Math. Softw. **26**(1), 19–45 (2000). https://doi.org/10.1145/347837.347846
6. Gunzburger, M.: Adjoint equation-based methods for control problems in incompressible, viscous flows. In: Flow, Turbulence and Combustion, vol. 65, pp. 249–272. Kluwer Academic Publishers (2000). https://doi.org/10.1023/A:1011455900396

7. Hascoet, L., Pascual, V.: The Tapenade automatic differentiation tool: Principles, model, and specification. ACM Trans. Math. Softw. **39**(3), 20:1–20:43 (2013). https://doi.org/10.1145/2450153.2450158
8. Heuveline, V., Walther, A.: Online checkpointing for parallel adjoint computation in PDEs: application to goal-oriented adaptivity and flow control. In: Euro-Par 2006 Parallel Processing, pp. 689–699. Springer (2006)
9. Hückelheim, J., Xu, S., Gugala, M., Müller, J.D.: Time-averaged steady vs. unsteady adjoint: a comparison for cases with mild unsteadiness. In: 53rd AIAA Aerospace Sciences Meeting. American Institute of Aeronautics and Astronautics (2015). https://doi.org/10.2514/6.2015-1953
10. Jameson, A.: Aerodynamic shape optimization using the adjoint method. In: Lectures at the Von Karman Institute, Brussels (2003)
11. Krakos, J.A., Darmofal, D.L.: Effect of small-scale output unsteadiness on adjoint-based sensitivity. AIAA J. **48**(11), 2611–2623 (2010). https://doi.org/10.2514/1.J050412
12. Lee, B.J., Liou, M.S.: Unsteady adjoint approach for design optimization of flapping airfoils. AIAA J. **50**(11), 2460–2475 (2012). https://doi.org/10.2514/1.J051663
13. Nadarajah, S., Jameson, A.: Optimum shape design for unsteady three-dimensional viscous flows using a nonlinear frequency-domain method. J. Aircr. **44**(5), 1513–1527 (2007). https://doi.org/10.2514/1.27601
14. Ratanaworabhan, P., Ke, J., Burtscher, M.: Fast lossless compression of scientific floating-point data. In: Proceedings of the Data Compression Conference, DCC '06, pp. 133–142. IEEE Computer Society, Washington, DC, USA (2006). https://doi.org/10.1109/DCC.2006.35
15. Rumpfkeil, M., Zingg, D.: A general framework for the optimal control of unsteady flows with applications. In: 45th AIAA Aerospace Sciences Meeting and Exhibit. American Institute of Aeronautics and Astronautics (2007). https://doi.org/10.2514/6.2007-1128
16. Schneider, T.M., Eckhardt, B., Yorke, J.A.: Turbulence transition and the edge of chaos in pipe flow. Phys. Rev. Lett. **99**, 034, 502 (2007). https://doi.org/10.1103/PhysRevLett.99.034502
17. Tim Wildey, E.C.C., Shadid, J.: Adjoint based a posteriori error estimates using data compression. In: Moitinho de Almeida, C.T.J.P., Díez, P., Parés, N. (eds.) VI International Conference on Adaptive Modeling and Simulation (2013)
18. Wang, Q., Hu, R., Blonigan, P.: Least squares shadowing sensitivity analysis of chaotic limit cycle oscillations. J. Comput. Phys. **267**(0), 210–224 (2014). https://doi.org/10.1016/j.jcp.2014.03.002
19. Xu, S., Jahn, W., Müller, J.D.: CAD-based shape optimisation with CFD using a discrete adjoint. Int. J. Numer. Methods Fluids **74**(3), 153–168 (2014)
20. Xu, S., Radford, D., Meyer, M., Müller, J.D.: Stabilisation of discrete steady adjoint solvers. J. Comput. Phys. **299**, 175–195 (2015)

Shape Optimization of Wind Turbine Blades Using the Continuous Adjoint Method and Volumetric NURBS on a GPU Cluster

Konstantinos T. Tsiakas, Xenofon S. Trompoukis, Varvara G. Asouti and Kyriakos C. Giannakoglou

Abstract This paper presents the development and application of the continuous adjoint method for the shape optimization of wind turbine blades aiming at maximum power output. A RANS solver, coupled with the Spalart-Allmaras turbulence model, is the flow (primal) model based on which the adjoint system of equations is derived. The latter includes the adjoint to the turbulence model equation. The primal and adjoint fields are used for the computation of the objective function gradient w.r.t. the design variables. A volumetric Non-Uniform Rational B-Splines (NURBS) model is used to parameterize the shape to be designed. The latter is also used for deforming the computational mesh at each optimization cycle. In order to reduce the computational cost, the aforementioned tools, developed in the CUDA environment, run on a cluster of Graphics Processing Units (GPUs) using the MPI protocol. Optimized GPU memory handling and GPU dedicated algorithmic techniques make the overall optimization process up to 50x faster than the same process running on a CPU. The developed software is used for the shape optimization of an horizontal axis wind turbine blade for maximum power output.

1 Introduction

Wind turbines design, and in particular their blade shapes, is a major application field in CFD. Though CFD methods are widely used for the aerodynamic analysis

K. T. Tsiakas (✉) · X. S. Trompoukis · V. G. Asouti (✉) · K. C. Giannakoglou
Parallel CFD & Optimization Unit, School of Mechanical Engineering,
National Technical University of Athens, Athens, Greece
e-mail: tsiakost@gmail.com

V. G. Asouti
e-mail: vasouti@mail.ntua.gr

X. S. Trompoukis
e-mail: xeftro@gmail.com

K. C. Giannakoglou
e-mail: kgianna@central.ntua.gr

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_8

of wind turbines [3], their use in shape optimization optimization of their bladings is still limited. The major drawback of CFD based optimization is its computational cost, especially when dealing with turbulent flows around complex geometries. The huge meshes (with millions of nodes) needed for the aerodynamic analysis of wind turbine blades make the use of stochastic, population-based optimization methods rather prohibitive. An alternative is the use of gradient-based optimization methods, such as steepest descent or quasi-Newton methods. In such a case, the computation of the gradient of the objective function is required. To do so, the adjoint method can be used and this makes the cost of computing the gradient independent of the number of design variables and approximately equal to that for solving the primal equations.

Over and above to any gain from the use of the less costly methods to compute the objective function gradient, a good way to reduce the optimization turnaround time is by accelerating the solution of the primal and adjoint equation using GPUs. Both the flow and adjoint solvers are ported on GPUs, exhibiting a noticeable speed-up compared to their CPU implementations [1, 4]. Though the use of a modern GPU can greatly accelerate CFD computations, its memory capacity is limited compared to a modern CPU RAM, posing a limitation when using GPUs for industrial applications. To overcome this problem, many GPUs, on different computational nodes if necessary, can be used to perform the computation in parallel, by making use of the CUDA environment together with the MPI protocol.

The geometry of wind turbine blades is quite complex, consisting of airfoil profiles varying largely along the spanwise direction. As a result, employing a scheme that parameterizes the exact geometry of the blade and incorporating it within the optimization process is not an easy task. Here, a volumetric NURBS model is used to parameterize the space around the blade over and above of the blade itself [5]. This model additionally undertakes mesh deformation, which would have to be carried out by a different method if a direct surface parameterization model was used. The main cost of the parameterization model is the computation of the B-Spline basis functions and their derivatives, which are herein required for the objective function gradient, according to the chain rule. In order to reduce this cost, their computation is also carried out on the GPUs.

The aforementioned methods and the corresponding software is applied for the shape optimization of the blades of a horizontal axis wind turbine.

2 Navier-Stokes, Adjoint Equations and Sensitivity Derivatives

The flow model is based on the incompressible flow equations using the Spalart-Allmaras turbulence model. The derivation of the adjoint equations along with the discretization of the resulting equations follows.

2.1 Flow (Primal) Equations

The flow equations used are the incompressible Navier-Stokes equations by applying the pseudo-compressibility approach, [2]. In order to predict the flow around the rotating blades in steady state, a multiple reference frame technique is used, where the equations are solved in a moving frame for the absolute velocity components. The flow equations read

$$R_{U_n} = \frac{\partial f_{nk}^{inv}}{\partial x_k} - \frac{\partial f_{nk}^{vis}}{\partial x_k} + S_n = 0 \tag{1}$$

where $U_n = [p \ \upsilon_1^A \ \upsilon_2^A \ \upsilon_3^A]^T$ is the vector of the state variables, $\upsilon_i^A, i = 1, 2, 3$ are the absolute velocity components and p is the pressure divided by the density. The inviscid and viscous fluxes f_{nk} and source terms S_n are given as

$$f_{nk}^{inv} = \begin{bmatrix} \beta \upsilon_k^R \\ \upsilon_k^R \upsilon_1^A + p\delta_{1k} \\ \upsilon_k^R \upsilon_2^A + p\delta_{2k} \\ \upsilon_k^R \upsilon_3^A + p\delta_{3k} \end{bmatrix}, \quad f_{nk}^{vis} = \begin{bmatrix} 0 \\ \tau_{1k} \\ \tau_{2k} \\ \tau_{3k} \end{bmatrix}, \quad S_n = \begin{bmatrix} 0 \\ \varepsilon_{1mk}\omega_m \upsilon_k^A \\ \varepsilon_{2mk}\omega_m \upsilon_k^A \\ \varepsilon_{3mk}\omega_m \upsilon_k^A \end{bmatrix} \tag{2}$$

with ω the blade rotational velocity and the stresses are

$$\tau_{mk} = (\nu + \nu_t)\left(\frac{\partial \upsilon_m^A}{\partial x_k} + \frac{\partial \upsilon_k^A}{\partial x_m}\right)$$

where ν and ν_t stand for the kinematic and turbulent viscosity. In Eq. 2, υ_i^R denote the relative velocity components. The absolute and relative velocity vectors are linked through $\upsilon_i^A = \upsilon_i^R - \upsilon_i^F$, with $\upsilon_i^F = \varepsilon_{ijk}\omega_j d_k$ and $d_k = x_k - x_k^C$ are the components of the position vector from the origin (x_k^C) which lies on the rotation axis.

Equations 1 are solved together with the Spalart-Allmaras turbulence model PDE ($R_{\tilde{\nu}} = 0$, [9]) according to a segregated time-marching scheme.

2.2 Continuous Adjoint Formulation

For the wind turbine application under consideration, the objective function F is the power output of the turbine blading for constant rotational velocity. Its maximization is, in fact, equivalent to that of the torque w.r.t. the axis of the wind turbine shaft. If r_k denotes the components of the unit vector aligned with the shaft, F can be expressed as

$$F = \int_{S_{Blade}} \varepsilon_{klm}\left(x_l - x_l^C\right)\left(pn_m - \tau_{mq}n_q\right) r_k dS \tag{3}$$

where S_{Blade} denotes the blade surface. In Eq. 3, n_q are the components of the unit vector normal to the blade surface, pointing towards the blade.

By introducing the adjoint mean-flow variables Ψ_n $(n = 1, \ldots, 4)$ and the adjoint turbulent variable $\tilde{\nu}^a$, the augmented objective function is defined as

$$F_{aug} = F + \int_{\Omega} \Psi_n R_{U_n} d\Omega + \int_{\Omega} \tilde{\nu}^a R_{\tilde{\nu}} d\Omega \tag{4}$$

Upon convergence of the primal equations, F_{aug} is equal to F. To compute the variations of F_{aug} w.r.t. the design variables b_i, we start by differentiating Eq. 4, which yields

$$\frac{\delta F_{aug}}{\delta b_i} = \frac{\delta F}{\delta b_i} + \frac{\delta}{\delta b_i} \int_{\Omega} \Psi_n R_{U_n} d\Omega + \frac{\delta}{\delta b_i} \int_{\Omega} \tilde{\nu}^a R_{\tilde{\nu}} d\Omega \tag{5}$$

By developing and eliminating the integrals including the variations in the flow quantities w.r.t. b_i, the field adjoint equations and their boundary conditions arise. The remaining integrals form the expression of the gradient of F w.r.t. b_i. The field adjoint equations read

$$R_{\Psi_n} = -\underbrace{A_{mnk}\frac{\partial \Psi_m}{\partial x_k}}_{Conv(\Psi)} - \underbrace{\frac{\partial \phi_{nk}^{vis}}{\partial x_k}}_{Diff(\Psi)} - \underbrace{S_n^{adj}}_{Source1(\Psi)} + \underbrace{T_n^{adj}}_{Source2(\tilde{\nu}_a)} = 0 \tag{6}$$

with

$$A_{nmk} = \begin{bmatrix} 0 & \beta\delta_{1k} & \beta\delta_{2k} & \beta\delta_{3k} \\ \delta_{1k} & \upsilon_1^R + \upsilon_1^A\delta_{1k} & \upsilon_1^A\delta_{2k} & \upsilon_1^A\delta_{3k} \\ \delta_{2k} & \upsilon_2^A\delta_{1k} & \upsilon_2^R + \upsilon_2^A\delta_{2k} & \upsilon_2^A\delta_{3k} \\ \delta_{3k} & \upsilon_3^A\delta_{1k} & \upsilon_3^A\delta_{2k} & \upsilon_3^R + \upsilon_3^A\delta_{3k} \end{bmatrix}$$

$$S_n^a = \begin{bmatrix} 0 \\ \varepsilon_{1mk}\omega_m\Psi_{k+1} \\ \varepsilon_{2mk}\omega_m\Psi_{k+1} \\ \varepsilon_{3mk}\omega_m\Psi_{k+1} \end{bmatrix} \qquad \phi_{nk}^{vis} = \begin{bmatrix} 0 \\ \tau_{1k}^a \\ \tau_{2k}^a \\ \tau_{3k}^a \end{bmatrix}$$

where δ_{ij} is the Kronecker's symbol and

$$\tau_{mk}^a = (\nu + \nu_t)\left(\frac{\partial \Psi_{m+1}}{\partial x_k} + \frac{\partial \Psi_{k+1}}{\partial x_m}\right) \tag{7}$$

are the adjoint stresses.

In Eq. 6, the terms marked as $Conv(\Psi)$ and $Diff(\Psi)$ correspond to the adjoint convection and diffusion respectively, $Source1(\Psi)$ corresponds to the adjoint source terms resulting from the frame rotation and $Source2(\widetilde{\nu^a})$ includes the contribution of the adjoint turbulence model to the adjoint mean–flow equations. The derivation of the adjoint turbulence model equation can be found in a previous work [10] published from the same group and will not be repeated here.

After solving the primal and adjoint equations, $\frac{\delta F}{\delta bi}$ can be computed once the geometric sensitivities $\frac{\delta x_l}{\delta b_i}$ and $\frac{\partial}{\partial x_k}\left(\frac{\delta x_l}{\delta b_i}\right)$ at the mesh nodes become available. The final expression of the sensitivity derivatives (gradF) reads

$$\begin{aligned}\frac{\delta F}{\delta b_i} = &\int_{S_{Blade}} \varepsilon_{klm} p n_m r_k \frac{\delta x_l}{\delta b_i} dS + \int_{S_{Blade}} \varepsilon_{klm} \left(x_l - x_l^C\right) p r_k \frac{\delta}{\delta b_i}(n_m dS) - \\ &\int_{S_{Blade}} \varepsilon_{klm} \tau_{mq} n_q r_k \frac{\delta x_l}{\delta b_i} dS - \int_{S_{Blade}} \varepsilon_{klm} \left(x_l - x_l^C\right) \tau_{mq} r_k \frac{\delta}{\delta b_i}(n_m dS) + \\ &T^{MF} + T^{SA}\end{aligned} \tag{8}$$

where the terms T^{MF} and T^{SA} correspond to the differentiation of the flow equations and the turbulence model respectively. These terms are herein omitted in the interest of space. The reader may find them in [10] and this is irrelevant to the objective function used.

2.3 Discretization and Numerical Solution

The primal and adjoint equations are discretized on hybrid meshes (consisting of tetrahedra, pyramids, prisms or hexahedra) using the vertex-centered finite volume method and solved using a time-marching scheme. The numerical fluxes crossing the finite volume interfaces are computed with second-order accuracy. The primal inviscid numerical flux crossing the interface between nodes P and Q reads

$$\Phi^{PQ} = \frac{1}{2}\left(f_{nk}^{inv,P} + f_{nk}^{inv,Q}\right) n_k^{PQ} - \frac{1}{2}\left|\overline{A}_{nmk}^{PQ} n_k\right| \left(U_m^R - U_m^L\right)$$

where n_k^{PQ} are the components of the unit vector normal to the finite volume interface between nodes P and Q and pointing to node Q and the Jacobian $\overline{A}^{PQ}$ is computed based on the Roe-averaged [7] flow variables. U^R and U^L are flow variables on the right and left sides of the finite volume interface, obtained by extrapolating U^Q and U^P, respectively.

On the other side, the adjoint inviscid numerical fluxes are computed using a non-conservative scheme

$$\Phi_n^{adj,PQ} = -\frac{1}{2} A_{mnk}^P \left(\Psi_n^P + \Psi_n^Q\right) n_k - \frac{1}{2}\left|\overline{A}_{mnk}^{PQ} n_k\right| \left(\Psi_n^R - \Psi_n^L\right)$$

$$\Phi_n^{adj,QP} = \frac{1}{2} A_{mnk}^Q \left(\Psi_n^P + \Psi_n^Q\right) n_k + \frac{1}{2}\left|\overline{A}_{mnk}^{PQ} n_k\right| \left(\Psi_n^R - \Psi_n^L\right)$$

For the viscous fluxes, the derivatives of any primal flow or adjoint quantity W on the finite volumes interface (between nodes P and Q) are computed as

$$\left(\frac{\partial W}{\partial x_k}\right)^{PQ} = \overline{\left(\frac{\partial W}{\partial x_k}\right)} - \left[\overline{\left(\frac{\partial W}{\partial x_m}\right)} t_m^{PQ} - \frac{W^Q - W^P}{\sqrt{(x_m^Q - x_m^P)^2}}\right] t_k^{PQ} \tag{9}$$

where

$$t_m^{PQ} = \frac{x_m^Q - x_m^P}{\sqrt{(x_m^Q - x_m^P)^2}}, \quad \overline{\left(\frac{\partial W}{\partial x_k}\right)} = \frac{1}{2}\left[\left(\frac{\partial W}{\partial x_k}\right)^P + \left(\frac{\partial W}{\partial x_k}\right)^Q\right]$$

The discretized equations are linearized and solved iteratively w.r.t. the correction of the primal/adjoint variables using a point-implicit Jacobi method.

3 Parameterization Through Volumetric NURBS

Volumetric NURBS are rational trivariate (in 3D) B-Splines defined on non-uniform knot vectors, used to parameterize the volume around the blade. Let (ξ, η, ζ) be the three parametric directions and X_m^{ijk} and w^{ijk} the (ijk)th control point coordinates and weight. Given the parametric coordinates of a point as well as the knot vectors and control points coordinates/weights, its physical coordinates $x_m (m = 1, 2, 3)$ can be computed as

$$x_m(\xi, \eta, \zeta) = \frac{\sum_i^{N_\xi} \sum_j^{N_\eta} \sum_k^{N_\zeta} \Xi_{i,p_\xi}(\xi) H_{j,p_\eta}(\eta) Z_{k,p_\zeta}(\zeta) X_m^{ijk} w^{ijk}}{\sum_i^{N_\xi} \sum_j^{N_\eta} \sum_k^{N_\zeta} \Xi_{i,p_\xi}(\xi) H_{j,p_\eta}(\eta) Z_{k,p_\zeta}(\zeta) w^{ijk}} \tag{10}$$

where, Ξ_{i,p_ξ} is the ith B-Spline basis function of degree p_ξ defined on the knot vector $K_\xi = \{\xi_0, \ldots, \xi_{m_\xi}\}$ (H_{j,p_η} and Z_{k,p_ζ} are defined similarly), N_ξ is the number of control points in the ξ direction and it must hold that $m_\xi = N_\xi + p_\xi + 1$ [6]. Knots must be arranged in non-decreasing order.

Specifying the control points, weights and knot vectors, a point inversion, via the Newton-Raphson method, is used to compute the parametric coordinates of the mesh nodes. The so-computed parametric coordinates as well as the knot vectors remain fixed during the optimization. All variations in geometric quantities, such as $\frac{\delta x_l}{\delta b_i}$ and $\frac{\partial}{\partial x_k}\left(\frac{\delta x_l}{\delta b_i}\right)$, involved in the computation of gradF are given by closed-form expressions resulting from the differentiation of Eq. 10.

During the optimization loop, the control point coordinates and weights are updated and Eq. 10 is used to deform the computational mesh and blade shape.

4 Implementation on GPUs

Nowadays, GPUs have become powerful parallel co-processors to CPUs, offering more than one order of magnitude more floating point operations per second (FLOPS) with lower memory latency compared to modern CPUs.

Although the GPU hardware capabilities are superior to the CPU ones, directly porting a CPU code on a GPU does not necessarily yields the desired high speed-ups, due to different architecture features. The Navier-Stokes/adjoint equations solver this paper makes use of, efficiently exploits the high computing capabilities that modern GPUs have, running on a GPU at least 50 times faster than the equivalent CPU solver. Such a high parallel efficiency mainly results from (a) the use of Mixed Precision Arithmetics (MPA), which allows the l.h.s. matrices to be computed using double-precision and stored using single-precision arithmetics [4], without harming the accuracy of the solver and (b) the minimization of random accesses to the relatively high latency device memory by concurrently running threads.

For maximum speed-up, the primal and adjoint solvers employ different algorithmic techniques for the computation of the nodal residuals and l.h.s. coefficients. In previous work by the authors [1], it is shown that, when processing large amount of data on a GPU, minimizing memory usage and non-coalesced memory accesses is more important than minimizing the number of (rather redundant) re-computations of the same quantity. Thus, the primal solver, in which the memory consuming Jacobians per finite volume interface need to be computed for the l.h.s. coefficients at each pseudo-iteration, uses a one-kernel scheme. According to this scheme, a single kernel is launched, associating each GPU thread with a mesh node. Each thread computes and accumulates the numerical fluxes crossing all boundaries of this node's finite volume and their Jacobians and, thus, forms residuals and l.h.s. coefficients. On the contrary, the l.h.s. coefficients in the adjoint system of equations depend only on the primal solution, the Jacobians are computed once, before the iterative solution of the adjoint equations. Thus, the adjoint solver employs a two-kernel scheme in which the less memory consuming adjoint numerical fluxes are computed by the first kernel (GPU threads associated with finite volume interfaces) and accumulated by the second kernel (GPU threads associated with mesh nodes).

The primal/adjoint solvers run on a cluster of GPUs. In order to run a case in many GPUs, the mesh is partitioned in overlapped sub-domains and each sub-domain is associated with one GPU. For instance, Fig. 1 (left) shows a triangular mesh generated around an isolated airfoil partitioned in three overlapped sub-domains. The shared regions of the mesh sub-domains are marked in white in Fig. 1. The whole mesh (i.e. including the overlapped regions) of the 3rd sub-domain, with the boundaries shared with sub-domains 1, 2, can be seen in Fig. 1 (right). To further reduce the wall-clock time, computations and data transfers overlap. For instance, when computing the primal/adjoint spatial gradients, each GPU associated with a sub-domain performs the same sequence of steps. As an example, the GPU associated with the 3rd sub-domain performs the following steps:

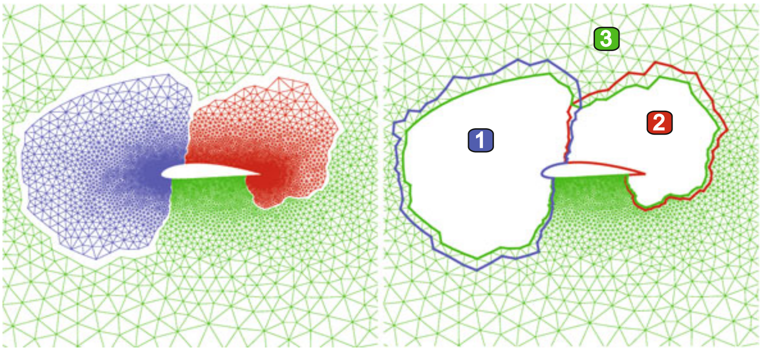


Fig. 1 Mesh with triangular elements around an isolated airfoil partitioned in three overlapped subdomains

Step A: Launches a kernel only for the computation of the gradients at the interface nodes with sub-domains 1 and 2 (i.e. nodes lying on the blue and red lines of Fig. 1 (right)).

Step B: Performs the data interchange between the sub-domains (assigned to different GPUs).

Step C: Launches a kernel for the computation of the gradients at the remaining nodes of the sub-domain.

Steps A, B are performed simultaneously with step C so that computations and data transfers overlap. Data transfers among GPUs on different computational nodes use the MPI protocol. The communication of GPUs on the same node is performed through the shared (on-node) CPU memory.

The computations of the parametric coordinates of the mesh nodes and the objective function gradients, which are computationally intensive and memory demanding, also run on GPUs. Since $\frac{\delta x_l}{\delta b_i}$, which is needed for $\frac{\delta F}{\delta b_i}$, are geometric quantities independent of the primal/adjoint solution, they could be computed and stored just once. However, the memory needed for storing $\frac{\delta x_l}{\delta b_i}$ often exceeds that required for the solution of the primal and adjoint equations. Hence, their storage is avoided and they are re-computed at the end of each optimization cycle using pre-allocated GPU memory.

The optimization flowchart is shown in Fig. 2. Steps performed exclusively on CPU or GPU are clearly marked. Expensive processes associated with the computation/update of the mesh geometrical data, such as computing node distances from the nearest wall, are performed on the GPU, while others such as computing the cells volumes are performed at the same time on the CPU. Thus, all available computing resources are exploited and the wall clock time needed to perform these tasks is reduced.

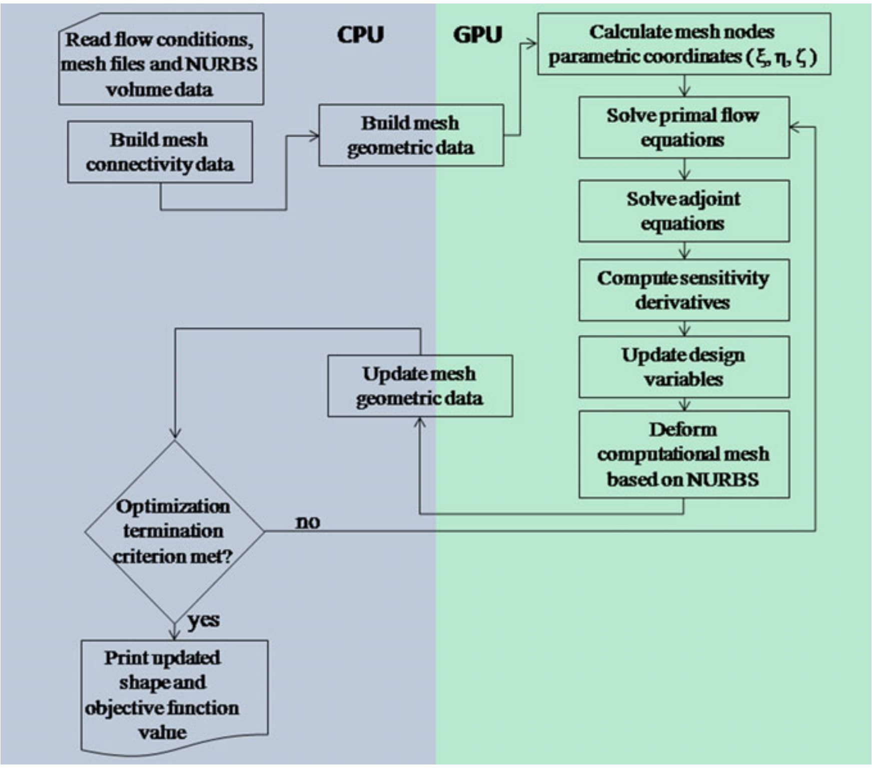

CPU
GPU
Read flow conditions, mesh files and NURBS volume data
Build mesh connectivity data
Build mesh geometric data
Calculate mesh nodes parametric coordinates (ξ, η, ζ)
Solve primal flow equations
Solve adjoint equations
Compute sensitivity derivatives
Update design variables
Deform computational mesh based on NURBS
Update mesh geometric data
Optimization termination criterion met?
no
yes
Print updated shape and objective function value

Fig. 2 Flowchart demonstrating the optimization algorithm steps. Steps performed on the CPU and GPU are distinguished

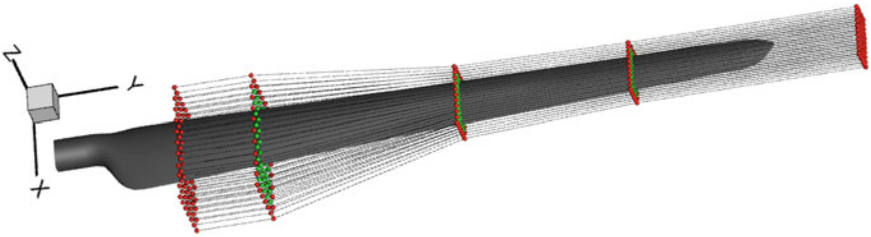


Fig. 3 Parameterization of the HAWT blade and the surrounding mesh through volumetric NURBS. Control points in red color are kept fixed during the optimization to ensure G^0 continuity with the surrounding undeformed/unparameterized mesh. The green control points are allowed to move during the optimization

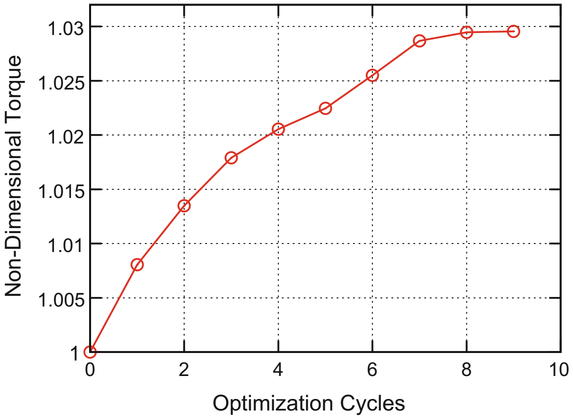


Fig. 4 Optimization convergence history. On the vertical axis, the objective function (power output to be maximized) is divided by the value this function takes on for the starting blade geometry

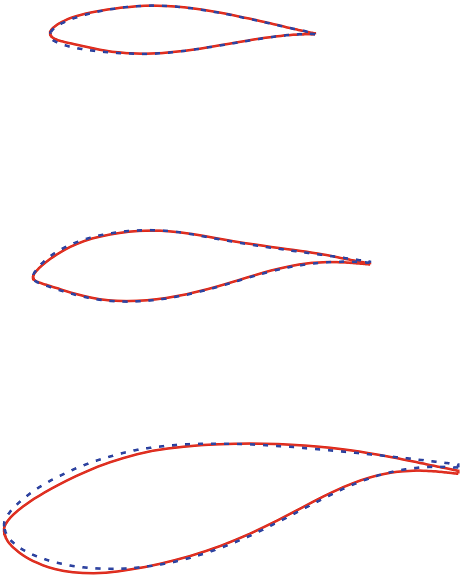


Fig. 5 Comparison of the optimized blade profile (solid/red line) with the starting (dashed/blue) at 35, 60 and 82% (from bottom to top) of the wind turbine blade span

5 Optimization of the Wind Turbine Blade

The developed software described in the previous sections, was used for the shape optimization of the MEXICO [8] horizontal axis wind turbine (HAWT) blade for maximum power output, when operating at $10\frac{m}{s}$ farfield velocity and 0° yaw angle. For the parameterization of the blade, a $11 \times 5 \times 7$ NURBS control volume is used, as shown in Fig. 3. All boundary control points are kept fixed in order to ensure G^0 continuity while the remaining ones are allowed to move along the z axis (Fig. 3) leading to 135 ($9 \times 3 \times 5$) design variables in total. The computational mesh consists of about 2.5×10^6 nodes and both the primal and adjoint solvers run on 4 NVIDIA

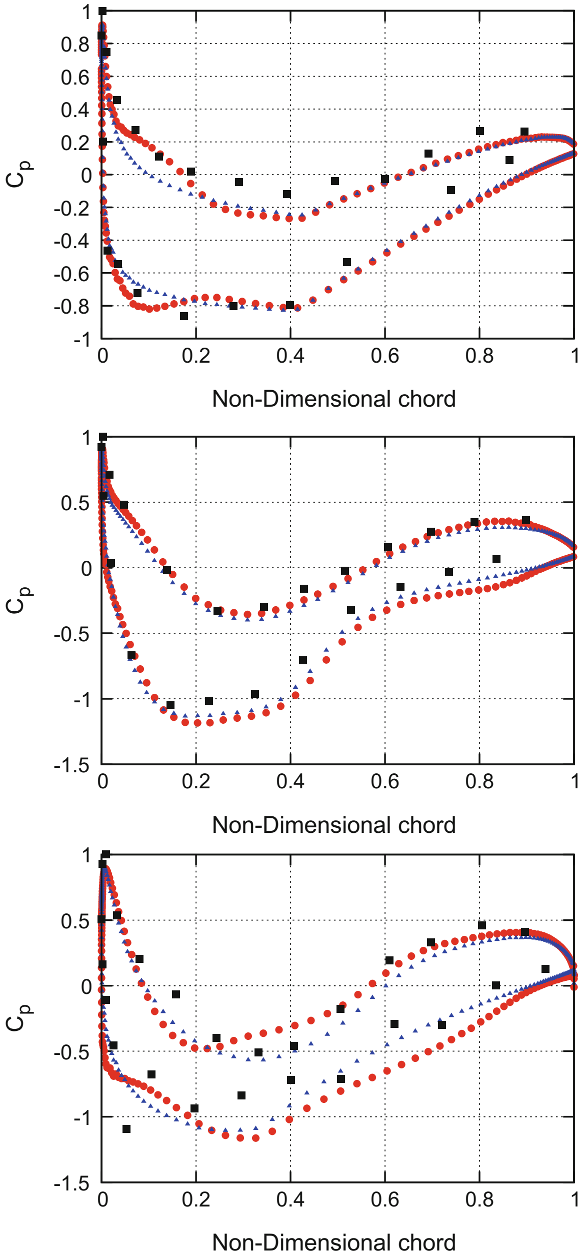


Fig. 6 Comparison of the pressure coefficient for the starting (blue triangles) and the optimized blade (red circles), along with the available experimental data (black squares) on the starting geometry, [8] at 35, 60 and 82% (from bottom to top) of the wind turbine blade span. The pressure coefficient is defined as $c_p = \frac{p-p_{far}}{\frac{1}{2}(V_{far}^2+\omega^2 R^2)}$, with R the local radius and far indexing farfield flow quantities

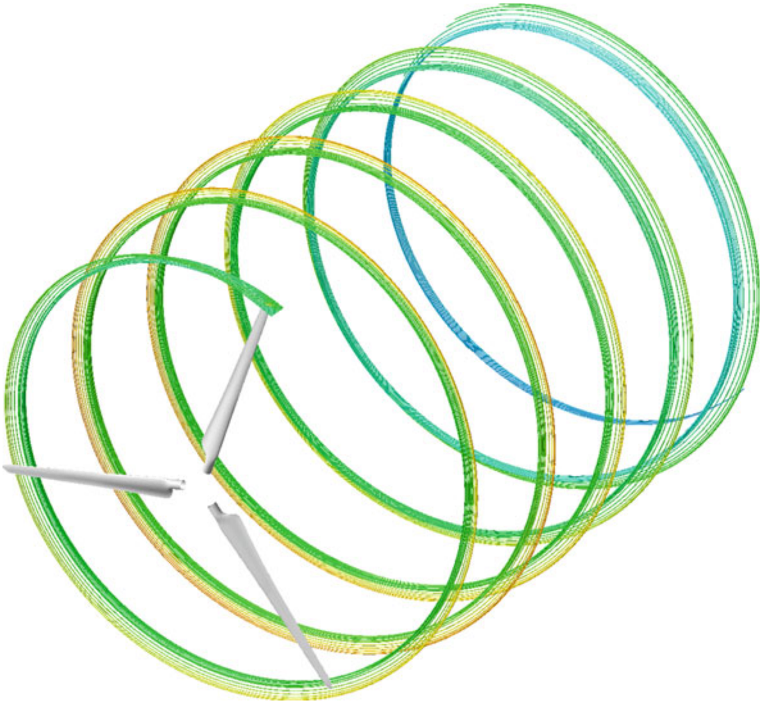


Fig. 7 Relative velocity streamlines (coloured based on the relative velocity magnitude) in the tip vortex region

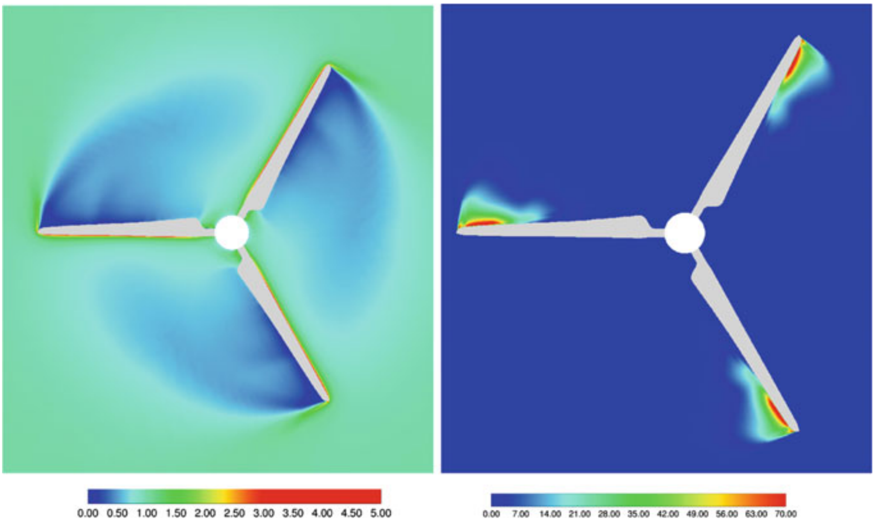


Fig. 8 Axial velocity (left) and turbulent viscosity (right) in a transversal slice through the wind turbine origin. The velocity values are normalized with respect to the farfield velocity magnitude

Kepler K20 GPUs, lying on two different nodes. On this platform each optimization cycle needs less than 2 h, 1 h for the solution of the primal and 45 min for the adjoint equations to fully converge. The convergence history of the optimization is shown in Fig. 4. The optimized blade yields 3% increased torque compared to the reference blade. The improvement is minor due to the degrees of freedom used, i.e. the NURBS control points were allowed to move only in the z direction.

Since the differences between the optimized and the reference blade are not visible in a 3D surface comparison, the blade profiles at three spanwise positions of the blade are compared instead (Fig. 5).

Figure 6 presents the comparison of the chordwise distribution of the pressure coefficient for the starting and the optimized blade, along with the experimental results (from [8]) for the same spanwise positions. It is clear from Fig. 5 that the shape of the blade airfoil changes significantly in the lower part of the blade. However, in Fig. 6, it is shown that the difference in blade loading between the reference and the optimized geometry is larger close to the tip, since this region has a major contribution in the torque generation. The relative velocity streamlines in the tip vortex region are plotted in Fig. 7.

Figure 8 shows the axial velocity and turbulent viscosity in a transversal slice through the wind turbine origin.

6 Conclusions

This paper presented the development and use of the continuous adjoint method for the shape optimization of a HAWT blade for maximum torque. Since wind turbine blades are complex geometries, parameterization was based on volumetric NURBS method, which also contributes to the mesh deformation at each optimization cycle. In order to reduce the optimization turnaround time, the solution of both the flow and the adjoint equations is carried out on 4 Nvidia Tesla K20 GPUs. In particular, each optimization cycle requires approximately 1 h for the primal and 45 min for the adjoint equations solution.

Acknowledgements This study has been co-financed by the European Union (European Social Fund-ESF) and Greek national funds through the Operational Program Education and Lifelong Learning of the National Strategic Reference Framework (NSRF) Research Funding Program: THALES. Investing in knowledge society through the European Social Fund.

References

1. Asouti, V., Trompoukis, X., Kampolis, I., Giannakoglou, K.: Unsteady CFD computations using vertex-centered finite volumes for unstructured grids on graphics processing units. Int. J. Num. Methods Fluids **67**(2), 232–246 (2011)

2. Chorin, A.: A numerical method for solving incompressible viscous flow problems. J. Comput. Phys. **2**(1), 12–26 (1967)
3. Hansen, M., Srensen, J., Voutsinas, S., Srensen, N., Madsen, H.: State of the art in wind turbine aerodynamics and aeroelasticity. Progress Aerosp. Sci. **42**(4), 285–330 (2006)
4. Kampolis, I., Trompoukis, X., Asouti, V., Giannakoglou, K.: CFD-based analysis and two-level aerodynamic optimization on graphics processing units. Comput. Methods Appl. Mech. Eng. **199**(9–12), 712–722 (2010)
5. Martin, M.J., Andres, E., Lozano, C., Valero, E.: Volumetric b-splines shape parametrization for aerodynamic shape design. Aerosp. Sci. Technol. **37**, 26–36 (2014)
6. Piegl, L., Tiller, W.: The NURBS Book, 2nd edn. Springer, New York (1997)
7. Roe, P.: Approximate Riemann solvers, parameter vectors, and difference schemes. J. Comput. Phys. **43**(2), 357–372 (1981)
8. Schepers, J., Snel, H.: Final report of IEA Task 29, Mexnext (Phase I): Analysis of Mexico wind tunnel measurements. Technical report, ECN (2012)
9. Spalart, P., Allmaras, S.: A one-equation turbulence model for aerodynamic flows. La Recherche Aérospatiale **1**, 5–21 (1994)
10. Zymaris, A., Papadimitriou, D., Giannakoglou, K., Othmer, C.: Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. Comput. Fluids **38**(8), 1528–1538 (2009)

Aerodynamic Shape Optimization Using the Adjoint-Based Truncated Newton Method

Evangelos M. Papoutsis-Kiachagias, Mehdi Ghavami Nejad and Kyriakos C. Giannakoglou

Abstract This paper presents the development and application of the truncated Newton (TN) method in aerodynamic shape optimization problems. The development is made for problems governed by the laminar flow equations of incompressible fluids. The method was developed in OpenFOAM© with the aim to stress its advantages over standard gradient-based optimization algorithms. The Newton equations are solved using the Conjugate Gradient (CG) method which requires the computation of the product of the Hessian of the objective function and a vector, escaping thus the need for computing the Hessian itself. The latter has a computational cost that scales with the number of design variables and becomes unaffordable in large-scale problems with many design variables. A combination of the continuous adjoint method and direct differentiation is used to compute all Hessian-vector products. A grid displacement PDE (Laplace equation) is also used to compute the necessary derivatives of grid displacements w.r.t. the design variables. The programmed method is used to optimize the sidewall shapes of $2D$ ducts for minimum total pressure losses.

1 Introduction to the Truncated Newton Method

An unconstrained optimization problem, in which the target is to minimize the objective function F by controlling the design variables $b_i, i = 1, \ldots, N$ can be solved by means of the Newton method, according to which the design variables are updated as follows

E. M. Papoutsis-Kiachagias (✉) · M. Ghavami Nejad (✉) · K. C. Giannakoglou
Parallel CFD & Optimization Unit, School of Mechanical Engineering,
National Technical University of Athens, Athens, Greece
e-mail: vaggelisp@gmail.com

M. Ghavami Nejad
e-mail: mehdi@mail.ntua.gr; mehgh750@student.liu.se

K. C. Giannakoglou
e-mail: kgianna@central.ntua.gr

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_9

$$b_i^{n+1} = b_i^n + \delta b_i \tag{1a}$$

$${\frac{\delta^2 F}{\delta b_i \delta b_j}}^{n} \delta b_j = -{\frac{\delta F}{\delta b_i}}^{n} \tag{1b}$$

where n is the Newton iteration counter, to be omitted hereafter. The direct solution of Eq. 1b requires the computation of the Hessian of F, with computational cost that scales with N.

Considering Eq. 1b as a linear system of equations of the form $Ax = q$, a possible way to solve it is through the Conjugate Gradient (CG) method, which is schematically given in Algorithm 1.

Algorithm 1 : The CG Method for the Solution of $Ax = q$

```
$m \leftarrow 0$
$x \leftarrow \text{init}()$
$r^m \leftarrow Ax - q;\ s \leftarrow -r^m$
while $r^m \neq 0$, (CG Iterations) do
    $\eta \leftarrow \frac{(r^m)^T r^m}{s^T As}$
    $x \leftarrow x + \eta s$
    $r^{m+1} \leftarrow r^m + \eta As$
    $\beta \leftarrow \frac{(r^{m+1})^T r^{m+1}}{(r^m)^T r^m}$
    $s \leftarrow -r^{m+1} + \beta s$
    $m \leftarrow m + 1$
end while
```

Based on Algorithm 1, the cost of each CG iteration is dominated by the cost of computing the matrix–vector product (As). In the so–called truncated variant (Truncated Newton, TN, [3]), the stopping criterion in Algorithm 1 becomes $m \leq M_{CG}$, where M_{CG} is a user-defined small integer (to be used instead of $r^m \neq 0$). Regarding Eq. 1b, since the Hessian matrix stands for A, the use of the TN method in aerodynamic shape optimization problems means that the Hessian matrix itself is no more needed and only its product with a vector must be computed. On the other hand, the gradient of F must be available and the (continuous) adjoint method, [4], can be used for this. Its major advantage is its ability to compute the exact gradient of any function, at CPU cost which is independent of N.

2 The Continuous Adjoint Method for the Computation of $\frac{\delta F}{\delta b_i}$

The continuous adjoint method, [4], starts by differentiating the objective function F augmented by the field integral of the flow equations multiplied by the so–called adjoint fields, in order to derive the adjoint PDEs. The latter are, then, discretized and numerically solved to compute the adjoint fields. The gradient of F is expressed in the form of field or boundary integrals of quantities involving the previously computed flow and adjoint fields.

Let us assume a $2D$ laminar flow of an incompressible fluid governed by the continuity ($R^p = 0$) and the momentum ($R_i^v = 0$) equations, where

$$R^p = -\frac{\partial v_j}{\partial x_j} \tag{2}$$

$$R_i^v = v_j \frac{\partial v_i}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial p}{\partial x_i}\,, \qquad i = 1, 2 \tag{3}$$

Here, v_i are the velocity components, p the static pressure divided by the constant density, $\tau_{ij} = \nu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$ the stress tensor and ν the constant viscosity.

Without loss in generality, let us assume that the objective function F to be minimized is the volume-averaged total pressure losses (for the flow inside a duct; internal aerodynamics), namely

$$F = \int_{S_{I,O}} F_{S,i} n_i dS, \qquad F_{S,i} = -\left(p + \frac{1}{2} v_k^2 \right) v_i \tag{4}$$

where $S = S_I \cup S_O \cup S_W$ is the domain boundary with S_I being the inlet, S_O the outlet and S_W the wall boundary and $\mathbf{n}$ the outward unit normal vector on the surface. Recall that, for any flow quantity Φ, the total derivative $\delta \Phi / \delta b_n$, which represents the total change in Φ caused by variations in b_n, is

$$\frac{\delta \Phi}{\delta b_n} = \frac{\partial \Phi}{\partial b_n} + \frac{\partial \Phi}{\partial x_k} \frac{\delta x_k}{\delta b_n} \tag{5}$$

In Eq. 5, the partial derivative $\partial \Phi / \partial b_n$ represents only the variation in Φ caused due to changes in the design and flow variables, without considering space deformations.

Then, the differentiation of F w.r.t. b_n gives

$$\frac{\delta F}{\delta b_n} = \int_S \frac{\partial F_{S,i}}{\partial b_n} n_i dS + \int_S \frac{\partial F_{S,i}}{\partial x_k} \frac{\delta x_k}{\delta b_n} n_i dS + \int_S F_{S,i} \frac{\delta \left(n_i dS \right)}{\delta b_n} \tag{6}$$

The development of the augmented objective function

$$F_{aug} = F + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega \tag{7}$$

leads to the adjoint continuity ($R^q = 0$) and adjoint momentum ($R_i^u = 0$) equations,

$$R^q = -\frac{\partial u_j}{\partial x_j} \tag{8}$$

$$R_i^u = u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial \left(u_i v_j\right)}{\partial x_j} - \frac{\partial \tau_{ij}^a}{\partial x_j} + \frac{\partial q}{\partial x_i}\ ,\ i = 1, 2 \tag{9}$$

where $\tau_{ij}^a = \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$ is the adjoint stress tensor. By satisfying Eqs. 8 and 9, all field integrals in $\delta F_{aug}/\delta b_n$ which depend on $\delta v_i/\delta b_n$ and $\delta p/\delta b_n$ are eliminated. The adjoint boundary conditions are derived by eliminating the total derivatives of the flow variables along the boundaries, while also considering the flow boundary conditions. In this paper, we will refrain from further developing the adjoint boundary conditions, see [4].

After satisfying the adjoint PDEs, Eqs. 8 and 9, the expression for the gradient of F is

$$\frac{\delta F}{\delta b_n} = \int_{\Omega} A_{jk} \frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega \tag{10}$$

where

$$A_{jk} = -u_i v_j \frac{\partial v_i}{\partial x_k} - u_j \frac{\partial p}{\partial x_k} - \tau_{ij}^a \frac{\partial v_i}{\partial x_k} + u_i \frac{\partial \tau_{ij}}{\partial x_k} + q \frac{\partial v_j}{\partial x_k} \tag{11}$$

3 Computation of Hessian(F)–Vector Products

As explained in Sect. 1, the TN method requires the computation of $\frac{\delta^2 F}{\delta b_n \delta b_m} s_m$, where s_m might be the components of any vector.

Let us use overbar to denote the product of the total gradient $\frac{\delta \Phi}{\delta b_m}$ of any quantity Φ and s_m, namely

$$\overline{\Phi} = \frac{\delta \Phi}{\delta b_m} s_m \tag{12}$$

It can be proved that

$$\overline{\frac{\partial \Phi}{\partial x_j}} = \frac{\delta}{\delta b_m}\left(\frac{\partial \Phi}{\partial x_j}\right) s_m = \frac{\partial \overline{\Phi}}{\partial x_j} - \frac{\partial \Phi}{\partial x_k}\frac{\partial \overline{x_k}}{\partial x_j} \tag{13}$$

Also, for any pair of Φ and Ψ,

$$\frac{\delta}{\delta b_m}\left(\Psi \frac{\partial \Phi}{\partial x_j}\right) s_m = \overline{\Psi}\frac{\partial \Phi}{\partial x_j} + \Psi \frac{\partial \overline{\Phi}}{\partial x_j} - \Psi \frac{\partial \Phi}{\partial x_k}\frac{\partial \overline{x_k}}{\partial x_j} \tag{14}$$

Based on the above, it is a matter of mathematical development to show that

$$\begin{aligned}\frac{\delta^2 F}{\delta b_n \delta b_m} s_m = &\int_\Omega \overline{A_{jk}} \frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega + \int_\Omega A_{jk} \frac{\delta}{\delta b_m}\left[\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\right] s_m d\Omega \\ &+ \int_\Omega A_{jk} \frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) s_m \frac{\delta(d\Omega)}{\delta b_m}\end{aligned} \tag{15}$$

where

$$\begin{aligned}\overline{A_{jk}} = &-\overline{u_i} v_j \frac{\partial v_i}{\partial x_k} - u_i \overline{v_j} \frac{\partial v_i}{\partial x_k} - u_i v_j \overline{\frac{\partial v_i}{\partial x_k}} + u_i v_j \frac{\partial v_i}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_k} - \overline{u_j} \frac{\partial p}{\partial x_k} - u_j \frac{\partial \overline{p}}{\partial x_k} \\ &+ u_j \frac{\partial p}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_k} - \nu \left(\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i}\right) \frac{\partial v_i}{\partial x_k} + \nu \left(\frac{\partial u_i}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_j} + \frac{\partial u_j}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_i}\right) \frac{\partial v_i}{\partial x_k} \\ &- \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \frac{\partial \overline{v_i}}{\partial x_k} + \nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \frac{\partial v_i}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_k} \\ &+ \overline{u_i} \frac{\partial}{\partial x_k} \left[\nu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] + u_i \frac{\partial}{\partial x_k} \left[\nu \left(\frac{\partial \overline{v_i}}{\partial x_j} + \frac{\partial \overline{v_j}}{\partial x_i}\right)\right] \\ &- u_i \frac{\partial}{\partial x_k} \left[\nu \left(\frac{\partial v_i}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_j} + \frac{\partial v_j}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_i}\right)\right] - u_i \frac{\partial}{\partial x_\lambda} \left[\nu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] \frac{\partial \overline{x_\lambda}}{\partial x_k} \\ &+ \overline{q} \frac{\partial v_j}{\partial x_k} + q \frac{\partial \overline{v_j}}{\partial x_k} - q \frac{\partial v_j}{\partial x_\lambda} \frac{\partial \overline{x_\lambda}}{\partial x_k}\end{aligned} \tag{16}$$

and, [2],

$$\frac{\delta(d\Omega)}{\delta b_m} s_m = \frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_\lambda}{\delta b_m} s_m\right) d\Omega = \frac{\partial \overline{x_\lambda}}{\partial x_\lambda} d\Omega \tag{17}$$

since $\overline{x_\lambda} = \frac{\delta x_\lambda}{\delta b_m}$. By denoting

$$\overline{\overline{x_{k,n}}} = \frac{\delta^2 x_k}{\delta b_n \delta b_m} s_n \tag{18}$$

it can be proved that

$$\int_\Omega A_{jk} \frac{\delta}{\delta b_m}\left[\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right)\right] s_m d\Omega = \int_\Omega A_{jk} \frac{\partial \overline{\overline{x_{k,n}}}}{\partial x_j} d\Omega - \int_\Omega A_{jk} \frac{\partial}{\partial x_\lambda}\left(\frac{\delta x_k}{\delta b_n}\right) \frac{\partial \overline{x_\lambda}}{\partial x_j} d\Omega \tag{19}$$

4 Computation of $\overline{v_i}$ and $\overline{p}$

Computing $\overline{v_i}$ and $\overline{p}$ is straightforward and can be done by formulating the product of the direct differentiation (DD, i.e. derivation w.r.t. b_n) of the flow equations and s_m. It is

$$\overline{R^p} = \frac{\delta R^p}{\delta b_m} s_m = 0 \quad , \qquad \overline{R_i^v} = \frac{\delta R_i^v}{\delta b_m} s_m = 0 \tag{20}$$

where

$$\overline{R^p} = \frac{\partial \overline{v_j}}{\partial x_j} - \frac{\partial v_j}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_j} \tag{21}$$

and

$$\begin{aligned} \overline{R_i^v} =& \frac{\partial (\overline{v_i} v_j)}{\partial x_j} + \frac{\partial (v_i \overline{v_j})}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\nu \left(\frac{\partial \overline{v_i}}{\partial x_j} + \frac{\partial \overline{v_j}}{\partial x_i}\right)\right] + \frac{\partial \overline{p}}{\partial x_i} \\ & - \frac{\partial (v_i v_j)}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_j} + \frac{\partial}{\partial x_j}\left[\nu \left(\frac{\partial v_i}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_j} + \frac{\partial v_j}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_i}\right)\right] \\ & + \frac{\partial}{\partial x_k}\left[\nu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right] \frac{\partial \overline{x_k}}{\partial x_j} - \frac{\partial p}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_i} \end{aligned} \tag{22}$$

5 Computation of $\overline{u_i}$ and $\overline{q}$

Similarly, the product of the DD of the adjoint equations and s_m yields

$$\overline{R^q} = \frac{\delta R^q}{\delta b_m} s_m = 0 \quad , \qquad \overline{R_i^u} = \frac{\delta R_i^u}{\delta b_m} s_m = 0 \tag{23}$$

where

$$\overline{R^q} = \frac{\partial \overline{u_j}}{\partial x_j} - \frac{\partial u_j}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_j} \tag{24}$$

and

$$\begin{aligned} \overline{R_i^u} =& \overline{u_j} \frac{\partial v_j}{\partial x_i} + u_j \frac{\partial \overline{v_j}}{\partial x_i} - \frac{\partial (\overline{u_i} v_j)}{\partial x_j} - \frac{\partial (u_i \overline{v_j})}{\partial x_j} \\ & - \frac{\partial}{\partial x_j}\left[\nu \left(\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i}\right)\right] + \frac{\partial \overline{q}}{\partial x_i} - u_j \frac{\partial v_j}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_i} \\ & + \frac{\partial (v_j u_i)}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_j} + \frac{\partial}{\partial x_j}\left[\nu \left(\frac{\partial u_i}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_j} + \frac{\partial u_j}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_i}\right)\right] \\ & + \frac{\partial}{\partial x_k}\left[\nu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \frac{\partial \overline{x_k}}{\partial x_j} - \frac{\partial q}{\partial x_k} \frac{\partial \overline{x_k}}{\partial x_i} \end{aligned} \tag{25}$$

6 Computation of $\overline{x_k}$ and $\overline{\overline{x_{k,n}}}$

In aerodynamic shape optimization problems, a widely used grid displacement model, i.e. a mathematical model that propagates variations in the boundary shape to the internal computational nodes, is based on the Laplace equation with Dirichlet boundary conditions. Written for the derivatives of the grid coordinates X_K w.r.t. the design variables, it takes the form

$$R_i^x = \frac{\partial^2}{\partial x_j^2}\left(\frac{\delta x_k}{\delta b_n}\right) = 0 \tag{26}$$

from which it can readily be deduced that

$$\frac{\partial^2 \overline{x_k}}{\partial x_j^2} = 0 \tag{27}$$

It can also be proved that

$$\frac{\partial^2 \overline{\overline{x_{k,n}}}}{\partial x_j^2} = 2\frac{\partial^2}{\partial x_j \partial x_\lambda}\left(\frac{\delta x_k}{\delta b_m}\right)\frac{\partial \overline{x_\lambda}}{\partial x_j} \tag{28}$$

which can numerically be solved to compute $\overline{\overline{x_{k,n}}}$ with appropriate boundary conditions depending also on the adopted parametrization model.

7 The TN Algorithm—Comments on the CPU Cost

Using Eqs. 17–19, Eq. 15 can be written as

$$\frac{\delta^2 F}{\delta b_n \delta b_m} s_m = \int_\Omega \left[\overline{A_{jk}} + A_{jk}\frac{\partial \overline{x_\lambda}}{\partial x_\lambda} - A_{\lambda k}\frac{\partial \overline{x_j}}{\partial x_\lambda}\right]\frac{\partial}{\partial x_j}\left(\frac{\delta x_k}{\delta b_n}\right) d\Omega$$
$$+ \int_\Omega A_{jk}\frac{\partial \overline{\overline{x_{k,n}}}}{\partial x_j} d\Omega \tag{29}$$

where $\overline{A_{jk}}$ is given by Eq. 16. To compute $\overline{A_{jk}}$, apart from the flow and adjoint fields, the "overbar" fields ($\overline{v_i}$, $\overline{u_i}$, $\overline{p}$, $\overline{q}$, as well as $\overline{x_i}$ and their spatial derivatives) must be available.

So, in each Newton cycle, the numerical solution of $R^p = 0$ and $R_i^v = 0$ (where R^p and R_i^v are given by Eqs. 2 and 3) yields the flow fields (p, v_i). The solution of $R^q = 0$ and $R_i^u = 0$ (where R^q and R_i^u are given by Eqs. 8 and 9) yields the adjoint fields (q, u_i). So, far, the computational cost is approximately equal to that of twice solving the flow equations or 2 EFS (EFS stands for Equivalent Flow Solutions, i.e. a way to measure computational cost).

Before solving for $\overline{p}$ and $\overline{v_i}$, $\overline{x_k}$ must be computed by solving Eq. 27 at the cost of 1 GDS (GDS stands for Grid Displacement Solutions, i.e.the cost of solving the grid displacement PDE or any of the PDEs that result from its differentiation). It should be mentioned that the cost of 1 GDS is significantly lower than the cost of 1 EFS. Equation 27 has to be solved once per CG iteration, contributing a total cost of M_{CG} GDS per optimization cycle.

Computing $\overline{p}$ and $\overline{v_i}$ requires the numerical solution of Eqs. 20 (considering also Eqs. 21 and 22). Similarly, to compute $\overline{q}$ and $\overline{u_i}$ requires the numerical solution of Eqs. 23 (considering also Eqs. 24 and 25). Both systems of equations should be performed within the CG loop (i.e. M_{CG} times) and contribute $2M_{CG}$ EFS to the overall cost of a Newton iteration or cycle.

Within each CG iteration, the computation of $\overline{A_{jk}}$ also requires the availability of the $\delta x_k/\delta b_n$ and $\overline{\overline{x_{k,n}}}$ fields. To this end, Eqs. 26 and 28 must be solved for $n \in [1, N]$. This results to $2M_{CG}N$ GDS per optimization cycle.

Based on the above, the overall CPU cost per Newton iteration is equal to $2 + 2M_{CG}$ EFS and $1 + 2M_{CG}N$ GDS. However, since the cost of a GDS is significantly lower than that of an EFS, the GDS part can be considered negligible for a moderate number of design variables. This leads to a cost per Newton cycle that is independent of the number of design variables N.

8 Applications

In this section, two applications of the developed TN optimization algorithm are presented.

The first one deals with the shape optimization of an S-bend duct. The flow is laminar with a Reynolds number of $Re = 785$ based on the inlet height and a mesh consisting of 27, 500 quadrilaterals is used. Each of the upper and lower sides are

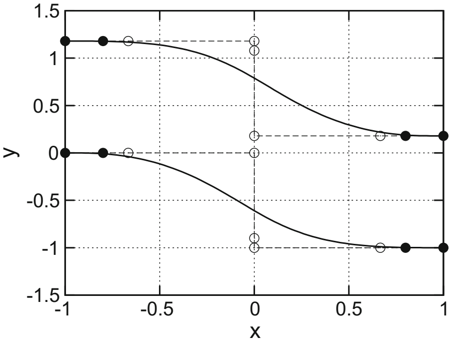


Fig. 1 S-bend duct optimization: dust shape and the Bézier–Bernstein control points parameterizing it. Axes not in scale. Control points depicted with a dark cycle remain fixed during the optimization

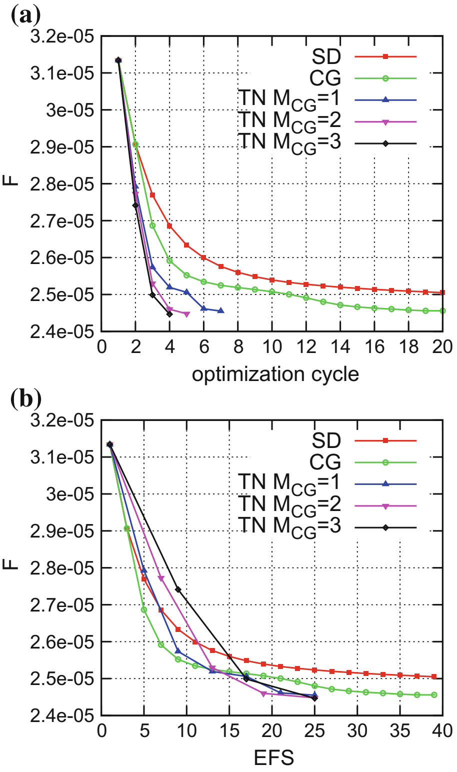


Fig. 2 S-bend duct optimization: Convergence of the steepest descent (SD), Conjugate Gradient (CG) and Truncated Newton (TN) optimization algorithms, w.r.t. optimization cycles (**a**) and EFS (**b**)

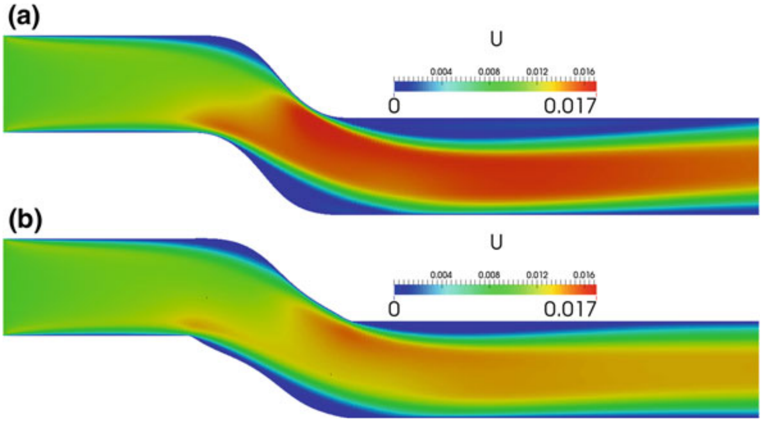


Fig. 3 S-bend duct optimization: Velocity magnitude for the initial (**a**) and optimized (**b**) geometries

parameterized using 9 Bézier–Bernstein control points, Fig. 1. The first and last two control points per side are kept fixed while the x and y coordinates of the rest are allowed to vary, giving rise to a total of 20 design variables. In Fig. 2, the convergence history of the developed TN algorithm is compared to those of steepest descent (SD) and the Fletcher-Rives Conjugate Gradient (CG), [1], method. Both the iterations required to reach a minimum and the corresponding EFS are compared. In addition, an investigation of the effect of the M_{CG} number can be seen in the same figures. It can be observed that TN outperforms SD and CG, since it computes the optimized duct shape using less optimization cycles and, especially, by requiring less EFS. In addition, it can be seen that even though increasing the M_{CG} number reduces the number of optimization cycles required to reach the minimum, there is no obvious gain from the EFS point of view. In Fig. 3, the flow velocity magnitude in the initial and optimized ducts is presented.

The second case is concerned with the optimization of a divergent duct. The flow Reynolds number is $Re = 475$ and a mesh consisting of 20000 quadrilaterals is used.

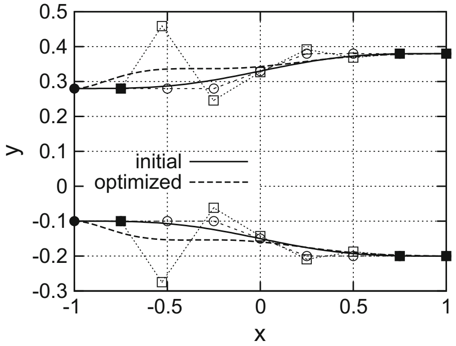


Fig. 4 Divergent duct optimization: dust shape and the Bézier–Bernstein control points parameterizing it. Axes not in scale. Control points depicted with a dark cycle remain fixed during the optimization

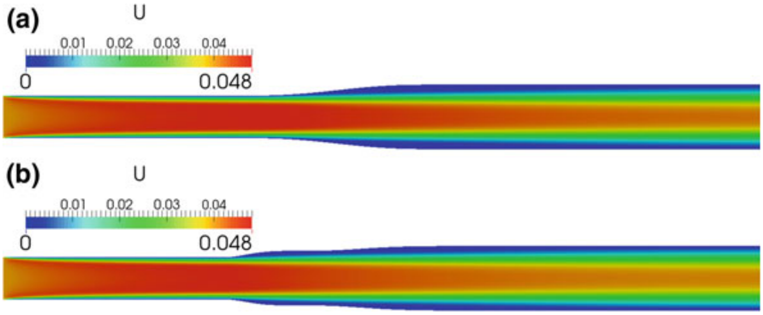
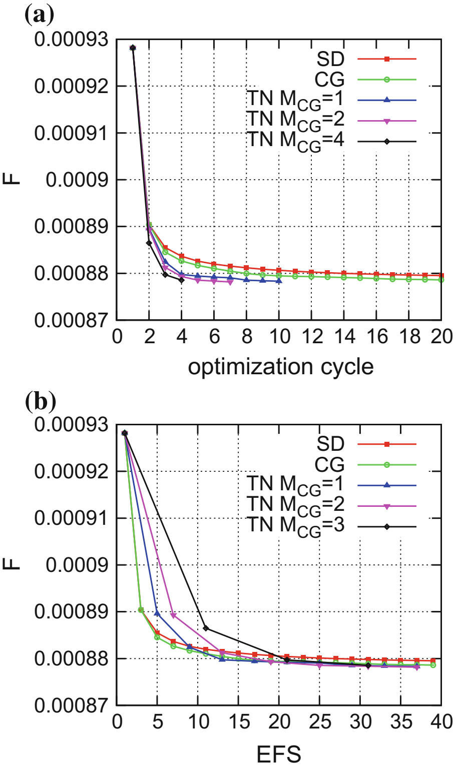


Fig. 5 Divergent duct optimization: Velocity magnitude for the initial (**a**) and optimized (**b**) geometries

Fig. 6 Divergent duct optimization: Convergence of the steepest descent (SD), Conjugate Gradient (CG) and Truncated Newton (TN) optimization algorithms, w.r.t. optimization cycles (**a**) and EFS (**b**)



The initial and optimized geometries along with the Bézier–Bernstein control points used to parameterize the duct shape are depicted in Fig. 4. In Fig. 5, the flow velocity magnitude in the initial and optimized ducts is presented. In Fig. 6, the convergence history of the TN, SD and CG are illustrated. In this case as well, TN outperforms SD and CG from the optimization cycles point of view; regarding EFS, TN and CG compute the optimal solution almost at the same cost. Increasing M_{CG} has the same effect as in the first case, i.e. the optimized geometry is computed in less optimization cycles but without a significant advantage in CPU cost. This seems to indicate that, at least for the cases studied, a low M_{CG} number should be chosen.

9 Conclusions

A Truncated Newton method for computing an approximation to the second-order correction of the design variables by iteratively solving Newton's equation using Conjugate Gradient was presented. The method built on previous work of the authors for Euler flows and extended the mathematical background for incompressible, laminar flows. The proposed Truncated Newton method computes the required Hessian-vector products by utilizing a combination of (continuous) adjoint and direct differentiation. The cost per optimization cycle is approximately equal to $2 + 2M_{CG}$ equivalent flow solutions, where M_{CG} is the number of CG iterations used to approximate the solution of Newton's equation; this cost is practically independent of the design variables number. In the two applications presented, each with a moderate number of design variables, it was shown that Truncated Newton outperforms other optimization methods in terms of optimization cycles and is, at least, as fast as Conjugate Gradient in terms of CPU cost. A parametric study for M_{CG} has also shown that its value should remain as low as possible.

Acknowledgements The second author is an Early Stage Researcher and his research is funded from the People Programme (ITN Marie Curie Actions) of the European Union's 7th Framework Programme (FP7/2007–2013) under REA grant agreement no 317006 (AboutFLOW project).

References

1. Fletcher, R., Reeves, C.M.: Function minimization by conjugate gradients. Comput. J. **7**, 149–154 (1964)
2. Papadimitriou, D., Giannakoglou, K.: A continuous adjoint method with objective function derivatives based on boundary integrals for inviscid and viscous flows. Comput. Fluids **36**, 325–341 (2007)
3. Papadimitriou, D., Giannakoglou, K.: Aerodynamic design using the truncated Newton algorithm and the continuous adjoint approach. Int. J. Num. Methods Fluids **68**, 724–739 (2012)
4. Papoutsis-Kiachagias, E., Giannakoglou, K.: Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: industrial applications. Arch. Comput. Methods Eng., 1–45 (2014)

Application of the Adjoint Method for the Reconstruction of the Boundary Condition in Unsteady Shallow Water Flow Simulation

Asier Lacasta, Daniel Caviedes-Voullieme and Pilar García-Navarro

Abstract Hydraulic phenomena in open-channel flows are usually described by means of the shallow water equations. This hyperbolic non-linear system can be used for predictive purposes provided that initial and boundary conditions are supplied and the roughness coefficient is calibrated. When calibration is required to fully pose the problem, several strategies can be adopted. In the present work, an inverse technique, useful for any of such purposes, based on the adjoint system and gradient descent is presented. It is used to find the optimal time evolution of the inlet boundary condition required to meet the 20 measured water depth data in an experimental test case of unsteady flow on a beach. The partial differential systems are solved using an upwind finite volume scheme. Several subsets of probes were selected and the quality of the reconstructed boundary tested against the experimental results. The results show that the adjoint technique is useful and robust for these problems, and exhibits some sensitivity to the choice of probes, which can be used to properly select probes in real applications.

1 Introduction

Shallow water equations (SWE) have been largely used for a wide range of applications related to hydraulic phenomena. Their ability to reproduce open-channel flow, for instance, makes them a very useful tool for engineering purposes. Nonetheless, their non linearity difficults calibration and their use for inverse design is not straightforward.

A. Lacasta (✉) · P. García-Navarro
LIFTEC, CSIC-University of Zaragoza, Calle María de Luna 3, 50012 Zaragoza, Spain
e-mail: alacasta@unizar.es

P. García-Navarro
e-mail: pigar@unizar.es

D. Caviedes-Voullieme
University of Sheffield, Broad Lane, Sheffield, S3 7HQ, UK
e-mail: d.caviedes@sheffield.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_10

In recent years, the application of adjoint equations as a technique for the control of PDEs has turned to be a good option [4]. Its application for inverse design of airfoil shapes in aeronautics is a good example of the capabilities of the technique.

The same idea may be applied for hydraulic phenomena. In [3, 7] the development of the adjoint formulation, as well as examples of application for control, were presented. Additionally, the application for calibration of the model can be found in [5].

In this work, the adjoint formulation of the one-dimensional SWE is applied to recover a boundary condition required to reproduce a laboratory experimental case of unsteady flow over a beach. In particular, there was no information related to the discharge or velocity of the flow at the seaside boundary, which made it impossible to accurately reproduce the experiment numerically. Therefore, the adjoint set of equations in combination with a gradient-descent optimizer is used to rebuild the boundary condition at the sea side.

2 Mathematical Model

The 1D shallow water equations are derived from the equations of conservation of mass and momentum when averaged over a cross section. They form a 2 × 2 hyperbolic system of equations, which, for an arbitrary channel with arbitrary cross sections is

$$\mathrm{G}_1 : \frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = 0 \tag{1}$$

$$\mathrm{G}_2 : \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right) + \frac{\partial}{\partial x}(gI_1) - g[I_2 + A(S_0 - S_f)] = 0$$

being (A, Q) the conserved variables that represent wetted area and discharge respectively. The friction S_f and bed slope S_0 contributions are

$$S_f = \frac{Q|Q|n^2}{A^2 R_h^{4/3}}, \qquad S_0 = -\frac{\partial z}{\partial x} \tag{2}$$

where n is Manning's roughness coefficient, $R_h =$ is the hydraulic radius and z is bed elevation and hydrostatic forces I_1 and pressure forces I_2:

$$I_1(x) = \int_0^h (h - \eta)\chi(x, \eta)\, d\eta = \frac{1}{2}h^2 \qquad I_2(x) = \int_0^h (h - \eta)\frac{\partial \chi(x, \eta)}{\partial x}\, d\eta = 0 \tag{3}$$

Herein, for simplicity, a prismatic and rectangular channel with unit width is considered, so that the conserved variables (A, Q) are transformed into (h, q), i.e., water-level and unit discharge respectively. Moreover, under such conditions, $I_1 = 0.5h^2$ and $I_2 = 0$.

2.1 Adjoint System

Control may be defined as the minimization of some functional $\mathbf{J}$, such as

$$\mathbf{J} = \int_0^T \int_0^L \mathscr{E}(x, t; h, q) dxdt \tag{4}$$

where, T is the maximum simulation time and L is the length of the channel. The error metric $\mathscr{E}$ may be defined by considering N_p measured water depths

$$\mathscr{E}(h, q) = \frac{1}{N_p} \sum_{i=1}^{N_p} (h_i - h_{obj,i})^2 \tag{5}$$

where h_i is simulated water depth and $h_{obj,i}$ is measured water depth at position x_i.

In order to obtain a suitable formulation to calculate sensitivities to, in turn perform control, the adjoint system of (1) is developed. An augmented functional is defined as $\mathbf{J}^+ = \mathbf{J} + \mathbf{P}$ where $\mathbf{P}$ may be obtained by multiplying both, mass (G_1) and momentum (G_2) equations by their adjoint variables (σ_1, σ_2)

$$\mathbf{P} = \int_0^T \int_0^L \sigma_1 \left(\mathrm{G}_1\right) + \sigma_2 \left(\mathrm{G}_2\right) dxdt \tag{6}$$

This expression can be integrated by parts leading to

$$\begin{gathered}
\mathbf{P} = \int_0^T \int_0^L -h\frac{\partial \sigma_1}{\partial t} dxdt + \int_0^L [\sigma_1 h]\Big|_0^T dx + \int_0^T \int_0^L -q\frac{\partial \sigma_1}{\partial x} dxdt + \int_0^T [\sigma_1 q]\Big|_0^L dt + \\
\textstyle\int_0^T \int_0^L - \frac{\partial \sigma_2}{\partial t} dxdt + \int_0^L [\sigma_2 q]\Big|_0^T dx + \int_0^T \int_0^L - \left(\frac{q^2}{h} + \frac{gh^2}{2}\right) \frac{\partial \sigma_2}{\partial x} dxdt + \\
\textstyle\int_0^T [\sigma_2 \left(\frac{q^2}{h} + \frac{gh^2}{2}\right)]\Big|_0^L dt - \int_0^T \int_0^L \sigma_2 gh(S_o - S_f) dxdt
\end{gathered} \tag{7}$$

Assuming stationary condition of the optimal solutions for the augmented functional $\delta\mathbf{J}^+ = \delta\mathbf{J} + \delta\mathbf{P} = 0$, the first variation $\delta\mathbf{P}$ of system $\mathbf{P}$ is formulated by taking increments with respect to h and q. The variation $\delta\mathbf{J}$ is also computed as

$$\delta\mathbf{J} = \int_0^T \int_0^L \left(\frac{\partial \mathscr{E}}{\partial h}\delta h + \frac{\partial \mathscr{E}}{\partial q}\delta q\right) dxdt \tag{8}$$

When the following system holds

$$\begin{gathered}
-\frac{\partial \sigma_1}{\partial t} + \left(\frac{q^2}{h^2} - gh\right) \frac{\partial \sigma_2}{\partial x} - g\sigma_2(S_0 - S_f) + g\sigma_2 h\frac{\partial S_f}{\partial h} + \frac{\partial \mathscr{E}}{\partial h} = 0 \\
-\frac{\partial \sigma_2}{\partial t} - \frac{\partial \sigma_1}{\partial x} - 2\frac{q}{h}\frac{\partial \sigma_2}{\partial x} + \sigma_2 gh\frac{\partial S_f}{\partial q} + \frac{\partial \mathscr{E}}{\partial q} = 0
\end{gathered} \tag{9}$$

and imposing initial conditions for δh, δq and final conditions σ_1, σ_2

$$\begin{aligned} &\sigma_1(x,T)=0, \sigma_2(x,T)=0\,, \qquad 0<x<L \\ &\delta h(x,0)=0, \delta q(x,0)=0\,, \qquad 0<x<L \end{aligned} \tag{10}$$

$\delta\mathbf{J}$ can be written as

$$\delta\mathbf{J}=\int_0^T\left[\sigma_1\delta q+\sigma_2\left(\frac{2q}{h}\delta q-\left(\frac{q^2}{h^2}-gh\right)\delta h\right)\right]\Bigg|_0^L dt \tag{11}$$

which relates functional $\mathbf{J}$ with variations on the boundary condition $q(0,t)$.

Depending on the variable to be controlled, boundary conditions are required to be properly chosen. Further details about this can be found in Sect. 3.2. It is important to note that domain (0, L) must always represent the wet domain.

2.2 Numerical Method

It is possible to write (1) in non-conservative form as

$$\frac{\partial\mathbf{U}}{\partial t}+\mathbf{M}\frac{\partial\mathbf{U}}{\partial x}=\mathbf{S} \tag{12}$$

where,

$$\mathbf{U}=\begin{pmatrix} h \\ q \end{pmatrix}, \quad \mathbf{M}=\begin{pmatrix} 0 & 1 \\ c^2-u^2 & 2u \end{pmatrix}, \quad \mathbf{S}=\begin{pmatrix} 0 \\ gh\left(S_0-S_f\right) \end{pmatrix} \tag{13}$$

being $u=q/h$ the velocity and $c=\sqrt{gh}$ the wave celerity. Additionally, it is possible to write (9) similarly as

$$-\frac{\partial\boldsymbol{\Psi}}{\partial t}+\boldsymbol{\Gamma}\frac{\partial\boldsymbol{\Psi}}{\partial x}=\mathbf{R} \tag{14}$$

where

$$\boldsymbol{\Psi}=\begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}, \boldsymbol{\Gamma}=\begin{pmatrix} 0 & u^2-c^2 \\ -1 & -2u \end{pmatrix}, \mathbf{R}=\begin{pmatrix} -\dfrac{\partial\mathscr{E}}{\partial h}+g\sigma_2\left[S_0-S_f-h\dfrac{\partial S_f}{\partial h}\right] \\ -\dfrac{\partial\mathscr{E}}{\partial q}-gh\sigma_2\dfrac{\partial S_f}{\partial q} \end{pmatrix} \tag{15}$$

An explicit Euler scheme for time integration and a first order, upwind finite volume scheme for the spatial integration are applied to both systems. In the case of the physical system (1), a compact updating formula using an augmented Riemann solver is achieved [6]

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta x}\left(\sum_{m=1}^{2}(\tilde{\lambda}^{+}\gamma\tilde{\mathbf{e}})_{i-1/2}^{m} + \sum_{m=1}^{2}(\tilde{\lambda}^{-}\gamma\tilde{\mathbf{e}})_{i+1/2}^{m}\right)^{n} \tag{16}$$

where $\tilde{\lambda}$, are the linearised eigenvalues, γ are the wave strengths, $\tilde{\mathbf{e}}$ are the linearised eigenvectors and $^{\pm}$ represents the two sides of the interface on which the Riemann problem is defined. In the case of the adjoint equations and regarding backward temporal-integration [5], the backward scheme for the adjoint system follows

$$\boldsymbol{\Psi}_i^{n} = \boldsymbol{\Psi}_i^{n+1} + \frac{\Delta t}{\Delta x}\left(\sum_{m=1}^{2}(\tilde{\lambda}_{\Psi}^{+}\gamma_{\Psi}\tilde{\mathbf{e}}_{\Psi})_{i-1/2}^{m} + \sum_{m=1}^{2}(\tilde{\lambda}_{\Psi}^{-}\gamma_{\Psi}\tilde{\mathbf{e}}_{\Psi})_{i+1/2}^{m}\right)^{n+1} \tag{17}$$

Additional considerations as well as details of the numerical techniques used herein are described in [5, 6].

3 Application

The application of the adjoint technique for inverse modeling is demonstrated, analyzed and validated in this work. In particular, the main interest lies on the reconstruction of a boundary condition necessary to adjust the simulated results to a particular experiment considering that the boundary data is not available. This is of course useful in cases such as the one in this work, in which laboratory data is incomplete for the modeling application. But it is also useful in real scale, practical applications, in which it is not uncommon to have scarce or incomplete information to construct numerical boundary conditions. Sometimes, no reliable information is available at all. It is easy to imagine a problem in which, for example, buoys located at sea are used to record information which may be too close to the shore so that the collected data is not appropriate for numerical boundaries. Or perhaps a buoy is lost because of sea or weather conditions, or even technical malfunction. In such cases, in which other measurements are available, an inverse technique can be a helpful tool to complete the data required to properly pose the numerical problem.

3.1 Case Description

An experiment presented in [1, 2] is used as benchmark for the technique. The experiment was concerned with the formation of waves on a laboratory channel with a variable bathymetry and a rough surface. The flume is a a reduced model of a typical coastal region. It consists of a prismatic channel, 250 m in length, with a bed profile as shown in Fig. 1. Initially, water in the flume was motionless, with constant

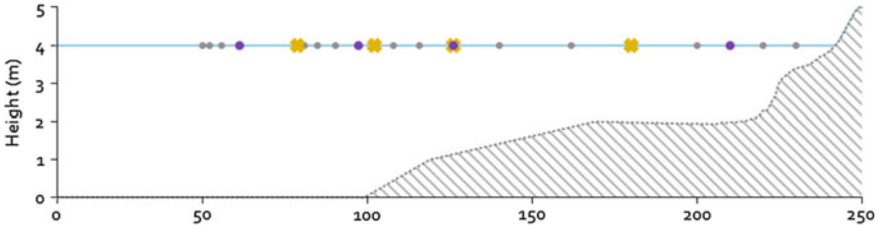


Fig. 1 Sketch of the initial configuration of the channel and probes position (•). In (●) are displayed those probes that have been included in the functional for subset *A* and (●) for the subset *B*

water surface elevation of $h + z = 4\,m$. On the seaside boundary ($x = 0$) a mechanic piston would move the water, thus generating waves.

As Fig. 1 shows, 20 probes recorded the water depth along the channel. These probes are categorized into subsets which will be discussed later. However, no information of water depth or discharge was recorded at $x = 0$. Indeed, discharge or velocity measurements are not available for any point in the channel.

As previously noted, in real applications there is often a lack of information at the boundaries of the problem. It is therefore interesting to test if an inverse technique, such as the one presented here, can reconstruct the boundary conditions so a given experimental data set can be properly reproduced by numerical simulations. Therefore, several numerical experiments were carried out to test the applicability and usefulness of the proposed methodology. In particular, the numerical experiments allow to study the following questions:

(i) Is it possible to reconstruct the boundary from depth data using the adjoint method?
(ii) How does the choice –number and location– of probes included in the functional **J** affect the quality of the reconstructed boundaries?
(iii) How well do numerical and experimental results match, at probe locations *not* included in the optimization functional?

The numerical setup consists of a mesh with 500 cells uniformly distributed ($\Delta x \approx 0.57\,\mathrm{m}$). This choice of mesh responds to the high numerical diffusivity that has been observed in this test problem for coarser meshes. The chosen mesh has been observed to perform properly, with low and acceptable numerical diffusion. Although it is likely that there is a relevant impact on control results and on the controllability of the problem, thoroghly studying such behaviour is out of the scope of this work. For friction modeling, Manning's coefficient was set to $n = 0.013$ to account for surface roughness. Additionally, $CFL = 0.9$ was set for both the SWE solver and the adjoint solver. Temporal resolution for the control (i.e., frequency at which the boundary is constructed) is $\Delta t = 0.25\,\mathrm{s}$ leading to a representation of the controlled function (the boundary condition) using 1000 points. On the other hand, an initial condition of $h + z(x) = 4.0\,\mathrm{m}$ and $q(x) = 0$ is imposed. The initial guess for the time-dependant boundary function was quiescent flow with $h_0(t) = 4\,\mathrm{m}$ for

Table 1 Subsets of probes for the numerical experiments

Set	x_1	x_2	x_3	x_4
A_1	79.05	102.09	126.22	180.00
A_2	–	–	126.22	180.00
A_3	–	–	–	180.00
B_1	61.3	97.3	126.22	210.00
B_2	–	–	126.22	210.00
B_3	–	–	–	210.00

all the numerical experiments The optimization method was applied with an initial step-size of $\varepsilon^0 = 1$.

From the numerical point of view, under subcritical flow conditions, inlet discharge or water depth are required as inlet boundary condition. Either can be reconstructed using the adjoint variables properly.

As previously described, the experimental data reports 20 probes [2]. The control method has been tested using two different subsets of these probes, summarized in Table 1. Subset $A = \{A_1, A_2, A_3\}$ which contains three combinations of probes p_5, p_{10}, p_{13} and p_{16} and subset $B = \{B_1, B_2, B_3\}$ containing p_3, p_9, p_{13} and p_{18}. Both have p_{13} in common and subset A is a tighter grouping of the probes than B.

Regarding these subsets, the six different combinations detailed in Table 1 lead to six different functionals $\mathbf{J}_{A_1}$, $\mathbf{J}_{A_2}$, $\mathbf{J}_{A_3}$, $\mathbf{J}_{B_1}$, $\mathbf{J}_{B_2}$ and $\mathbf{J}_{B_3}$. The first group for each subset contains 4 probes, the second contains two and the last contains one probe, which is the farthest probe for both subsets.

3.2 Numerical Optimization

The optimization has been defined as the minimization of the cost function (4) considering different subsets of N_p probes (from the complete set of 20 probes) in the functional $\mathbf{J}$. Each probe in the subset is equally weighted ($\omega = 1$) in the functional. Therefore Eq. (4) using (5) becomes

$$\mathbf{J} = \int_0^T \sum_{i=1}^{N_p} \frac{\omega}{N_p} (h(t) - h_{obj})_i^2 \tag{18}$$

By using the functional (18) in combination with (11), the sensitivity with respect to perturbations over $q_0(t)$ as well as over $h_0(t)$ can be evaluated. To recover h_0, the following boundary conditions are required.

Boundary conditions for $(\delta h, \delta q)$

$$\delta q(0,t) = 0, \quad \delta q(L,t) = 0, \quad \delta h(L,t) = 0\,, \quad 0 < t < T \tag{19}$$

allowing to define the sensitivity of the functional as follows

$$\delta \mathbf{J}_i = \sigma_2 \left(gh - \frac{q^2}{h^2} \right) \delta h_i \Big|_{x=0,t=t_i} \tag{20}$$

Using (20), a discrete version of the gradient is defined as the sensitivity of the functional with respect to the inlet boundary condition:

$$\nabla \mathbf{J}_i = \frac{\delta \mathbf{J}}{\delta h(0, t_i)} = \sigma_2 \left(gh - \frac{q^2}{h^2} \right) \Big|_{x=0,t=t_i} \tag{21}$$

The gradient (21) is then introduced in the gradient-descent optimizer to obtain a new set of values for the boundary condition.

$$h(0, t)_{m+1} = h(0, t)_m - \varepsilon_m \nabla \mathbf{J} \tag{22}$$

There are many gradient based methods to minimize functions. Indeed, these techniques are not only based on the first variation but also on Hessian information. In [8] a large description of them can be found. Nonetheless, the gradient-descent algorithm is used because its simplicity. In Algorithm 1 a brief description is displayed.

Algorithm 1: Gradient descent method

```
Result: A new set of values for $f$ closer to the solution that minimizes J
input  : $f^0$, $\varepsilon$, $tolX_{max}$, $tolX_{max}$, $iter_{max}$
output: $f^n$
nIter=0;
while $n_i \leq iter_{max}$ do
    $f^{n+1} \leftarrow f^n - \varepsilon^n \nabla \mathbf{J}$;
    if $||\nabla \mathbf{J}(f^{n+1})|| \leq tol_{max}$ then
        /* converged on critical point                          */
        return;
    else
        if $||f^{n+1} - f^n|| \leq tolX_{max}$ then
            /* converged on an $f$ value                         */
            return;
        else
            if $\mathbf{J}(f^n) < \mathbf{J}(f^{n+1})$ then
                /* It has diverged                              */
                $\varepsilon^{n+1} = 0.5\varepsilon^n$;
            else
                $\varepsilon^{n+1} = 1.1\varepsilon^n$;
                $n_i$++;
```

4 Results and Discussion

Optimization was performed using each of the subsets described in Table 1. The evaluation of the adjoint system results in $\sigma_1(x, t), \sigma_2(x, t)$ as displayed in Fig. 2. Figure 2 shows both σ_1, σ_2 in the time-space (x, t) domain for the first iteration of the gradient method using subsets A_1 and A_3. It is very interesting to observe how $\partial_h \mathscr{E}$ is included as a source term for σ_1 and the effect that it produces on σ_2 where discrepancies are presented. In particular, the injection at each of the selected probes in each subset can be observed as a modification of adjoint variables in the (x, t) plane. This illustrates that, in fact, the adjoint equations model a transport phenomenon for the sensitivity. In Fig. 2 it is possible to observe how the adjoint variables act as a scalar tracer traveling through physical space similiar to how water waves do.

Figure 3 displays (σ_1, σ_2) for subset B_2. When comparing Fig. 3 to Fig. 2, differences can be observed between cases B_2 and A_1, for example. These differences are more significant on the landward end of the domain than at the seaward end. Comparing $\sigma_1(0, t)$ for both A_1 and B_1 subsets, differences are increasingly important as time 'advances' backwards. In particular, sensitivity for earlier times comes from the combination of the waves coming from the farthest probe and the interaction of the adjoint waves with the wet/dry front.

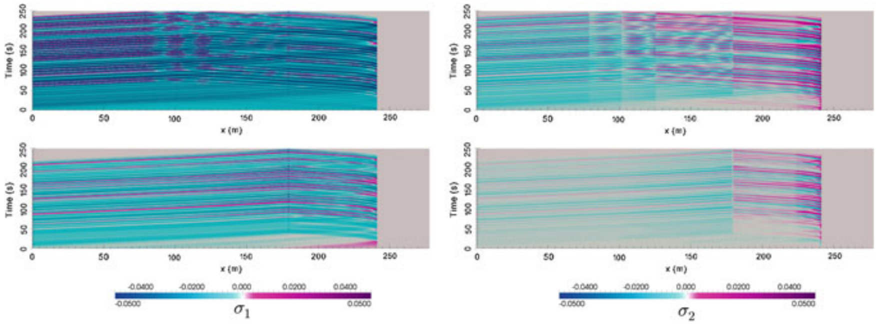


Fig. 2 Plot of $\sigma_1(x, t)$ (left) and $\sigma_2(x, t)$ (right) for the first iteration of the optimizer using the set A_1 (top) A_2 (middle) and A_3 (bottom)

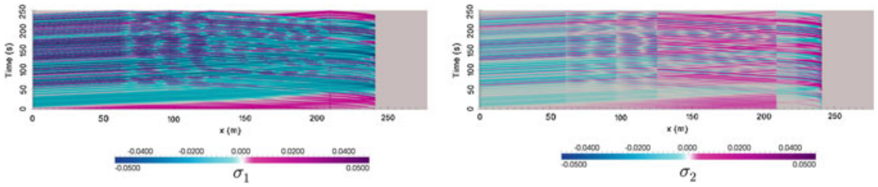


Fig. 3 Plot of $\sigma_1(x, t)$ (left) and $\sigma_2(x, t)$ (right) for the first iteration of the optimizer using the set A_1 (top) A_2 (middle) and A_3 (bottom)

In addition, both Figs. 2 and 3 show the wet/dry boundary approximately at $x = 240$ m. The wet/dry boundary works as an internal boundary condition that reflects the adjoint variables, as happens also to the physical waves. This feature is tremendously important because it shows the interaction between that front and the inlet boundary condition which is sought.

In general, it was possible to reconstruct the boundary using different sets of probes. Nevertheless, the results differ both in terms of convergence and quality. Table 2 summarizes convergence results for the six data sets analyzed, including the criteria on which convergence was achieved. In addition, Fig. 4 show the convergence curves obtained using subsets A and B. Note that the whole metric (the complete functional) achieves lower values as less probes are included. This is not due to the overall error being reduced, but because less contributions are considered (probes in the functional) are being considered. From Table 2 and Fig. 4 it can be observed that, in both subsets, as less probes are included in the functional, the algorithm requires more iterations to achieve convergence. Additionally, subset A requires less iterations than subset B, except in the third case, in which notably, the algorithm was stopped by the maximum number of iterations criterion (tolerance as defined as $tol = 10^{-5}$). Otherwise, for all other cases the algorithm converged using the same criterion. Interestingly, as further results will show, case A_3 still results in a reasonable solution although it exceeded the maximum number of allowed iterations. Furthermore, subset A results in converged values for the functional lower than the

Table 2 Convergence results for the different numerical experiments

Set	Set A			Set B		
	A_1	A_2	A_3	B_1	B_2	B_3
Iterations	38	54	99	40	79	92
J	0.311216	0.222080	0.154499	0.474967	0.211037	0.1818
Stop criterion	$\|\|\delta h_0\|\| \leq tol$	$\|\|\delta h_0\|\| \leq tol$	$iters \geq iter_{max}$	$\|\|\delta h_0\|\| \leq tol$	$\|\|\delta h_0\|\| \leq tol$	$\|\|\delta h_0\|\| \leq tol$

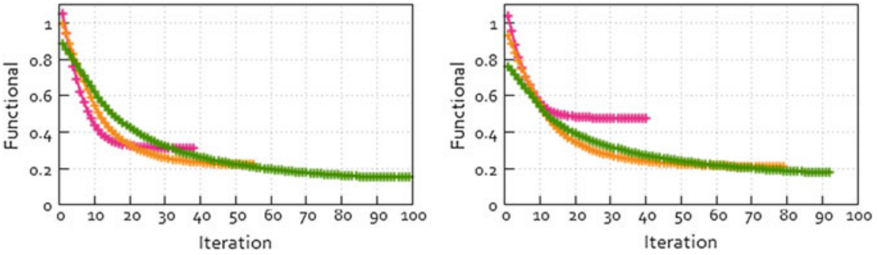


Fig. 4 Convergence of the gradient optimizer for the reconstruction of the inlet BC using subset A (left) and for the subset B (right). Groups 1 (A_1, B_1) (-x-), 2 (A_2, B_2) (-x-) and three 3 (A_3, B_3)(-x-) are displayed

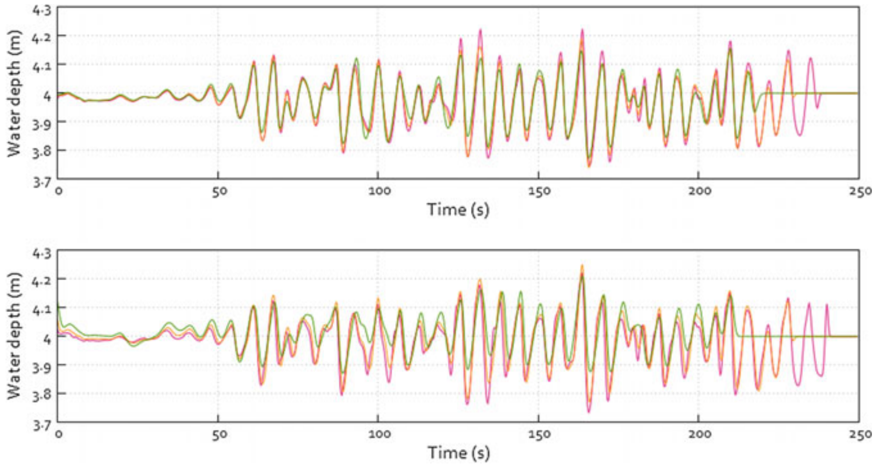


Fig. 5 Reconstructed time-dependent boundary condition using subset A (upper) and subset B (lower). Boundary water depth for A_1 and B_1 (—), sets A_2 and B_2 (—) and sets A_3 and B_3 (—) are shown.

one obtained with subset B, for the cases with 4 probes and a single probe. Case A_2 shows a larger functional value than B_2.

Fig. 5 shows the reconstructed boundary condition using each subset. When using subset A, little differences are observed when the number of probes is varied, for most of the simulation. On the other hand, when varying the number of probes using subset B, specially when selecting B_3, much larger differences appear in the reconstructed boundary. For both cases, note that when decreasing the number of probes, the boundary time evolution fails to capture phenomena in the latest parts of the experiment. Very clearly, the green line goes flat before the orange, and this one before the pink. This is reasonable and is caused by the fact that, for very late times, adjoint information cannot travel all the way across the domain, and therefore, no information is available at such late times to reconstruct the boundary. This is clearly a limitation for this strategy, which may be effectively solved by taking a time window larger for the optimization than the one intended for simulations.

From Fig. 4, it might be interpreted that, following the values of the functional, cases with less probes achieve better results than cases with more probes. This is not the case. Actually, as more probes are considered, the simulated results better match experimental results. This is very clear in Table 3 and in Fig. 7, where mean square error (MSE) for all of the probes (not only those in the functional), obtained with the optimal boundary condition for each of the subsets are shown. Overall, A outperforms B, except for case 2. Moreover, case 1 outperforms case 2 which outperforms case 3, for both subsets A and B, which is expected from the fact that more information is being used to obtain a better boundary condition.

The influence of choosing the probe locations can be extracted from Table 3. Subset A_2 reduces the mean global error using 2 probes as much as B_1 using 4.

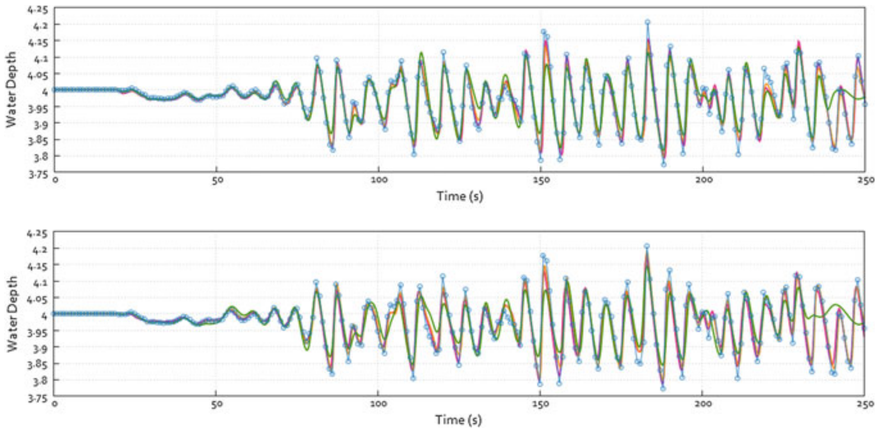


Fig. 6 Comparison of experimental water depth (-○-) for probe 13, with numerical results for subset A (upper) and B (lower). Water depth using subsets A_1 and B_1 (—), subsets A_2 and B_2 (—) and subsets A_3 and B_3 (—) are shown

Additionally using only one probe it is possible to fit the error for that position better than when using more, but in general it is not enough for the rest of the positions (the mean and also individual MSE is higher). This can be observed comparing MSE at probe 18 with B_3 and B_1 where B_3 reduces the error, but only for that position. Results suggest that using a large number of points to perform this control is less significative than choosing their position properly.

It was previously shown in Fig. 5 and discussed that by the end of the simulation ($t \geq 200\, s$) the reconstructed boundaries are incorrect, because information does not arrive to control them. This also implies that the error in all probes, towards the end of the simulation should also increase, and specially in those probes close to the boundary. Such probes will receive incorrect boundary information at later times, but probes farther away, may not receive the incorrect information before the simulation ends. This also shows the *transport* nature of the adjoint system, as well of course as of the shallow water system. To illustrate this, consider the errors computed only until $t = 200$, but with a control that goes on until $t = 250$, as shown in Fig. 8. It is clear there that is a larger reduction of error in the seaward probes than in the landward probes.

To further explore the effect of the different subsets, consider Fig. 6, which shows the results for probe p_{13} which is included within both subset A and B. The results obtained with the three subsets A show less differences than those obtained with B. In the figure it is clear that simulated and experimental results match better when more probes are used in the functional, for both A and B subsets, consistently with what is observed in Fig. 7. Furthermore, in the case of subset A the energy (amplitude) of the waves seems to be more affected than the shape of the wave structures. In contrast, with subset B, the simulated results are somewhat distorted and shifted in

Table 3 Mean Square Error for all probes and numerical experiments with the reconstruced boundaries (in bold, values which intervene in the functional in each case)

	Set A			Set B		
Set	A_1	A_2	A_3	B_1	B_2	B_3
p_1	0.4416	0.5652	0.7306	0.5780	0.5525	0.8565
p_2	0.4490	0.5421	0.7065	0.4981	0.5305	0.8321
p_3	0.4142	0.5454	0.7113	**0.5268**	0.5318	0.8364
p_4	0.3583	0.5049	0.6653	0.5213	0.5038	0.8128
p_5	**0.2972**	0.4045	0.6028	0.4111	0.4108	0.7426
p_6	0.2897	0.4431	0.6189	0.4725	0.4540	0.7602
p_7	0.3769	0.4123	0.5976	0.3571	0.4155	0.7306
p_8	0.3475	0.4249	0.5916	0.4066	0.4344	0.7379
p_9	0.3375	0.4091	0.5851	**0.3492**	0.4154	0.7159
p_{10}	**0.3590**	0.4269	0.5956	0.3507	0.4294	0.7160
p_{11}	0.3301	0.4150	0.5808	0.3883	0.4075	0.6881
p_{12}	0.3012	0.3223	0.5191	0.3858	0.3107	0.6429
p_{13}	**0.3560**	**0.2599**	0.5454	**0.2933**	**0.2419**	0.6608
p_{14}	0.3947	0.3673	0.4898	0.3609	0.3847	0.6186
p_{15}	0.3295	0.3575	0.4266	0.3498	0.3672	0.5186
p_{16}	**0.2370**	**0.2218**	**0.1550**	0.3880	0.3598	0.5022
p_{17}	0.3370	0.3727	0.4120	0.3495	0.3304	0.4332
p_{18}	0.3399	0.3756	0.3853	**0.2994**	**0.2181**	**0.1818**
p_{19}	0.3000	0.3257	0.3286	0.3309	0.3237	0.4136
p_{20}	0.4388	0.4289	0.4620	0.3912	0.4618	0.5748
Mean	0.3518	0.4063	0.5355	0.4004	0.4042	0.6488
Mean (200 s)	0.2851	0.2980	0.3766	0.3302	0.2910	0.4658

comparison to the experimental data. Importantly, this effect increases as the number of probes reduces (Fig. 8).

This behavior is highly visible for farther probes. Figure 9 shows the results for probe p_{19}, and plainly speaks that maximum amplitude peaks cannot be reproduced by any of the combinations. It is interesting to note that numerical results are bounded within the experimental measurements, for most of the time.

Notably, the lower quality results using subset B are likely caused by probe p_{19}. This is due to the fact that, as selected probes are located farther from the boundary, the boundary optimization becomes more complex. In addition, phenomena near the shore, where surf is produced, are strongly governed by vertical accelerations not considered in the shallow water approach. Therefore, an uncomfortable contradiction may be the cause of the problems: choosing probes for the functional where shallow flow is observed, may result in inclusion of information related to *incorrect* modellization, and may provide an incorrect solution. In turn, this means that the

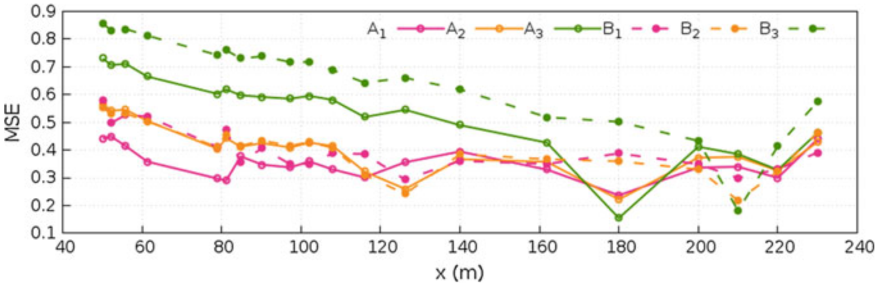


Fig. 7 Mean Square Error obtained at each probe location for all numerical experiments

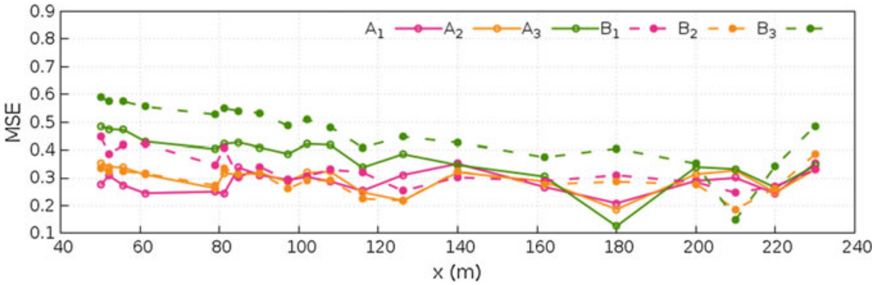


Fig. 8 Mean Square Error obtained at each probe location for all numerical experiments, using only results for $t \leq 200\,\text{s}$

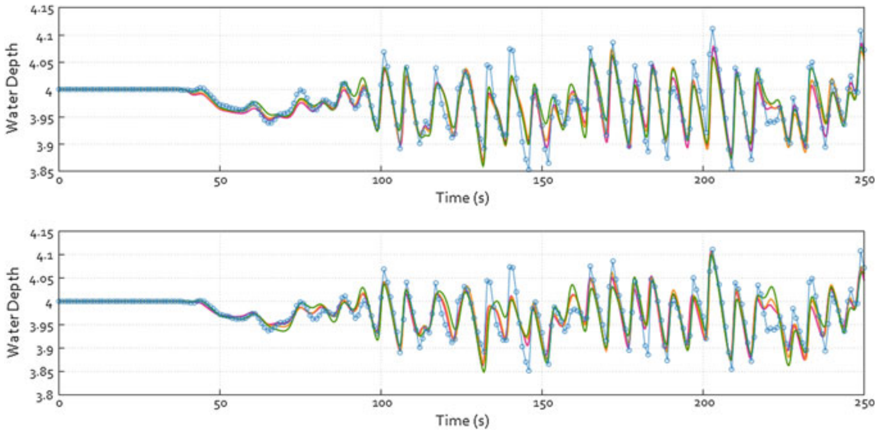


Fig. 9 Comparison of experimental water depth (-○-) for probe 19, with numerical results for subset A (upper) and B (lower). Water depth using subsets A_1 and B_1 (—), subsets A_2 and B_2 (—) and subsets A_3 and B_3 (—) are shown

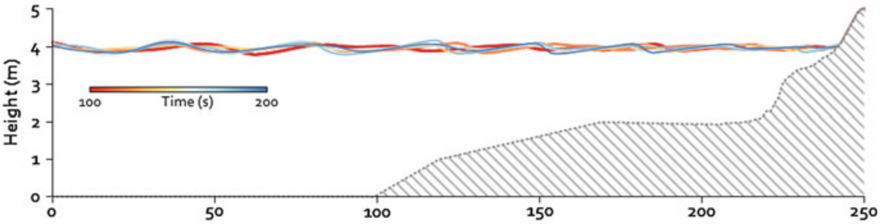


Fig. 10 Evolution of the problem at different times from $t = 100$ (red) to $t = 200$ (blue). Results using controlled using BC A_1

rather low quality of results obtained with B_3 are not caused from the fact that a single probe is used, but because this single probe is inappropriate for inverse modelling.

Figure 10 shows the flume state at different times with the controlled BC using A_1. It is possible to observe how, when reaching the shore, wave amplitude is reduced, while simultaneously a shock is formed. It is important to note that this non-linear effect represents an important challenge for the optimization method.

5 Conclusions

The adjoint method has been applied on a hyperbolic system of PDEs with source terms. In particular, the adjoint system of equations of the shallow water equations has been obtained. It has been shown to be a useful technique to solve an inverse problem to reconstruct the boundary condition based on data from several experimental measurement points, in order to fit simulated results to such experimental measurements. Furthermore, several combinations of experimental points were selected to test the sensitivity of the method to this choice. Depending on this combination, control achievements are different, both in terms of convergence and quality. The tests show that, by increasing the number of measurements and constraints included in the functional, the reconstructed boundaries produce better numerical results. However, in light of the results, it is still possible to reconstruct such boundary using even a single point, which makes this technique interesting and useful in real applications with scarce field or experimental data. Nevertheless, care must be taken to feed the optimization process with appropriate data, that is, close enough to the boundary so that information can travel across the domain to reach the boundary during the time window of the simulation, and that it falls not in a place where model assumptions might break, such as wave breaking, surf regions or the dry region further landwards.

From the point of view of interpreting the results of the discrete adjoint problem, it is interesting to note that, in the case of transient phenomena, the adjoint system acts as a transport law for the sensitivity along the characteristic space. Because of that, this technique is a very fast solution that does not require to cover all the solution space, but only those regions in which the adjoint information goes in the optimal direction to find the correct solution. This advantage also implies a drawback, and it is that, depending on the time-window selected for controlling, information may not have enough time to travel across the domain and reach the controlled boundary, thus failing to reproduce late times of the simulation.

Acknowledgements Authors would like to thank Dr. Alessandra Saponieri for the access to the experimental background and data used in this work. This worked has been partially funded by the Spanish Ministry of Innovation and Science trough research project BIA2011-30192-C02-01.

References

1. Damiani, L., Aristodemo, F., Saponieri, A., Verbeni, B., Veltri, P., Vicinanza, D.: Full-scale experiments on a beach drainage system: hydrodynamic effects inside beach. J. Hydraul. Res. **49**(sup1), 44–54 (2011)
2. Damiani, L., Vicinanza, D., Aristodemo, F., Saponieri, A., Corvaro, S.: Experimental investigation on wave set up and nearshore velocity field in presence of a bds. J. Coast. Res. 55–59 (2011)
3. Ding, Y., Wang, S.S.: Optimal control of open-channel flow using adjoint sensitivity analysis. J. Hydraul. Eng. **132**(11), 1215–1228 (2006)
4. Giles, M.B., Pierce, N.A.: An introduction to the adjoint approach to design. Flow Turbul. Combust. **65**(3–4), 393–415 (2000)
5. Lacasta, A., Morales-Hernández, M., Brufau, P., Gacía-Navarro, P.: Calibration of the 1d shallow water equations using adjoint variables. In: Advances in Numerical Modelling of Hydrodynamics Workshop, Sheffield, UK (2015)
6. Murillo, J., García-Navarro, P.: Weak solutions for partial differential equations with source terms: application to the shallow water equations. J. Comput. Phys. **229**(11), 4327–4368 (2010)
7. Sanders, B.F., Katopodes, N.D.: Adjoint sensitivity analysis for shallow-water wave control. J. Eng. Mech. **126**(9), 909–919 (2000)
8. Wright, S.J., Nocedal, J.: Numerical optimization, vol. 2. Springer, New York (1999)

Aerodynamic Optimization of Car Shapes Using the Continuous Adjoint Method and an RBF Morpher

E. M. Papoutsis-Kiachagias, S. Porziani, C. Groth, M. E. Biancolini, E. Costa and K. C. Giannakoglou

Abstract This paper presents the application of the continuous adjoint method, programmed in OpenFOAM$^{©}$, combined with an RBF-based morpher to the aerodynamic optimization of a generic car model. The continuous adjoint method produces accurate sensitivities by utilizing the full differentiation of the Spalart–Allmaras turbulence model, based on wall functions, while the RBF-based morpher provides a fast and versatile way to deform both the surface of the car and the interior mesh nodes. The integrated software is used to minimize the drag force exerted on the surface of the DrivAer car model.

1 Introduction

During the last years, CFD-based aerodynamic shape optimization has been attracting the interest of both academia and industry. The constituents needed for executing an automated shape optimization loop include the flow solver, the

E. M. Papoutsis-Kiachagias (✉) · K. C. Giannakoglou (✉)
National Technical University of Athens, Parallel CFD & Optimization Unit,
Athens, Greece
e-mail: vaggelisp@gmail.com

K. C. Giannakoglou
e-mail: kgianna@central.ntua.gr

E. Costa (✉)
RINA Consulting S.p.A., Viale Cesare Pavese, 305 - 00144 Rome, Italy
e-mail: emiliano.costa@rina.org

S. Porziani · C. Groth · M. E. Biancolini (✉)
University of Rome Tor Vergata (UTV), Rome, Italy
e-mail: porziani@ing.uniroma2.it

C. Groth
e-mail: corrado.groth@uniroma2.it

M. E. Biancolini
e-mail: biancolini@ing.uniroma2.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_11

geometry parameterization (the parameters of which act as the design variables), an optimization method capable of computing the optimal values of the design variables and a way to adapt (or regenerate) the computational mesh to each candidate solution.

Nowadays, a great variety of in-house and commercial flow solvers exist and are in widespread use. In the study presented in this paper, the steady-state flow solver of the open-source CFD toolbox, OpenFOAM©, is used to numerically solve the Navier-Stokes equations for incompressible, turbulent flows.

Shape parameterization techniques can be divided into two categories, i.e. those parameterizing only the surface to be optimized and those which also deform the surrounding nodes of the interior mesh. The former include the normal displacement of surface wall nodes [15], the control points of Bézier–Bernsteinor NURBS curves or surfaces and CAD parameters [16, 18]. The latter include volumetric B-splines or NURBS [11], Radial Basis Functions (RBFs) [2, 4, 7], the harmonic coordinates method [8], etc. The great advantage of the this category is that the interior of the computational mesh is also deformed, avoiding, thus, costly re-meshing and allowing the initialization of the flow field from the solution obtained in the previous optimization cycle, since the mesh topology is preserved. In this paper, a number of parameters controlling the positions of groups of RBF control points are used as the design variables, using technology and methods developed in the context of the RBF Morph software [3].

Optimization methods can be separated into two main categories, i.e. stochastic and gradient-based ones. Stochastic optimization methods, with Evolutionary Algorithms (EAs) as their main representative, are extremely versatile and have the ability to compute global optima but suffer from a computational cost that scales with the number of design variables, making their use impractical for large scale optimization problems. On the other hand, gradient-based optimization methods require a higher effort to develop and maintain but can have a cost per optimization cycle that does not scale with the number of design variables, when the adjoint method is used to compute the gradients of the objective function. Both discrete and continuous adjoint methods [5, 13], have been developed. In this work, a continuous adjoint method that takes into consideration the differentiation of the turbulence model PDE is used to increase the accuracy of the computed sensitivities of the drag force objective function w.r.t. the shape modification parameters [14]. The continuous adjoint solver has been implemented on an in-house version of the OpenFOAM© software.

The above-mentioned tools are combined in order to form an automated optimization loop, targeting the minimization of the drag force exerted on the surface of a generic car model. In specific, a configuration of the DrivAer car model [6], developed by the Institute of Aerodynamics and Fluid Mechanics of TU Munich, is studied. The constituents of the optimization loop were combined under the RBF4AERO project. Funded in the Aeronautics and Air Transport (AAT) research thematic area of the EU Seventh Framework Programme, the RBF4AERO Project aims at developing the RBF4AERO Benchmark Technology, namely a numerical platform conceived to face the requirements of top-level aeronautical design studies such as multi-physics and multi-objective optimization, fluid-structure interaction (FSI), adjoint-driven optimization and ice accretion simulation. Based on the RBF mesh morphing technique, such a numerical platform allows to significantly boost the aerodynamic design process and a relevant impact is then expected in the ever-growing technological demand posed by aeronautical manufacturers in relation to the performance and reliability of aircrafts constituting components. To demonstrate the general validity and the effective usage of the RBF4AERO platform in the industrial field, one of its capabilities envisaging the adjoint-morphing coupling is, herein, described for a car aerodynamics optimization problem.

2 The Continuous Adjoint Method

In this section, the formulation of the continuous adjoint PDEs, their boundary conditions and the sensitivity derivatives (gradient) expression are presented in brief. The development is based on the incompressible Navier-Stokes equations.

2.1 Flow Equations

The mean flow equations together with the Spalart–Allmaras turbulence model PDE [17], comprise the flow or primal system of equations that reads

$$R^p = -\frac{\partial v_i}{\partial x_i} = 0 \tag{1a}$$

$$R_i^w = v_j \frac{\partial v_i}{\partial x_j} + \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} = 0 \tag{1b}$$

$$R^{\widetilde{\nu}} = \frac{\partial (v_j \widetilde{\nu})}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu}}{\partial x_j}\right] - \frac{c_{b2}}{\sigma}\left(\frac{\partial \widetilde{\nu}}{\partial x_j}\right)^2 - \widetilde{\nu}\, P\left(\widetilde{\nu}, \varDelta\right) + \widetilde{\nu}\, D\left(\widetilde{\nu}, \varDelta\right) = 0 \tag{1c}$$

where v_i are the components of the velocity vector, p is the static pressure divided by the constant density, $\tau_{ij} = (\nu + \nu_t)\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)$ are the components of the stress tensor, ν and ν_t the kinematic and turbulent viscosity, respectively, $\widetilde{\nu}$ the Spalart–Allmaras model variable and $\varDelta$ the distance from the wall boundaries. Details about the turbulence model constants, source terms and boundary conditions can be found in [17].

2.2 General Objective Function

Let F be the objective function to be minimized by computing the optimal values of the design variables $b_n, n \in [1, N]$. A general expression for an objective function defined on (parts of) the boundary S of the computational domain $\varOmega$ is given by

$$F = \int_S F_{S_i} n_i dS \tag{2}$$

where **n** is the outward facing normal unit vector.

Differentiating Eq. 2 w.r.t. to b_n and applying the chain rule yields

$$\frac{\delta F}{\delta b_n} = \int_S \frac{\partial F_{S_i}}{\partial v_k} n_i \frac{\partial v_k}{\partial b_n} dS + \int_S \frac{\partial F_{S_i}}{\partial p} n_i \frac{\partial p}{\partial b_n} dS + \int_S \frac{\partial F_{S_i}}{\partial \tau_{kj}} n_i \frac{\partial \tau_{kj}}{\partial b_n} dS + \int_S \frac{\partial F_{S_i}}{\partial \widetilde{\nu}} n_i \frac{\partial \widetilde{\nu}}{\partial b_n} dS$$
$$+ \int_S n_i \frac{\partial F_{S_i}}{\partial x_k} \frac{\delta x_k}{\delta b_n} n_k dS + \int_S F_{S_i} \frac{\delta (n_i dS)}{\delta b_n} \tag{3}$$

where $\delta \Phi / \delta b_n$ is the total derivative of any quantity Φ while $\partial \Phi / \partial b_n$ is its partial derivative. These are related by

$$\frac{\delta \Phi}{\delta b_n} = \frac{\partial \Phi}{\partial b_n} + \frac{\partial \Phi}{\partial x_k} \frac{\delta x_k}{\delta b_n} \tag{4}$$

Computing the variation of the flow variables on the r.h.s. of Eq. 3, either through Direct Differentiation (DD) or Finite Differences (FD) would require at least N equivalent flow solutions. To avoid this computational cost that scales with N, the adjoint method is used, as presented in the next subsection.

2.3 Continuous Adjoint Formulation

Starting point of the continuous adjoint formulation is the introduction of the augmented objective function

$$F_{aug} = F + \int_\Omega u_i R_i^v d\Omega + \int_\Omega q R^p d\Omega + \int_\Omega \widetilde{\nu_a} R^{\widetilde{\nu}} d\Omega \tag{5}$$

where u_i are the components of the adjoint velocity vector, q is the adjoint pressure and $\widetilde{\nu_a}$ is the adjoint turbulence model variable, respectively. Dropping the last integral on the r.h.s. of Eq. 5 would result to the so-called "frozen turbulence" assumption which neglects the differentiation of the turbulence model PDE. This assumption leads to reduced gradient accuracy, possibly even to wrong sensitivity signs [20]. To avoid making the "frozen turbulence" assumption, the Spalart–Allmaras model PDE has been differentiated, see [20]. A review on continuous adjoint methods for turbulent flows can be found in [14].

The differentiation of Eq. 5, based on the Leibniz theorem, yields

$$\frac{\delta F_{aug}}{\delta b_n} = \frac{\delta F}{\delta b_n} + \int\limits_{\Omega} u_i \frac{\partial R_i^v}{\partial b_n} d\Omega + \int\limits_{\Omega} q \frac{\partial R^p}{\partial b_n} d\Omega + \int\limits_{\Omega} R^{\widetilde{\nu}} \frac{\partial R^{\widetilde{\nu_a}}}{\partial b_n} d\Omega \\ + \int\limits_{S_W} (u_i R_i^v + q R^p + \widetilde{\nu_a} R^{\widetilde{\nu}}) n_k \frac{\delta x_k}{\delta b_n} dS \tag{6}$$

Then, the derivatives of the flow residuals in the volume integrals on the r.h.s. of Eq. 6 are developed by differentiating Eq. 1 and applying the Green-Gauss theorem, where necessary. This development can be found in [14, 20].

In order to obtain a gradient expression which does not depend on the partial derivatives of the flow variables w.r.t. b_n, their multipliers in (the developed form of) Eq. 6 are set to zero, giving rise to the field adjoint equations

$$R^q = -\frac{\partial u_j}{\partial x_j} = 0 \tag{7a}$$

$$R_i^v = u_j \frac{\partial v_j}{\partial x_i} - \frac{\partial (v_j u_i)}{\partial x_j} - \frac{\partial \tau_{ij}^a}{\partial x_j} + \frac{\partial q}{\partial x_i} + \widetilde{\nu_a} \frac{\partial \widetilde{\nu}}{\partial x_i} - \frac{\partial}{\partial x_l}\left(\widetilde{\nu_a}\widetilde{\nu}\frac{\mathscr{C}_Y}{Y} e_{mjk} \frac{\partial v_k}{\partial x_j} e_{mli}\right) = 0 \tag{7b}$$

$$R^{\widetilde{\nu_a}} = -\frac{\partial (v_j \widetilde{\nu_a})}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\widetilde{\nu}}{\sigma}\right)\frac{\partial \widetilde{\nu_a}}{\partial x_j}\right] + \frac{1}{\sigma}\frac{\partial \widetilde{\nu_a}}{\partial x_j}\frac{\partial \widetilde{\nu}}{\partial x_j} + 2\frac{c_{b2}}{\sigma}\frac{\partial}{\partial x_j}\left(\widetilde{\nu_a}\frac{\partial \widetilde{\nu}}{\partial x_j}\right) + \widetilde{\nu_a}\widetilde{\nu}\mathscr{C}_{\widetilde{\nu}} \\ + \frac{\partial \nu_t}{\partial \widetilde{\nu}}\frac{\partial u_i}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) + (-P + D)\,\widetilde{\nu_a} = 0 \tag{7c}$$

where $\tau_{ij}^a = (\nu + \nu_t)\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$ are the components of the adjoint stress tensor. Equation 7c is the adjoint turbulence model equation, from which the adjoint turbulence model variable $\widetilde{\nu_a}$ is computed.

The adjoint boundary conditions are derived by treating the flow variations in the boundary integrals (of the developed form of) Eq. 6. This development is presented in detail in [14, 20].

In industrial applications, the wall function technique is used routinely in analysis and design. When the design is based on the adjoint method, considering the adjoint to the wall function model becomes necessary. The continuous adjoint method in optimization problems, governed by the RANS turbulence models with wall functions, was initially presented in [21], where the adjoint wall function technique was introduced for the $k - \varepsilon$ model and a vertex–centered finite volume method with slip velocity at the wall. The proposed formulation led to a new concept: the "adjoint law of the wall". This bridges the gap between the solid wall and the first node off the wall during the solution of the adjoint equations. The adjoint wall function technique has also been implemented for flow solvers based on cell-centered finite-volume schemes, for the Spalart–Allmaras [14], and $k - \omega\ SST$ [9], turbulence models.

After satisfying the adjoint PDEs and their boundary conditions, the remaining terms in Eq. 6 yield the sensitivity derivatives

$$
\begin{aligned}
\frac{\delta F_{aug}}{\delta b_n} = T_{SD}^{WF} &- \int\limits_{S_{W_p}} SD_1 \frac{\partial \tau_{ij}}{\partial x_m} n_j t_i^I n_m n_k \frac{\delta x_k}{\delta b_n} dS - \int\limits_{S_{W_p}} SD_1 \tau_{ij} \frac{\delta (n_j t_i^I)}{\delta b_n} dS \\
&+ \int\limits_{S_{W_p}} SD_{2,i} v_{\langle t \rangle}^I \frac{\delta t_i^I}{\delta b_n} dS - \int\limits_{S_{W_p}} SD_{2,i} \frac{\partial v_i}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \\
&- \int\limits_{S_{W_p}} \left[\left(\nu + \frac{\widetilde{\nu}}{\sigma} \right) \frac{\partial \widetilde{\nu_a}}{\partial x_j} n_j + \frac{\partial F_{S_z}}{\partial \widetilde{\nu}} n_z \right] \frac{\partial \widetilde{\nu}}{\partial x_m} n_m n_k \frac{\delta x_k}{\delta b_n} dS \\
&- \int\limits_{S_{W_p}} \left(-u_{\langle n \rangle} + \frac{\partial F_{S_{W_p,k}}}{\partial \tau_{lm}} n_k n_l n_m \right) TS_1 dS - \int\limits_{S_{W_p}} \frac{\partial F_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^I t_m^I TS_2 dS \\
&- \int\limits_{S_{W_p}} \left(\frac{\partial F_{S_{W_p,k}}}{\partial \tau_{lm}} n_k (t_l^{II} t_m^I + t_l^I t_m^{II}) \right) TS_3 dS - \int\limits_{S_{W_p}} \frac{\partial F_{S_{W_p,k}}}{\partial \tau_{lm}} n_k t_l^{II} t_m^{II} TS_4 dS \\
&+ \int\limits_{S_{W_p}} (u_i R_i^v + q R^p + \widetilde{\nu_a} R^{\widetilde{\nu}}) \frac{\delta x_k}{\delta b_n} n_k dS
\end{aligned}
\tag{8}
$$

where

$$
SD_1 = -u_{\langle t \rangle}^I + \phi_{ij} t_i^I n_j + \phi_{ij} n_i t_j^I, \ SD_{2,i} = \tau_{a.ij} n_j - q n_i + \frac{\partial F_{S_{W_p,k}}}{\partial v_i} n_k, \ \phi_{ij} \quad = \frac{\partial F_{S_{W_p,k}}}{\partial \tau_{ij}} n_k
$$

Functions TS_1 to TS_4 can be found in [14] while term T_{SD}^{WF} results from the differentiation of the law of the wall.

The deformation velocities, $\delta x_k / \delta b_n$, included in Eq. 8 express the dependency of the boundary wall nodes on the shape modification parameters. This can be computed by differentiating the surface parameterization scheme presented in the next section.

3 RBF-based Morphing

In this section the mesh morphing algorithm based on RBFs is described. The background theory of RBFs is first introduced providing details of its application in mesh morphing field; the industrial implementation of the method as provided by the stand alone version of the software RBF Morph is then described; finally the coupling of the mesh morphing tool and the adjoint sensitivity is explained.

Table 1 Typical RBF functions

Radial Basis Functions (RBF) with global support	$\varphi(r), r = \|r\|$
Spline type (R_n)	$r^n, nodd$
Radial Basis Functions (RBF) with compact support	$\varphi(r) = f(\xi), \xi \leq 1, \xi = \frac{r}{R_{sup}}$
Wendland C^0 (C0)	$(1-\xi)^2$
Wendland C^2 (C2)	$(1-\xi)^4(4\xi+1)$
Wendland C^4 (C4)	$(1-\xi)^6\left(\frac{35}{3}\cdot\xi^2+6\xi+1\right)$

3.1 RBFs Background

RBFs are powerful mathematical functions able to interpolate data defined at discrete points only (source points) in a n-dimensional environment. The interpolation quality and its behavior depends on the chosen radial basis function.

In general, the solution of the RBF mathematical problem consists on the calculation of the scalar parameters (sought coefficients) of a linear system of order equal to the number of considered source points. The RBF system solution, determined after defining a set of source points with their displacement, is employed to operate mesh morphing to the discretized domain of the computational model. Operatively, once the RBF system coefficients have been calculated, the displacement of an arbitrary node of the mesh, either inside (interpolation) or outside (extrapolation) the domain, can be expressed as the sum of the radial contribution of each source point (if the point falls inside the influence domain). In such a way, a desired modification of the mesh nodes position (smoothing) can be rapidly applied preserving mesh topology.

RBFs can be classified on the basis of the type of support (global or compact) they have, meaning the domain where the chosen RBF is non zero-valued.

Typical RBFs with global and compact support are shown in Table 1. RBFs are scalar functions with the scalar variable r, which is the Euclidean norm of the distance between two points defined in a generic n-dimensional space.

An interpolation function composed of a radial basis φ and a polynomial h of order $m-1$, where m is said to be the order of φ, introduced with the aim to guarantee the compatibility for rigid motions, is defined as follows if N is the total number of contributing source points

$$s(x) = \sum_{i=1}^{N} \gamma_i \varphi\left(\left\|x - x_{k_i}\right\|\right) + h(x) \tag{9}$$

The degree of the polynomial has to be chosen depending on the kind of RBF adopted. A radial basis fit exists if the coefficients γ_i and the weights of the polynomial can be found such that the desired function values are obtained at source points and the polynomial terms gives no contributions at source points, that is

$$s(x_{k_i}) = g_i, 1 \leq i \leq N \tag{10}$$

$$\sum_{i=1}^{N} \gamma_i q(x_{k_i}) = 0 \tag{11}$$

for all polynomials q with a degree less or equal to that of polynomial h. The minimal degree of polynomial h depends on the choice of the RBF. A unique interpolant exists if the basis function is a conditionally positive definite function [12]. If the RBFs are conditionally positive definite of order $m \leq 2$ [1], a linear polynomial can be used

$$h(x) = \beta_1 + \beta_2 x + \beta_3 y + \beta_4 z \tag{12}$$

The subsequent exposition will assume that the aforementioned hypothesis is valid. The values for the coefficients γ_i of RBF and the coefficients β of the linear polynomial can be obtained by solving the system

$$\begin{pmatrix} \mathbf{M} & \mathbf{P} \\ \mathbf{P^T} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \gamma \\ \beta \end{pmatrix} = \begin{pmatrix} \mathbf{g} \\ \mathbf{0} \end{pmatrix} \tag{13}$$

where g are the known values at the source points and $\mathbf{M}$ is the interpolation matrix defined calculating all the radial interactions between source points

$$M_{ij} = \varphi\left(\left\|x_{k_i} - x_{k_j}\right\|\right), 1 \leq i \leq N, 1 \leq j \leq N \tag{14}$$

P is a constraint matrix that arises to balance the polynomial contribution and contains a column of"1" and the x y z positions of source points in the other three columns

$$\mathbf{P} = \begin{pmatrix} 1 & x_{k_1} & y_{k_1} & z_{k_1} \\ 1 & x_{k_2} & y_{k_2} & z_{k_2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{k_N} & y_{k_N} & z_{k_N} \end{pmatrix} \tag{15}$$

RBF interpolation works for scalar fields. For the smoothing problem, each component of the displacement field prescribed at the source points is interpolated as follows

$$\left\{ \begin{array}{c} s_x(x) = \sum\limits_{i=1}^{N} \gamma_i^x \varphi\left(\left\|x - x_{k_i}\right\|\right) + \beta_1^x + \beta_2^x x + \beta_3^x y + \beta_4^x z \\ s_y(x) = \sum\limits_{i=1}^{N} \gamma_i^y \varphi\left(\left\|x - x_{k_i}\right\|\right) + \beta_1^y + \beta_2^y x + \beta_3^y y + \beta_4^y z \\ s_z(x) = \sum\limits_{i=1}^{N} \gamma_i^z \varphi\left(\left\|x - x_{k_i}\right\|\right) + \beta_1^z + \beta_2^z x + \beta_3^z y + \beta_4^z z \end{array} \right\} \tag{16}$$

The RBF method has several advantages that make it very attractive for mesh smoothing. The key point is that being a meshless method only grid points are moved regardless of which elements are connected to them and it is suitable for parallel implementation. In fact, once the solution is known and shared in the memory of each calculation node of the cluster, each partition has the ability to smooth its nodes without taking care of what happens outside because the smoother is a global point function and the continuity at interfaces is implicitly guaranteed. Furthermore, despite its meshless nature, the method is able to exactly prescribe known deformations onto the surface mesh: this effect is achieved by using all the mesh nodes as RBF centres with prescribed displacements, including the simple zero field to guarantee that a surface is left untouched by the morphing action.

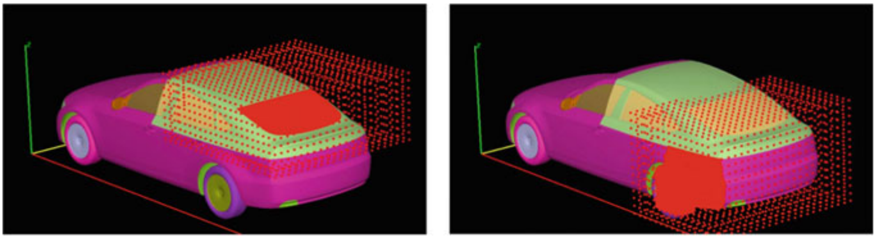
3.2 RBF Morph Tool

The industrial implementation of the RBF mesh morphing poses two challenges: the numerical complexity related to the solution of the RBF problem for a large number of centers and the definition of suitable paradigms to effectively control shapes using RBF. The software RBF Morph allows to deal with both as it comes with a fast RBF solver capable to fit large dataset (hundreds of thousands of RBF points can be fitted in a few minutes) and with a suite of modeling tools that allows the user to set-up each shape modification in an expressive an flexible way. A detailed description of the usage of RBF Morph for the external aero optimization of the Volvo XC60 is given in [10] where the 50:50:50 approach demonstrates how 50 different shape variations can be explored using an high fidelity 50 millions cells mesh in less than 50 wall clock hours.

RBF Morph allows to extract and control points from surfaces and edges, to put points on primitive shapes (boxes, spheres and cylinders) or to specify them directly by individual coordinates and displacements. Primitive shapes can be combined in a Boolean fashion and allow to limit the action of the morpher itself. Two shape modifications used in this study are represented in Fig. 1. It is worth noticing that the shape information coming from an individual RBF set-up are generated interactively with the help of the GUI and are used subsequently in batch commands that allows to combine many shape modifications in a non linear fashion (non linearity occurs when rotation axis are present in the RBF set-up).

3.3 Coupling of RBF Mesh Morphing with Adjoint Sensitivities

Once the adjoint sensitivities are available as surface mesh information it is possible to easily compute the sensitivities w.r.t. shape parameters exploiting the parametric mesh available using the mesh morphing tool. In order to take into account the non linear fashion of the morphing field the mesh deformation velocities are generated



(a) rear window

(b) boat tail

Fig. 1 Example of RBF points arrangement for the definition of two shape parameters. The rear window angle is controlled imposing a rigid rotation to the nodes on the window whilst preserving the shape of the roof and gently deforming the tail. The morphing volume is limited by a Box Encapsulation. Boat tail angle is changed applying a rotation around a proper axis of part of the rear car whilst preserving the shape of the wheel; also in this case the morphing action is limited by a box

by numerical differentiation of the morphing field around the current design point in the parametric space. For a given set of shape parameters the morpher is capable to update the baseline mesh into the current one. A perturbed mesh, w.r.t. the current one, can then be obtained for each shape parameter, computing the mesh resulting from its perturbation (keeping all the other constant). The sensitivity w.r.t. each given parameter is then obtained multiplying the surface perturbation field by the surface sensitivities. It is worth noticing that the aforementioned coupling works not just at the origin of the parametric space (baseline model) but at any given design point; adjoint data need to be recomputed for each explored design point for which local sensitivities are required. The coupling can be used to enrich DOE based exploration for the parametric shape; in the industrial application presented herein, the parameters sensitivities are used in a local optimization method based on the gradient.

4 Optimization Algorithm

The gradient-based algorithm used to minimize the drag force is described in brief below:

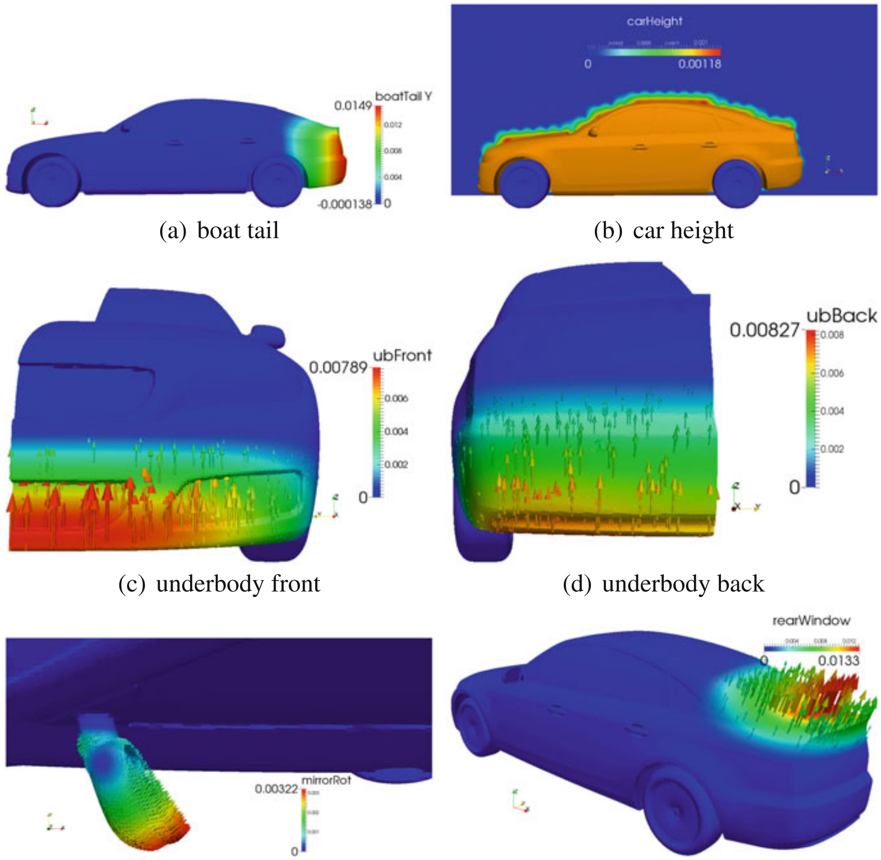
1. Define the shape modification parameters, Sect. 3.
2. Solve the flow equations, Eq. 1.
3. Compute the drag force value, $F_D = \int_{S_W} \left(-\tau_{ij} + p\delta_i^j\right) n_j r_i dS$, $\mathbf{r} = [0, 0, 1]^T$.
4. Solve the adjoint equations, Eq. 7.
5. Compute the deformation velocities and through them, the sensitivity derivatives, Eq. 8.
6. Update the design variables by $\Delta b_i = -\eta \delta F / \delta b_i$, where η is a user-defined step.
7. Morph the car surface and displace the interior mesh nodes.
8. Unless the stopping criterion is satisfied, go to step 2.

5 Applications

In this section, the optimization algorithm presented in in Sect. 4, in the form of an automated software, is used to minimize the drag force exerted on the surface of the DrivAer car model. In specific, the fast-back configuration with a smooth underbody, with mirrors and wheels (F_S_wm_ww) is used as a test case.

Six shape deformation variables (design variables) are defined in total. The part of the car surface parameterized by each of them and the corresponding deformation velocities are depicted in Fig. 2.

The minimization of the drag force is targeted by simultaneously varying all shape deformation parameters. A computational grid of approximately 3.8 million cells is used and turbulence is modeled by means of the Spalart–Allmaras model with wall



(a) boat tail

(b) car height

(c) underbody front

(d) underbody back

(e) mirror rotation

(f) rear window

Fig. 2 DrivAer shape optimization: Part of car surface controlled by the six shape deformation parameters and the corresponding deformation velocities ($\delta x_k/\delta b_n$)

functions. Even though the flow around a car varies in time, the steady state primal and adjoint PDEs are solved, to avoid the practical difficulties faced when solving the unsteady adjoint equations in medium and large scale computational grids [19], by proceeding backwards in time. Hence, the objective function cannot reach a constant value within each optimization cycle but oscillates around a "mean" value. The evolution of the objective function value during the flow solver iterations over the optimization cycles is shown in Fig. 3.

The cumulative deformation magnitude after 15 optimization cycles, which led to a reduction by more than 7% in the mean drag value, is shown in Fig. 4. The pressure field plotted over the initial and optimized geometries is depicted in Fig. 5.

As expected, the area with the highest deformation is located, in the rear part of the car. In specific, two major trends are present. The first one is to lower the height of the rear window and to form a sort of a spoiler at the end of the trunk. This creates an area of increased pressure at the bottom of the rear window and despite the increased pressure on top of the formed spoiler, a resultant force that pushes the car forward is generated, Fig. 6. The second trend is to create a "boat tail" effect (see Fig. 4c) which leads to an increased pressure in the back side of the car, contributing thus to drag reduction.

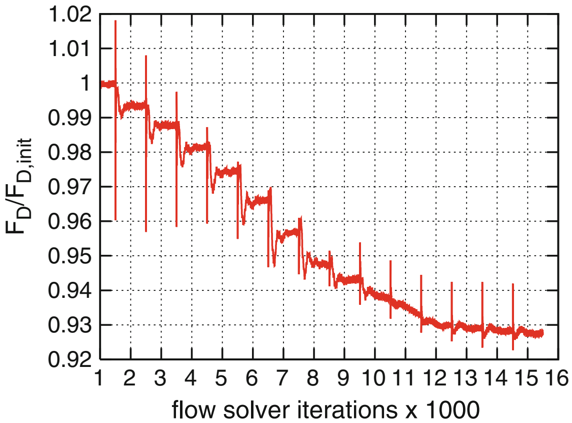


Fig. 3 DrivAer shape optimization: Evolution of the drag force value through the iterations of the flow solver over the optimization cycles, normalized with the mean value obtained using the baseline geometry. The flow solver run for 1000 iterations for each optimization cycle (a previously "converged" solution was used to initialize the optimization, so only 100 iterations were executed during the first optimization cycle). Kinks in the objective function value correspond to the first iterations after each shape update (new optimization cycle). A decrease of more than 7% can be observed in the "mean" drag value

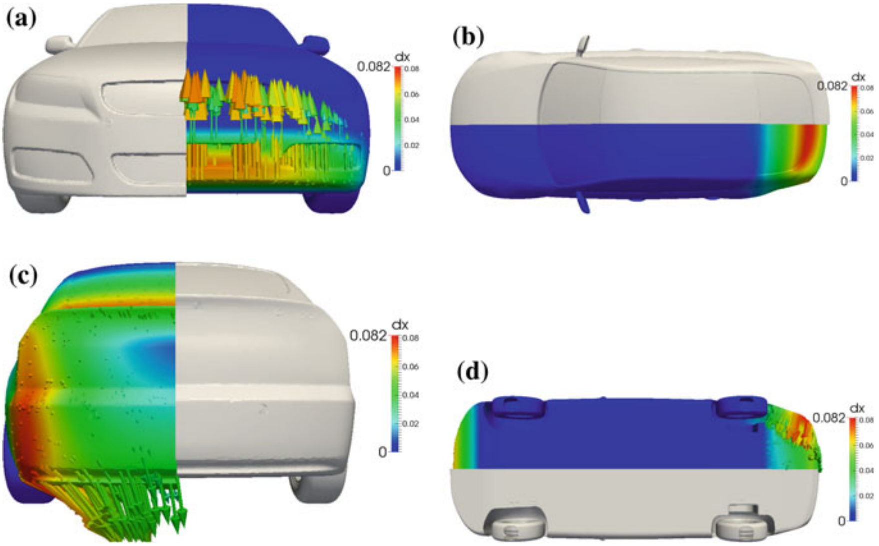


Fig. 4 DrivAer shape optimization: initial (starboard side) and optimized (port side) geometries. The latter is coloured based on the cumulative deformation of the car surface after 15 optimization cycles. The areas with the highest deformation and, thus, the higher impact on the objective value are the ones affected by the boat-tail and rear-window shape modification parameters

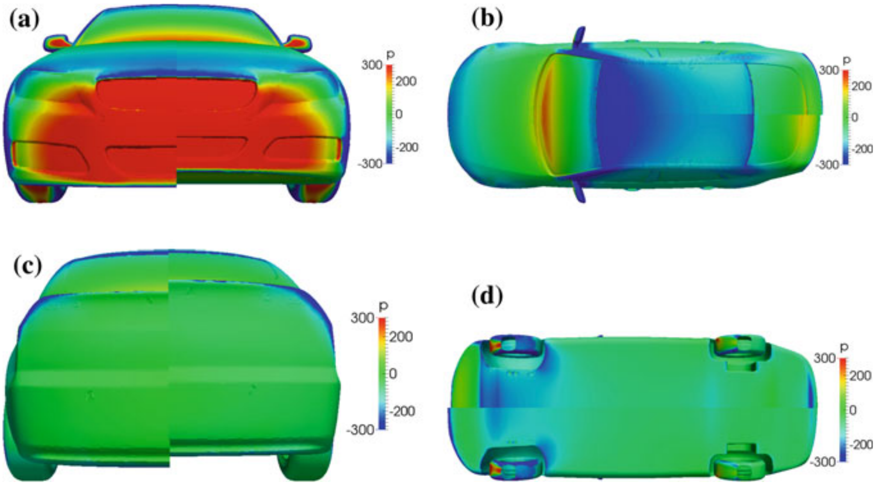


Fig. 5 DrivAer shape optimization: Pressure distribution over the initial (starboard side) and optimized (port side) geometries

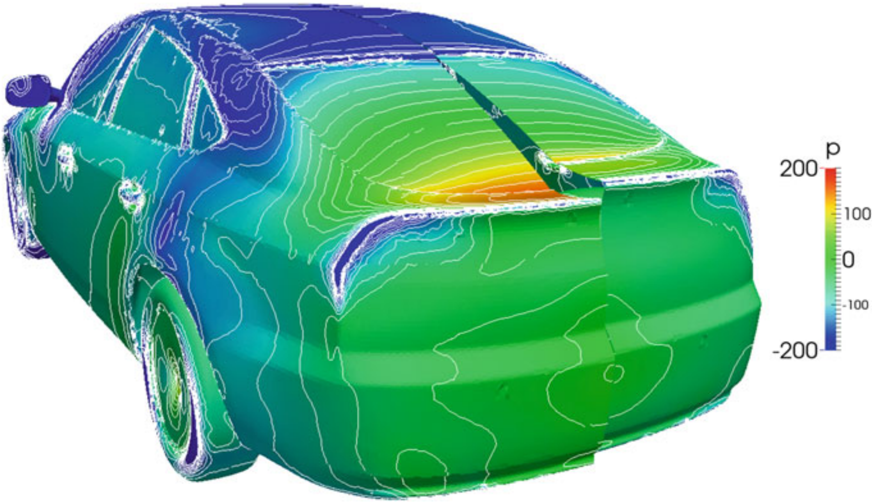


Fig. 6 DrivAer shape optimization: initial (right) and optimized (left) geometries, coloured based on pressure. Lowering the rear window, creating a spoiler at the end of the trunk and creating a boat-tail shape for the rear side lead to an increased pressure at the rear part of the car, contributing to drag reduction

6 Conclusions

In this paper, the continuous adjoint method and an RBF-based morpher, combined into an automated optimization software in the context of a research project funded by the EU, were used as the constituents of a gradient-based optimization algorithm, targeting the drag minimization of the DrivAer generic car model. A significant reduction in the drag value was observed after 15 optimization cycles which required approximately 16 h on 64 Intel(R) Xeon(R) CPUs E5-2630 @2.30 GHz. The utilization of the RBF-based shape modification parameters allowed the design of a smooth and manufacturable car shape.

Acknowledgements This work was funded by the RBF4AERO "Innovative benchmark technology for aircraft engineering design and efficient design phase optimisation" project funded by the EUs 7th Framework Programme (FP7-AAT, 2007–2013) under Grant Agreement no. 605396.

References

1. Beckert, A., Wendland, H.: Multivariate interpolation for fluid-structure-interaction problems using radial basis functions. Constr. Approx. **5**(2), 125–134 (2011)
2. Biancolini, M., Viola, I., Riotte, M.: Sails trim optimisation using CFD and RBF mesh morphing. Comput. Fluids **93**, 46–60 (2014)

3. Biancolini, M.E.: Mesh morphing and smoothing by means of radial basis functions (RBF): a practical example using Fluent and RBF Morph. In: Handbook of Research on Computational Science and Engineering: Theory and Practice (2 vol), pp. 347–380
4. Biancolini, M.E.: Fast radial basis functions for engineering applications. Springer (2018). ISBN 978-3-319-75009-5, https://doi.org/10.1007/978-3-319-75011-8
5. Giles, M., Pierce, N.: An introduction to the adjoint approach to design. Flow Turbul. Combust. **65**, 393–415 (2000)
6. Heft, A., Indinger, T., Adams, N.: Experimental and numerical investigation of the DrivAer model. In: ASME 2012 Symposium on Issues and Perspectives in Automotive Flows. Puerto Rico, USA, pp. 41–51 (2012)
7. Jakobsson, S., Amoignon, O.: Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization. Comput. Fluids **36**, 1119–1136 (2007)
8. Joshi, P., Meyer, M., DeRose, T., Green, B., Sanocki, T.: Harmonic coordinates for character articulation. Technical Report
9. Kavvadias, I., Papoutsis-Kiachagias, E., Dimitrakopoulos, G., Giannakoglou, K.: The continuous adjoint approach to the k–ω SST turbulence model with applications in shape optimization. Eng. Optim. (2014) (to appear). https://doi.org/10.1080/0305215X.2014.979816
10. Khondge, A., Sovani, S.: An accurate, extensive, and rapid method for aerodynamics optimization: The 50:50:50 method. SAE Technical Paper **01**(0174) (2012)
11. Martin, M.J., Andres, E., Lozano, C., Valero, E.: Volumetric B-splines shape parametrization for aerodynamic shape design. Aerosp. Sci. Technol. **37**, 26–36 (2014)
12. Micchelli, C.: Interpolation of scattered data: Distance matrices and conditionally positive definite functions. Constr. Approx. **2**(1), 11–22 (1986)
13. Nadarajah, S., Jameson, A.: Studies of the continuous and discrete adjoint approaches to viscous automatic aerodynamic shape optimization. AIAA Paper **25**(30) (2001)
14. Papoutsis-Kiachagias, E., Giannakoglou, K.: Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: industrial applications. Arch. Comput. Methods Eng. (2014). https://doi.org/10.1007/s11831-014-9141-9
15. Papoutsis-Kiachagias, E., Kyriacou, S., Giannakoglou, K.: The continuous adjoint method for the design of hydraulic turbomachines. Comput. Methods Appl. Mech. Eng. **278**, 612–639 (2014)
16. Robinson, T., Armstrong, C., Chua, H., Othmer, C., Grahs, T.: Optimizing parameterized cad geometries using sensitivities based on adjoint functions. Comput. Aided Des. Appl. **9**(3), 253–268 (2012)
17. Spalart, P., Jou, W., Stretlets, M., Allmaras, S.: Comments on the feasibility of LES for wings and on the hybrid RANS/LES approach. In: Proceedings of the first AFOSR International Conference on DNS/LES (1997)
18. Thompson, P., Robinson, T., Armstrong, C.: Efficient CAD-based aerodynamic design optimization with adjoint CFD data. In: 21st AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-located Conferences (2013)
19. Vezyris, C., Kavvadias, I., Papoutsis-Kiachagias, E., Giannakoglou, K.: Unsteady continuous adjoint method using POD for jet-based flow control. In: 11th World Congress on Computational Mechanics, ECCOMAS. Barcelona, Spain (2014)
20. Zymaris, A., Papadimitriou, D., Giannakoglou, K., Othmer, C.: Continuous adjoint approach to the Spalart-Allmaras turbulence model for incompressible flows. Comput. Fluids **38**(8), 1528–1538 (2009)
21. Zymaris, A., Papadimitriou, D., Giannakoglou, K., Othmer, C.: Adjoint wall functions: a new concept for use in aerodynamic shape optimization. J. Comput. Phys. **229**(13), 5228–5245 (2010)

Part IV
Holistic Optimization in Marine Design

Upfront CAD—Parametric Modeling Techniques for Shape Optimization

S. Harries, C. Abt and M. Brenner

Abstract The paper presents an overview of parameter-based geometric modeling as used for shape optimization with respect to fluid-dynamic performance. Parametric modeling is well established in Computer Aided Design, particularly in the phases of detailed design and production. However, production-centric models often require considerable effort, e.g. de-featuring, to prepare them for simulation, above all for Computational Fluid Dynamics. Consequently, to investigate a large number of design variants special engineering models are built, deliberately omitting certain details, especially if they cannot be captured by the simulation within reasonable effort anyway. In the context of aero- and hydrodynamic design dedicated parametric models are utilized that define shapes of high quality with as few parameters as possible. Parametric modeling for shape optimization can be subdivided into fully-parametric and partially-parametric modeling. In fully-parametric modeling the entire shape is defined and realized by means of parameters while in partially-parametric modeling only the changes to an existing shape are described parametrically. Prominent techniques of partially-parametric modeling are free-form deformation, shift transformations and morphing. The most popular techniques are summarized, giving some of their mathematical background while discussing advantages and drawbacks. Examples are drawn from the maritime and aerospace industries, turbomachinery and automotive design.

S. Harries (✉) · C. Abt · M. Brenner
FRIENDSHIP SYSTEMS AG Benzstr. 2, 14482 Potsdam, Germany
e-mail: harries@friendship-systems.com

C. Abt
e-mail: abt@friendship-systems.com

M. Brenner
e-mail: brenner@friendship-systems.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_12

1 Introduction

In product development the potential of gaining a benefit by changing a design is highest, not surprisingly, when undertaken early in the process. The later a problem becomes apparent—and the more advanced a product is defined already—the higher the expense and time required to fix it. Competition and risk mitigation increasingly call for optimization as early as the concept and preliminary design phases. Those products whose success depends critically on flow performance are more and more analyzed by means of Computational Fluid Dynamics (CFD) not only when fine-tuning the final shapes but already upfront when taking critical decisions.

Shape optimization of components and systems play a major role in the maritime, aerospace, turbomachinery and automotive industries. At one end of the spectrum the maritime industry is characterized by expensive one-off designs while at the other end the automotive industry frequently sees large series. Still, a common denominator is that small concerted changes in shapes often lead to substantial improvements in performance. Furthermore, even small improvements often yield important benefits for the producer, the consumer and the environment, e.g. when reducing energy consumption and emissions.

Clearly, the more variants are studied the better the chances of finding design candidates that are outstanding. In order to study many variants, the process of creating and analyzing them has to be streamlined. A key to success are engineering models that capture the essence of the product while deliberately omitting less important details. More specifically, the Computer Aided Design (CAD) models need to be simulation-ready. Otherwise, lots of engineering resources go into individually preparing each single variant for subsequent investigations. For flow-related shapes—both for external and internal flows—special parametric modeling techniques have been developed to enable meaningful modifications with small sets of parameters to control them.

Flow-related shapes, also known as functional surfaces [14], are typically characterized by compound curvature. They can rarely be described by simple sweeps or modeled as developable surfaces. Consequently, for flow-related shapes boundary representation methods (B-reps) dominate over constructive solid geometry (CSG). B-reps, furthermore, often serve as input to CSG in hybrid CAD models.

Therefore, within this paper focus will be put on B-reps, addressing parameter-based geometric modeling for shape optimization, so-called "upfront CAD." It gives an overview of techniques, shows examples and looks at advantages and drawbacks. It is important to note that no single method is considered the "Holy Grail" of parametric modeling. Rather, the available techniques are seen as tools within an engineering tool box that should be selected thoughtfully according to the task at hand.

2 Overview

Parametric modeling is the definition of a product (or the representation of system behavior) by means of important descriptors. For shape optimization using CFD special parametric models are needed, so-called engineering models, which describe the product with as few significant parameters as possible, sometimes deliberately leaving out characteristics that are of lesser importance for a specific design task (e.g. fillets and bevels). Two major categories of parametric modeling are distinguished: Fully-parametric modeling (FPM) and partially-parametric modeling (PPM) [1, 5, 7]. For surveys, primarily on PPM, see [9, 13, 15].

In fully-parametric modeling the entire shape is defined by means of parameters. A hierarchical model is created in which parameters describe all features of the envisioned product as required to quantify product performance. A parametric model can be looked at as a system that takes parameters as input and produces a shape—i.e., an instance—as an output. Any shape is realized from scratch and variants are brought about by simply changing the values of one or several of the inputs.

In partially-parametric modeling an existing shape—i.e., the initial shape or baseline—is taken and changes are defined by parameters that then modify the baseline. This results in an image that features much of the baseline's characteristics but looks different. The baseline may stem from any previous modeling process, for example from a traditional CAD system. Prominent representatives of partially-parametric modeling are free-form deformation (FFD), shift transformations and morphing.

There is a compromise to make with regard to flexibility to model and efficiency to vary a shape. Figure 1 gives an overview. The traditional approach of modeling shapes with various surface types (e.g. Bézier and B-splines, Coons patches) controlled by point sets is very flexible. Any geometry can be represented even though it may take the CAD engineer quite some time to establish the final shape.

If just one (instance of the) shape is needed this traditional path is often the quickest. However, if many variants shall be investigated the additional investment of setting up a suitable parametric model is worthwhile to spend. For optimization, fully-parametric modeling is very powerful since it enables both large changes in the early design phase and small adjustments when fine-tuning the shape at a later point in time. Depending on the complexity of the product (and the quality requirements needed for the simulations) some lead time is required to establish a fully-parametric model. Since a fully-parametric model is made with a dedicated purpose in mind it is rarely applicable to many different design tasks (e.g. you cannot create a slender super-yacht with the model for a high-blockage tanker). Partially-parametric modeling can be viewed as a trade-off between traditional modeling and fully-parametric modeling.

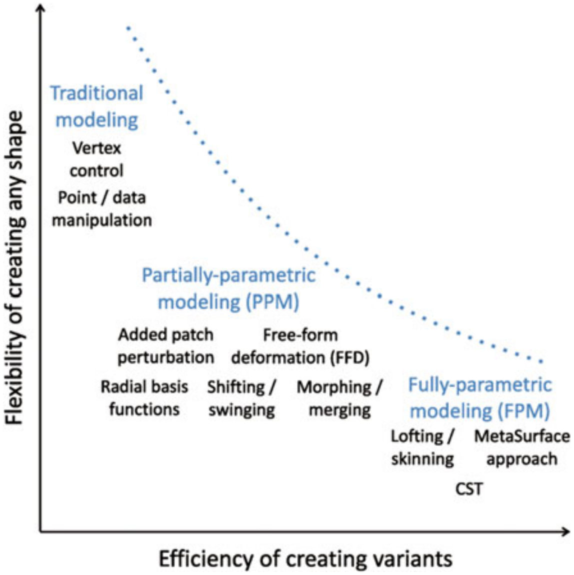

Traditional modeling
Vertex control
Point / data manipulation
Partially-parametric modeling (PPM)
Added patch perturbation
Free-form deformation (FFD)
Radial basis functions
Shifting / swinging
Morphing / merging
Fully-parametric modeling (FPM)
Lofting / skinning
MetaSurface approach
CST
Flexibility of creating any shape
Efficiency of creating variants

Fig. 1 Flexibility versus efficiency in geometric modeling

3 Partially-Parametric Modeling (PPM)

There are quite a few partially-parametric modeling techniques that have gained popularity in shape optimization. Very prominent is free-form deformation which was first introduced in the mid 1980s [14]. Some approaches go back several decades and were originally not called partially-parametric modeling. In the maritime industry, for example, the swinging of the sectional area curve by shifting transversal sections has been used regularly (and with great success) since the early 1950s [2, 11].

In general, partially-parametric modeling is applicable to both continuous (e.g. a mathematically-closed surface definition) and discrete data (e.g. tri-meshes for STL data exchange). For shape optimization this means that a modification can be applied to the geometry only, requiring subsequent re-meshing for the simulation, or that the changes are made directly to the CFD mesh (cp. Sect. 6), necessitating some reverse engineering of the final design at the end.

3.1 Free-Form Deformation

In free-form deformation (FFD) [9, 14], sometimes simply called box deformation or space deformation, the geometry to be modified is enclosed by a regular grid of vertices, i.e., rows, columns and layers defining a B-spline volume $\overrightarrow{Q}(u, v, w)$.

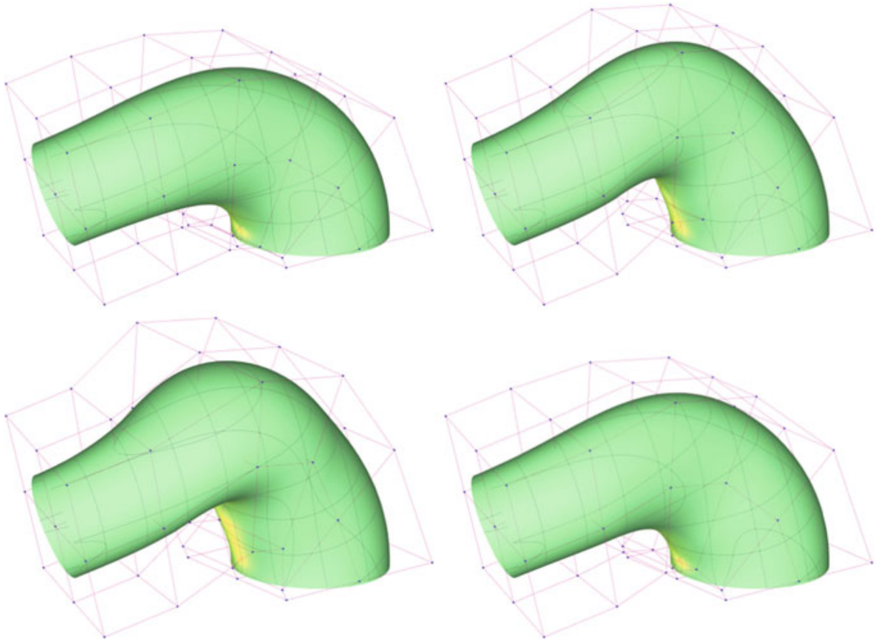


Fig. 2 Free-form deformation for curved diffusing duct

For any point $\overrightarrow{P} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$ of the initial shape that lies within this lattice the triple (u, v, w) of the B-spline volume is determined that yields the point's coordinates in Cartesian space, thus coupling the initial shape to the B-spline. Moving any vertex then changes the B-spline volume and, along with it, the embedded baseline. Figure 2 depicts an example taken from the automotive industry.

The free coordinates of the vertices serve as parameters. Theoretically, the FFD has as many degrees-of-freedom (DoF) as the number of vertex coordinates that are allowed to change. A box of $3 * 3 * 6$ vertices as shown in Fig. 2 would give $54 * 3 = 162$ DoF. Practically, the actual DoF are reduced by (a) freezing quite many vertices and (b) orchestrating the modification of several vertices.

Freezing vertex coordinates is done at symmetry planes and, furthermore, so as to ensure geometric continuity. If the first layer of vertices is left unchanged positional integrity (G^0) is maintained. If the first two layers are kept the modified shape features a tangent-continuous transition (G^1) to the unmodified part of the baseline. (It should be noted, nevertheless, that continuity will at best be the same for the variant than it was for the initial shape.)

A simple example for an orchestrated change of several vertices is to move all vertices belonging to the same layer by the same distance into the same direction. The variation illustrated in Fig. 2 shows such a concerted change of all vertices of the third and fourth layers in the middle of the box while the first two and the last two layers are fixed.

3.2 *Shift Transformations*

Shift transformations are realized by moving any point of the initial shape by a specified amount in the principle directions of the chosen coordinate system. Often this is done in Cartesian space. Each point $\overrightarrow{P} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$ receives its displacement Δx, Δy and/or Δz depending on its original position and the specified shift function.

Figure 3 illustrates a vertical shift for the bulbous bow of a cargo ship. The baseline's bulb is shifted upwards and downwards as shown in Fig. 3. The amount of displacement is given by the curves (shown in red), starting with a zero value at the keel line (close to the forward perpendicular) and ending with the maximum displacement at the bulb's tip.

In the example a cubic B-spline curve was used to define the shift function. Here, $\Delta z = f(x)$ in which $f(x)$ is simply the z-component of the B-spline curve at position x. The z-value of the B-spline's last vertex served as the only parameter (denoted "moveBulb" in Fig. 3).

The geometric continuity of the shift function influences the quality of the variation. Choosing a horizontal tangent at the transition from the unmodified to the modified part of the shape (see Fig. 3) ensures a tangent-continuous shift (G^1). Zero curvature of the shift function at the point of transition would yield a curvature-continuous shift (G^2) and so forth.

Shift transformations can be concatenated, directly summed up and/or multiplied to form complex modifications. Figure 4 gives the superposition of shifts in vertical, transversal and longitudinal directions (shown in green). The black lines correspond

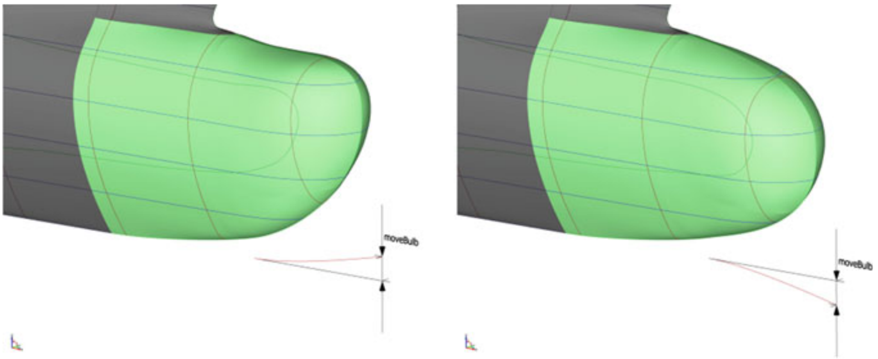


Fig. 3 Vertical shift of geometry for a bulbous bow

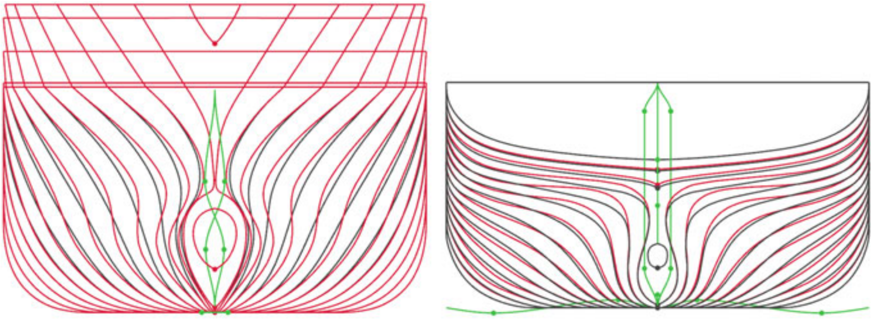


Fig. 4 Concerted shift for a complete ship hull form

to the sections of the baseline, here the same hull form as used for the bulb modification in Fig. 3, while the red lines depict the sections of the variant.

Shift functions are often made of curves but can also be derived from surfaces. In the latter case the shift becomes a function of two coordinates. For example, $\Delta y = f(x, z)$ describes a transversal shift as a function of the longitudinal and vertical coordinates x and z.

3.3 Added Patch Perturbation

Added patch perturbation can be looked at as a generalization of a surface shift. A new surface patch, say $\vec{A}(u, v)$, is added to the surface(s) of the initial shape, say $\vec{S}(u, v)$. Simply, any new variant is given by $\vec{N}(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} = \vec{S}(u, v) + \vec{A}(u, v)$.

Theoretically, a superposition in parameter space according to (u, v) is more flexible than a surface shift undertaken in physical space (x, y, z). Practically, some preparatory work may be necessary to define sub-surfaces or poly-surfaces for the baseline such that the (u, v)-spaces of the surfaces align as needed.

Parameters of this partially-parametric model are those used to control the added patch. A popular approach is to take a standard B-spline surface and select several of its inner vertices for change, keeping vertices at the edge(s) fixed. Often, only one or two of the vertex coordinates are modified [12].

3.4 Morphing

Morphing, sometimes also called merging, is the interpolation (possibly the extrapolation) between two or more baselines. The baselines need to be topologically identical in order to easily compute a new variant which is done by a linear superposition of all initial shapes $\vec{S}_i$.

For two baselines a one-parameter model is established, i.e., $\vec{N}(u, v) = (1 - w) \cdot \vec{S}_1(u, v) + w \cdot \vec{S}_2(u, v)$ where $w \in [0, 1]$. Three baselines provide a two-parameter model etc.

Figure 5 shows a sequence of four variants for a duct. The upper left duct (light turquois) represents the first baseline, the lower right duct (dark green) the second. The intermediate shapes are produced for $w = 0.3$ (upper right bluish duct) and $w = 0.7$ (lower left greenish duct). The geometry for the inlet and the outlet are the same for both baselines, giving the same inlets and outlets for all variants.

A practical approach taken in shape optimization is to generate a family of initial shapes based on experience and inspired by intuition. Subsequently, within the optimization the best mix of the baselines is identified [8].

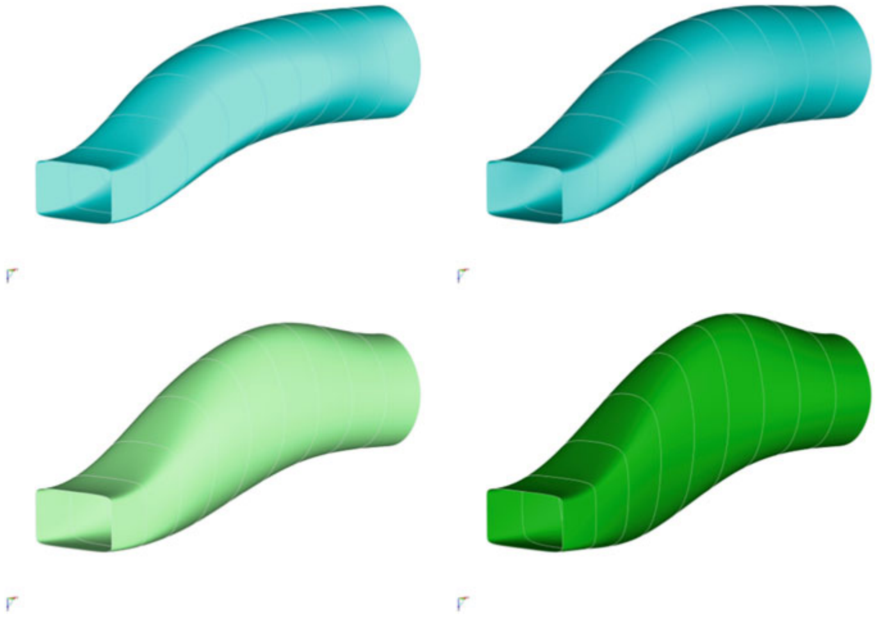


Fig. 5 Morphing of a duct

3.5 Radial Basis Functions

A radial basis function (RBF) is a function whose value is directly coupled to its distance from a chosen center point. The value of the RBF tapers off from its maximum at the center towards zero in all coordinate directions. Although being popular primarily for regression analysis, it also is utilized as a mechanism to change geometry smoothly within a confined region.

An example radial basis function is a cos-square function of amplitude A and radius r, see for instance [1]. The center of the radial basis function is placed at some arbitrary point in space. At this point the change to the original geometry is largest, namely equal to the amplitude A. All points within the radius r also experience a shift, albeit with decreasing magnitude for increasing Euclidian distance to the center point. Beyond radius r no more changes are introduced and the geometry remains as-is.

Figure 6 illustrates this for the nose of an airplane. The modification is applied to a discrete data set that stems from an STL description of the geometry. Here, just one RBF is used whose center point is located in the tip. More complex modifications can be realized by combining several RBFs, see e.g. [15].

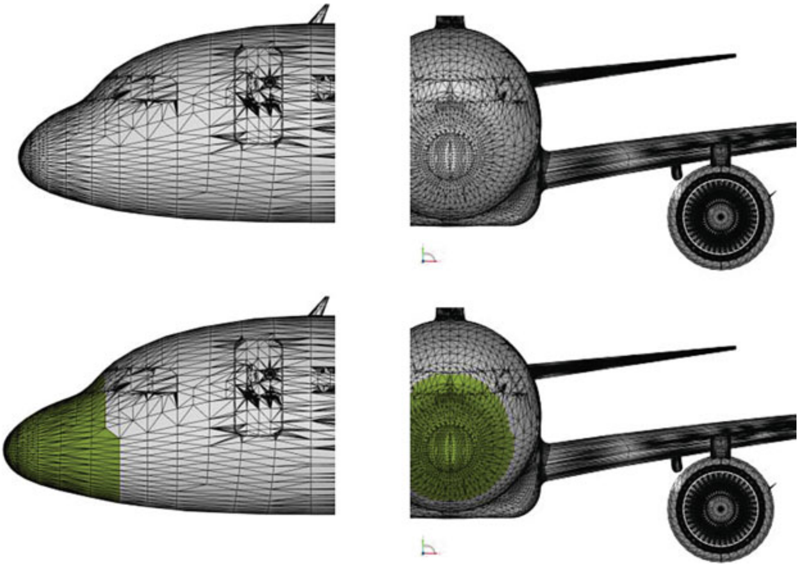


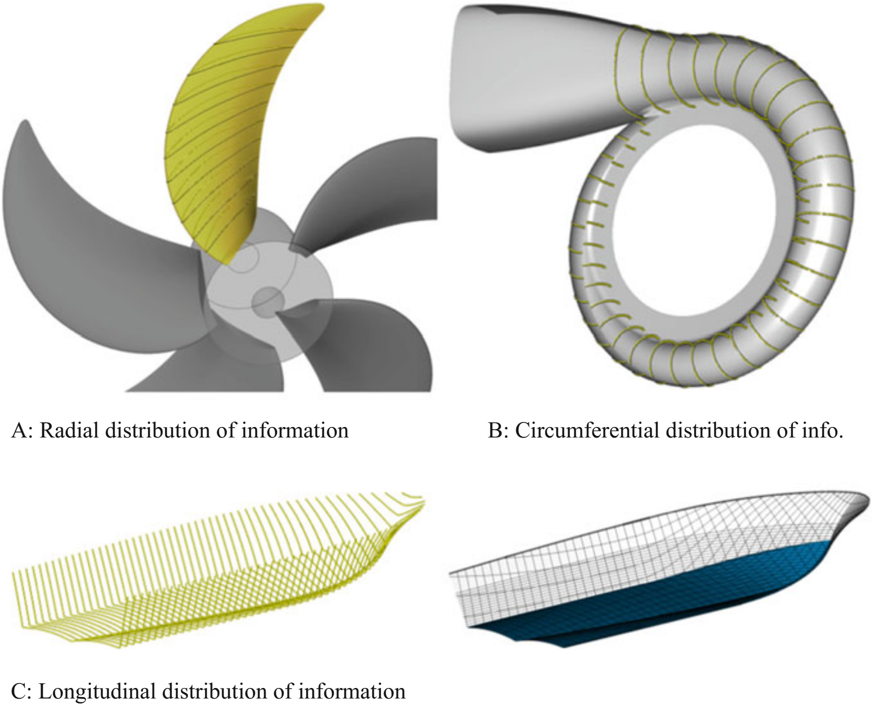
Fig. 6 Modification of an airplane nose by a single RBF

4 Fully-Parametric Modeling (FPM)

Fully-parametric models are built from scratch, i.e., the entire shape is defined by parameters and each variant is generated as an instance according to the current parameter values.

When looking at flow-related shapes often there are two distinct directions of information. In one direction a design changes quite slowly, it rather evolves, while in the other direction, mostly orthogonal to the first, there is a certain building pattern that remains topologically unchanged. This is illustrated for different products in Fig. 7.

As shown in Fig. 7a the blades of a propeller do not change significantly from hub to tip. Rather, the profiles are continuously defined by the same parameter set, typically chord length, maximum thickness and camber along with pitch, rake and skew (see also below). Figure 7b depicts the sections of a volute where the building pattern stays the same circumferentially even though from one angle to the next small modifications occur. Likewise, the sections of a hull as shown in Fig. 7c change only gradually in longitudinal direction when moving from stern to stem (see also Fig. 4



A: Radial distribution of information

B: Circumferential distribution of info.

C: Longitudinal distribution of information

Fig. 7 Gradual change of information for different shapes

for a different type of ship). From one station to the next the modifications are minor while over the entire length of the vessel quite some differences occur.

4.1 MetaSurface Approach for Fully-Parametric Modeling

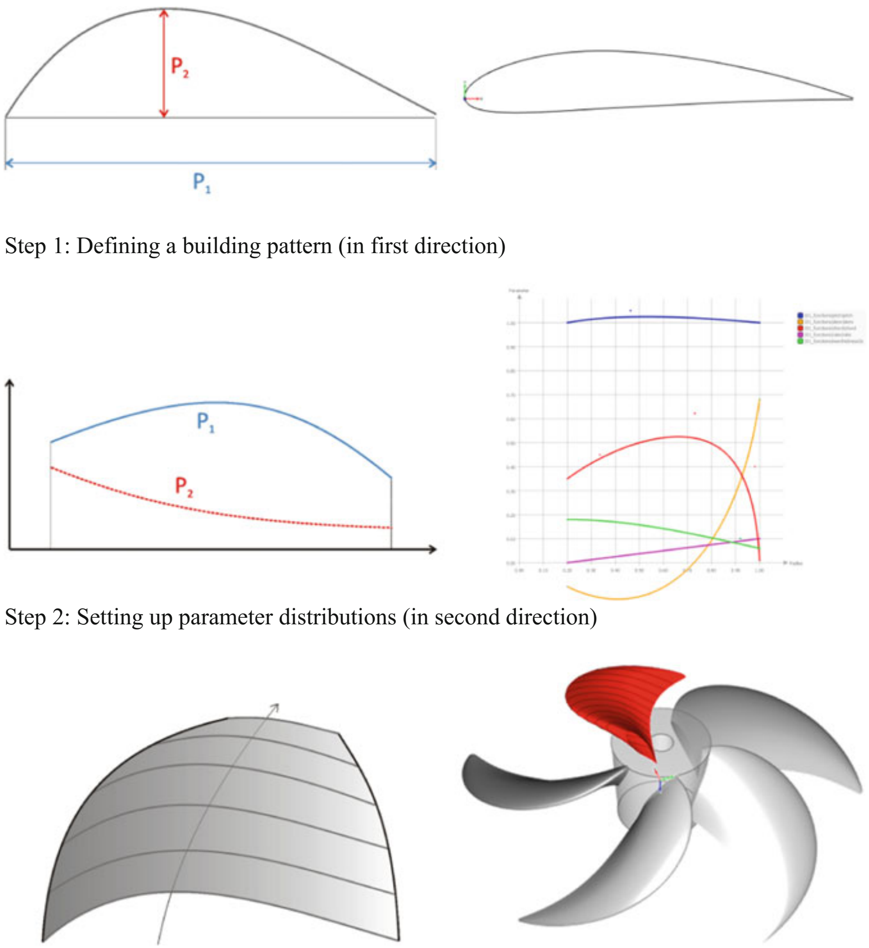
A powerful approach is to define both the building pattern (curve definition) and the distribution of the inputs to this pattern in the other direction parametrically. Figure 8 illustrates the superposition of information in an abstract form (left column) and specifically for a propeller blade (right column). The outcome is a mathematically closed definition, a so-called MetaSurface [4], that allows computing and addressing any point on the surface.

Figure 9 illustrates this MetaSurface approach for a propeller. The blade is produced from a chosen profile, here a modified NACA66 as the building pattern, along with the radial distributions of all parameters defining the profile and its position in space. As can be seen each parameter changes as a function of the radius. Every cylindrical profile is fully defined by the values of all parameters at any given radius, bringing about a closed mathematical description. In Fig. 9 the radial distribution for chord length (shown in red) is changed, here by just one parameter, resulting in blade variants.

The idea of combining information from two distinct directions is obviously related to sweeping, i.e., the continuous transfer of a curve (often planar) along a given path. However, typical sweeps either keep the generating curve as is or just apply standard transformations (such as scaling and rotation) along the path. This brings about a fully-parametric model, too, but with less control.

Another similar approach of surface generation is lofting (likewise skinning), i.e., the longitudinal interpolation of a set of transversal curves by means of a surface. This also constitutes a fully-parametric model. However, each transversal curve is treated individually, i.e., each curve is generated from its specific parameters. An important difference to the MetaSurface approach is that the values of the input parameters are not necessarily related. This independence leads to higher degrees-of-freedom (DoF) and, quickly, to oscillations in stacking direction, in particular when each transversal curve is handled separately during an automated optimization.

Suppose that within a lofting approach the propeller blade shown in Fig. 9 would be generated from twelve individual cylindrical profiles stacked from hub to tip. With only five parameters per profile the design space would already feature $12 * 5 = 60$ DoF. In comparison, the design space of the corresponding MetaSurface (Fig. 8) is more compact as it is spanned only by the parameters that control the radial distributions. If four parameters are used for each of these curves on average, the design space shrinks to $5 * 4 = 20$ DoF. In practice, during an optimization only subsets of the parameters are employed as free variables. For instance, let the propeller blade be varied for the inner radii by changing chord length (as shown in Fig. 9) and camber. Taking two parameters to modify the chord length and camber distributions would give two DoF for the MetaSurface. For the lofted surface one would choose



Step 1: Defining a building pattern (in first direction)

Step 2: Setting up parameter distributions (in second direction)

Step 3: Combining the building pattern and the parameter distributions (MetaSurface)

Fig. 8 Shape definition by superposition of information given in different directions

maybe six to eight profiles, freezing hub and tip. This would give rise to $6 * 2 = 12$ to $8 * 2 = 16$ DoF.

4.2 Parametric Modeling of Building Patterns

Frequently, building patterns for fully-parametric models (see Fig. 8) are planar shapes since they are easy to both generate and understand.

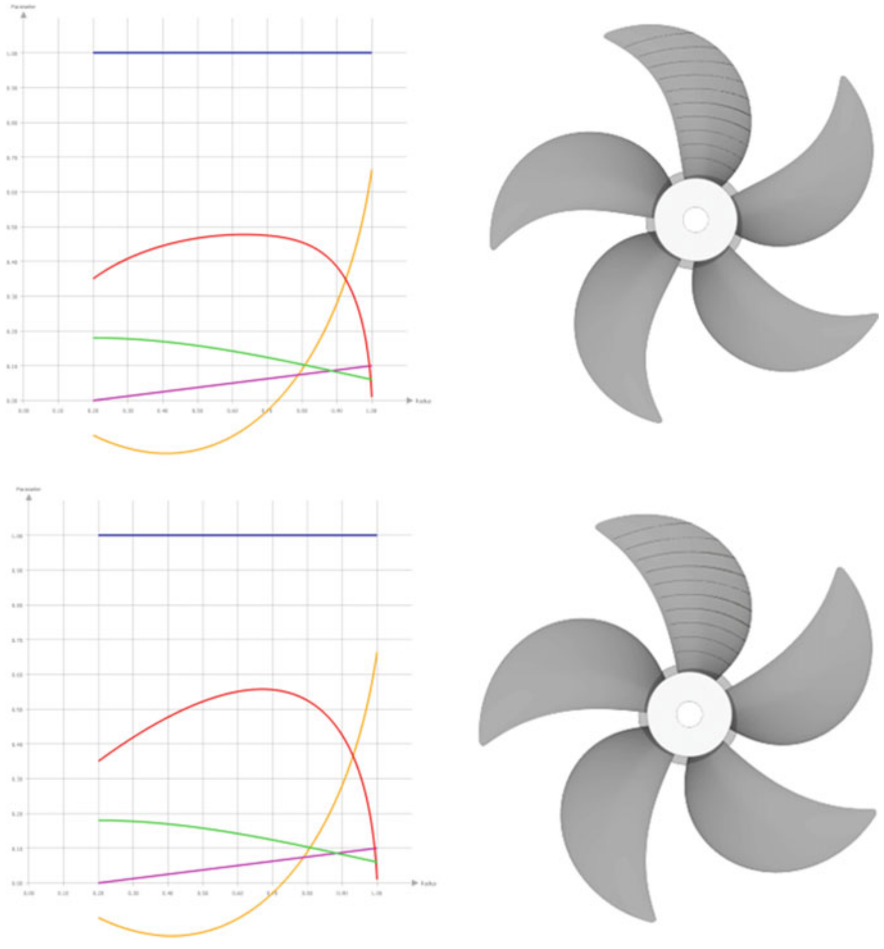
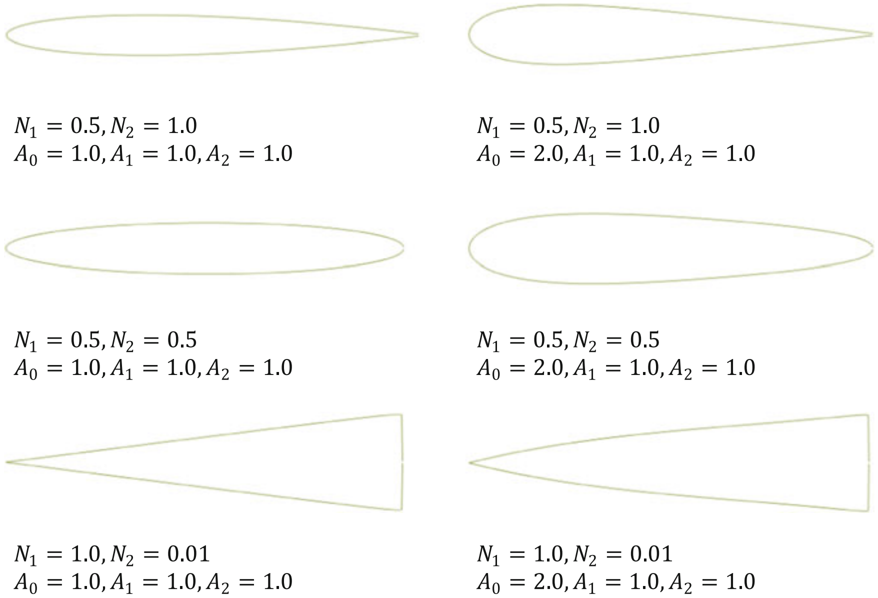


Fig. 9 Variation of chord length distribution for a propeller and resulting blades

For propellers, turbine blades and airfoils often standard profiles are chosen, for instance the NACA66 and NACA four digit series. Some of these shapes have analytical definitions. An option then is to utilize the coefficients as parameters for shape control, particularly if they have a task-specific meaning (e.g. maximum camber).

Another approach is to define profile-like shapes such as airfoil sections, nacelles, ducts and streamline bodies explicitly by the class-and-shape function transformation (CST) as introduced in [10]. The non-dimensional ordinate of a profile is written as $\frac{y}{c} = C \cdot S + \frac{x}{c}\frac{\Delta y_{TE}}{c}$ with the class function $C = \left(\frac{x}{c}\right)^{N1} \cdot \left(1 - \frac{x}{c}\right)^{N2}$ and the shape function $S = \sum_{i=0}^{N} \left[A_i \cdot \left(\frac{x}{c}\right)^i\right]$, $\frac{x}{c}$ being the non-dimensional abscissa. N_1 and N_2 define the fundamental geometry, i.e., the class, while A_i serve as parameters of the actual



$N_1 = 0.5, N_2 = 1.0$
$A_0 = 1.0, A_1 = 1.0, A_2 = 1.0$

$N_1 = 0.5, N_2 = 1.0$
$A_0 = 2.0, A_1 = 1.0, A_2 = 1.0$

$N_1 = 0.5, N_2 = 0.5$
$A_0 = 1.0, A_1 = 1.0, A_2 = 1.0$

$N_1 = 0.5, N_2 = 0.5$
$A_0 = 2.0, A_1 = 1.0, A_2 = 1.0$

$N_1 = 1.0, N_2 = 0.01$
$A_0 = 1.0, A_1 = 1.0, A_2 = 1.0$

$N_1 = 1.0, N_2 = 0.01$
$A_0 = 2.0, A_1 = 1.0, A_2 = 1.0$

Fig. 10 Instances of scaled CST shapes

shape. The shape functions can be cast into different polynomial representations, making parameters A_i easier to interpret.

Figure 10 shows several instances of profiles (mirrored at the x-axis). Three quadratic Bernstein polynomials, featuring partition of unity, are employed for S. Different exponents N_i of the class function (left column) lead to wider metamorphoses while modifications of the shape coefficients A_o, A_1 and A_2 (right column) yield slighter variations.

For free-form shapes often Bézier or B-spline curves are utilized. The coordinates of the vertices are frequently treated as parameters. This has a smoothing effect on the shape when changing vertex positions. Nevertheless, the B-spline vertices are only indirectly related to shape properties such as tangents, curvatures, area and centroid. These more sophisticated parameters frequently have a direct, task-related meaning. If an additional fairing functional is introduced [6] the DoF become independent of the number of vertices and the B-spline is computed from an (inner) optimization, leading to so-called F-splines [4].

Figure 11 illustrates the variation of an F-spline for different inputs. A cubic open B-spline with six vertices is varied for its area (with reference to the ordinate axis) and its tangent at the upper right end. The start and end points along with the tangent at the start point are fixed.

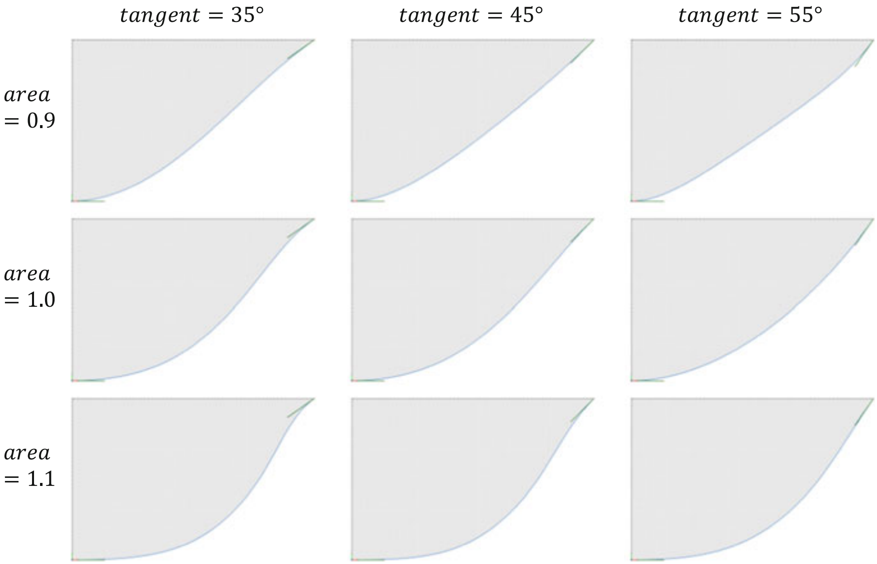


Fig. 11 Instances of a two-parameter family of F-splines

4.3 Incorporated Handling of Constraints

Frequently, shapes have to comply with a set of equality and inequality constraints. While some constraints can only be evaluated by means of a simulation, others may directly relate to the shape itself. These constraints can be utilized such that a feasible instance is produced for each parameter set which nicely speeds up any subsequent optimization.

Fully-parametric modeling is particularly well-suited to incorporate equality constraints. Figure 12 illustrates this for the piston bowl of a Diesel engine. Here the compression volume is supposed to stay constant during the variation of the bowl (highlighted in red). The planar curve, i.e., the building pattern, used to define the surface of revolution is determined such that the specified volume is met automatically for each instance.

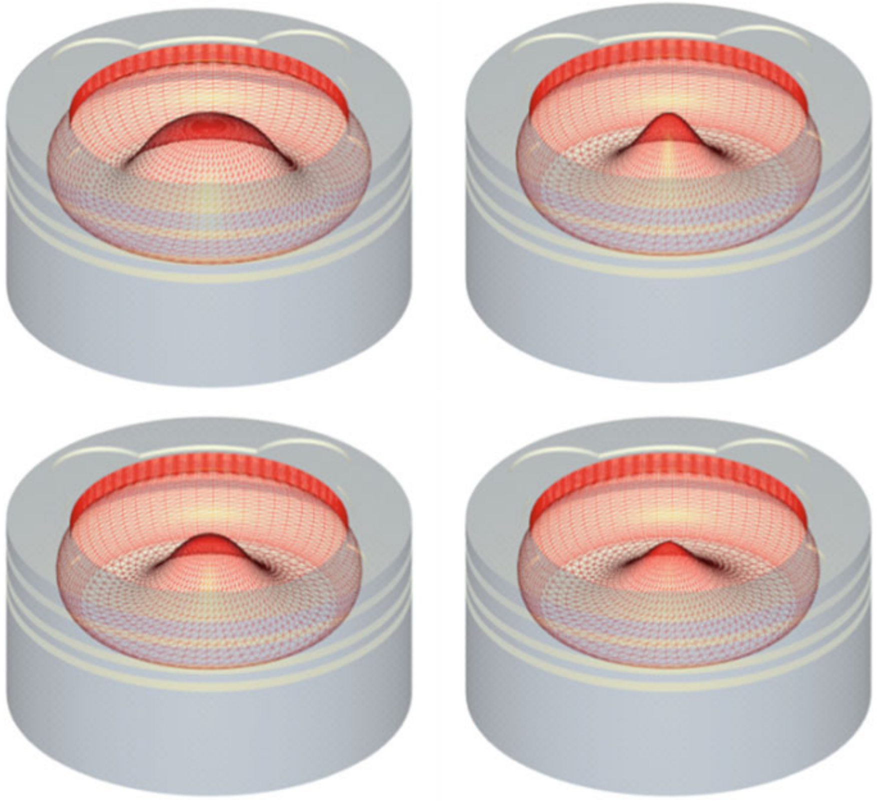


Fig. 12 Four instances of a Diesel piston bowl featuring the same compression volume

5 Design Velocities

Comprehending a sophisticated parametric model is not always an easy task, particularly if the designer is not the person who set up the model or the model was defined some time ago.

A nice way of illustrating the influence of each parameter in order to gain an appreciation of the model is to depict the so-called design velocities. Changing one parameter at a time just slightly allows computing the displacement of each point form the initial to its new position. Figure 13 gives the design velocities for three parameters of a pressure valve as computed by the displacement in normal direction to the initial surface.

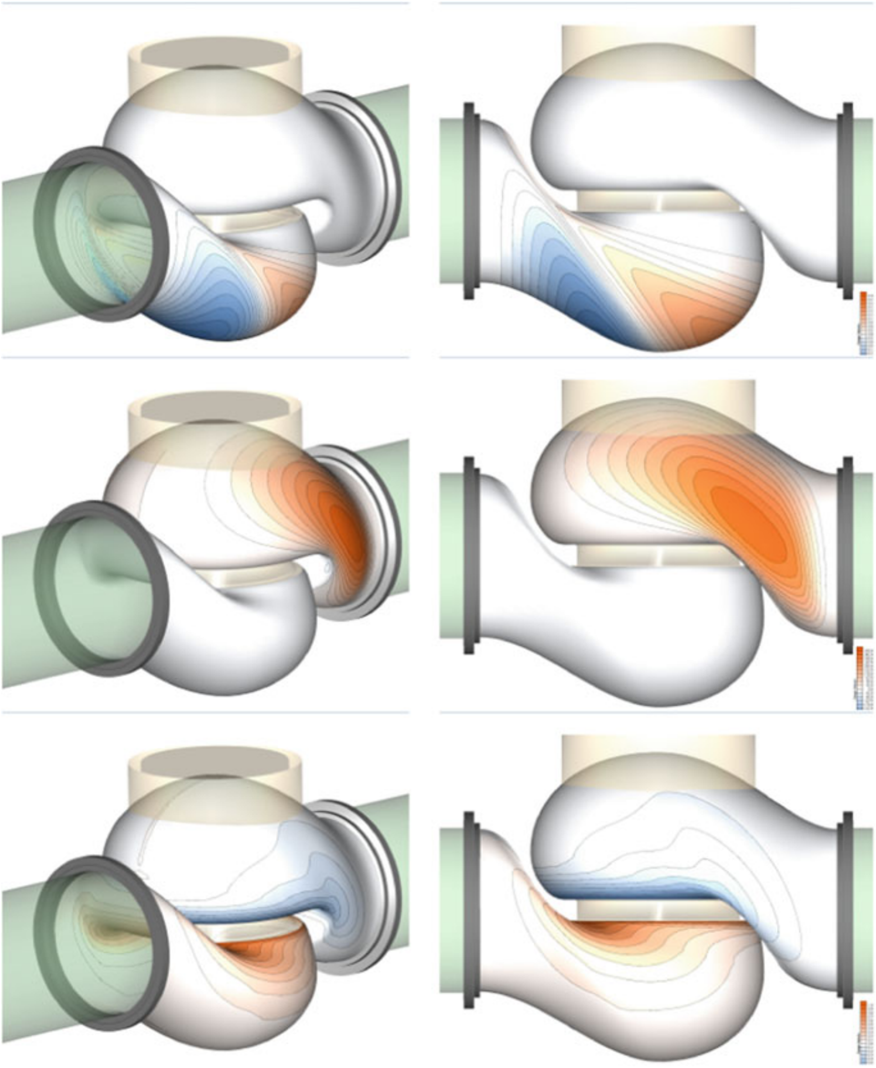


Fig. 13 Design velocities for three parameters of a pressure valve

6 Simulation-Driven Design

Developing a good parametric model requires both experience in geometric modeling and knowledge about the design task at hand. An investment into a good parametric model is worthwhile to make if variants need to be generated quickly and without any need of further interactive work. Consequently, simulation-driven design (SDD)

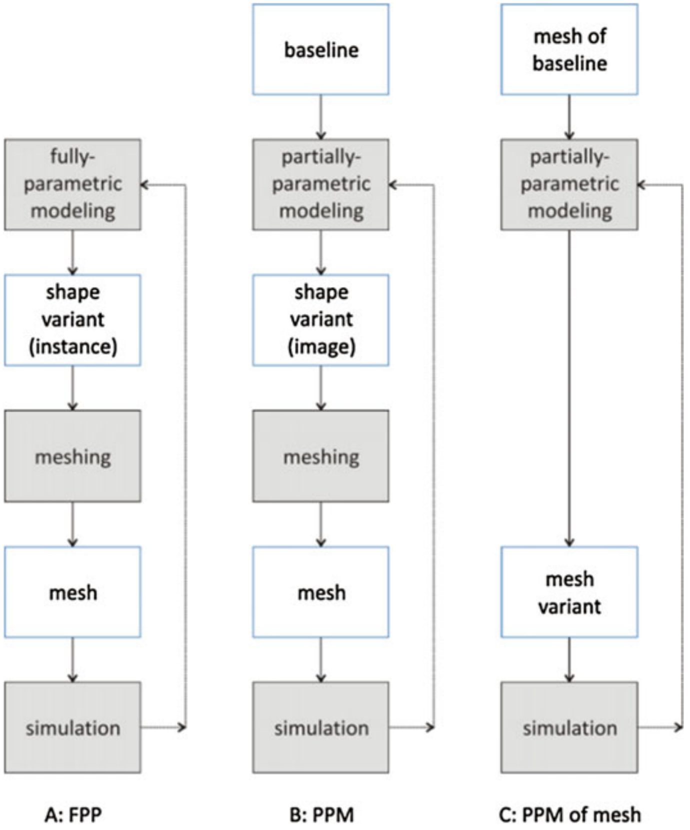


Fig. 14 Steps taken in simulation-driven design depending on modeling approach

is a key field of application for parametric modeling (upfront CAD) since tens, if not hundreds or even thousands of variants are to be studied.

In the context of CFD simulation the flow field typically is discretized by means of a mesh. Depending on the parametric modeling approach, different steps are involved in the process from producing a variant to undertaking and assessing the CFD simulation. Figure 14 summarizes the steps for both fully- and partially-parametric modeling. In fully-parametric modeling (FPM) a shape variant is created (from scratch) for which a new mesh is generated, ideally by keeping all mesh settings constant. In partially-parametric modeling (PPM) there are two flavors, either a shape variant is brought about for which a new mesh needs to be produced or the modifications to the shape are directly imposed on the existing mesh, too. The latter approach may lead to possible mesh deterioration when changes become too large while the former may occasionally suffer if (re)meshing is not successful for all variants.

7 Requirements and Comparison

A number of requirements can be formulated which characterize an ideal parametric model:

- Independence of parameters:
  Each parameter influences the shape uniquely and no combination of parameters renders any other parameter superfluous,
- Fairness of resulting variants:
  All potential variants are free of any unwanted shape characteristics,
- Balance:
  Shapes can be produced beyond the current engineering practice while still avoiding unacceptable artifacts, yielding a manageable design space,
- Direct incorporation of (equality) constraints:
  (Geometric) constraints are used as part of the shape definition (reducing DoF),
- Fitness for simulation:
  All variants are free of gaps, folds, overlaps etc. (robustness),
- Ease of understanding:
  The parameterization has some task-related meaning, can be readily understood, is reasonably documented and/or visualized.

Meeting all of these requirements is a challenge. Often a parametric model is adapted or set up anew after some preliminary work. A comparison of advantages and disadvantages of the various techniques is given in Table 1 (refer also to Fig. 1). It summarized the discussion given in the previous sections. It should be noted that ratings such as low, medium and high can only be indicative for standard implementations without considering specific improvements as reported e.g. in [9, 15]. Recent developments are targeted towards further reducing parameter sets and ensuring parameter independence [3]. The aim is to build a reduced-dimensionality representation of the shape modification without losing the essence of the design space. Clearly, this is particularly useful if expensive simulations are utilized and the degrees-of-freedom shall be further lowered.

8 Conclusions

Parametric modeling techniques are a key prerequisite to successful shape optimization with regard to fluid-dynamic performance. Different techniques are available, ranging from partially-parametric to fully-parametric modeling. While partially-parametric modeling is relatively easy to set up and apply, particularly when doing detailed design and fine-tuning, fully-parametric modeling needs higher investment but is applicable from concept design to fine-tuning and yields the highest potential for substantial improvements of product performance.

Table 1 Comparison of common parametric modeling techniques

Approach	Design stages	Preparation time required	Know-high required	Potential for shape optimization
FPM: MetaSurface approach	Concept design to fine-tuning	Medium to high (suitable for a wide range of shapes)	High (task specific)	High (small design spaces)
FPM: Lofting	Concept design to fine-tuning	Medium (easily applicable to blades)	Low to medium	Low to medium
PPM: Free-form deformation	Detailed-design to fine-tuning	Medium (boxes for complex shapes non-trivial to set up)	Medium	Medium
PPM: Shift transformation	Concept design to fine-tuning	Small	Low	Medium
PPM: Morphing	Concept design to fine-tuning	Medium (various traditional baselines needed)	Medium	Medium (for good baselines)
PPM: Added patch	Detailed-design to fine-tuning	Small to medium	Low	Medium
PPM: Radial basis functions	Detailed-design to fine-tuning	Medium	Low	Medium

The right choice which parametric modeling technique(s) to utilize and possibly to combine, depends on the design task at hand. For optimization using CFD an upfront CAD approach is recommended in which less important details are deliberately left out, speeding up the process and increasing the chances of finding shapes with substantially improved performance.

Acknowledgements All examples shown were realized within CAESES®, FRIENDSHIP SYSTEMS' integration platform for robust variable CAD and optimization. Parts of the work and developments leading to the material presented in this paper were undertaken within the research and development project No-Welle, funded by the Federal Ministry of Economics and Technology (BMWi) on the orders of the German Bundestag and PtJ as the conducting agency (FKZ 03SX362 D).

References

1. Abt, C., Harries, S.A.: New approach to integration of CAD and CFD for naval architects. In: 6th International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2007), Cortona, Italy, Apr 2007
2. Abt, C., Harries, S.: The Tools are at hand. In: The Naval Architect, pp. 166–167, Sept 2007

3. Diez, M., Campagna, E.F., Stern, F.: Design-space dimensionality reduction in shape optimization by Karhunen-Loève expansion. Comput. Methods Appl. Mech. Eng. **283**(2015), 1525–1544 (2015)
4. FRIENDSHIP SYSTEMS: CAESES®, Embedded technical documentation. www.caeses.com (2015)
5. Harries, S.: Serious play in ship design. Tradition and Future of Ship Design in Berlin, Colloquium, Technical University Berlin, Germany, Feb 2008
6. Harries, S., Abt, C.: Parametric curve design applying fairness criteria. International Workshop on Creating Fair and Shape-Preserving Curves and Surfaces, Teubner (1998)
7. Harries, S., Abt, C., Hochkirch, K.: Modeling meets Simulation—Process Integration to improve Design, Honorary colloquium for Prof. Hagen, Prof. Schlüter and Prof. Thiel, Duisburg, Germany, July 2004
8. Hoekstra, M., Raven, H.: A practical approach to constrained hydrodynamic optimization of ships. In: International Conference Ship and Shipping Research (NAV 2003), Palermo, 2003
9. Koshakji, A., Quarteroni, A., Rozza, G.: Free form deformation techniques applied to 3D shape optimization problems. MATHICSE Technical Report Nr. 44.2013, Ecole Polytechnique Federale de Lausanne (2013)
10. Kulfan, B.M., Bussoletti, J.E.: Fundamental parametric geometry representations for aircraft component shapes. In: 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference: The Modeling and Simulation Frontier for Multidisciplinary Design Optimization, AIAA Paper 2006–6948 (2006)
11. Lackenby, H. On the Systematic Geometrical Variation of Ship Forms, *RINA-Transactions*, Vol. 92, 1950
12. Peri, D., Rossetti, M., Campana, E.F.: Design optimization of ship hulls via CFD techniques. J. Ship Res. **45**(2), 140–149 (2001)
13. Samareh, J.A.: A survey of shape parameterization techniques. In: CEAS/AIAA/ICASE/NASA Langley International Forum on Aeroelasticity and Structural Dynamics, NASA/CP-1999-209136, pp. 333–343 (1999)
14. Sederberg, T.W., Parry, S.R.: Free-form deformation of solid geometric models. SIGGRAPH Comput. Graph. (ACM) **20**(4), 151–160 (1986)
15. Sieger, D., Menzel, S., Botsch, M.: A comprehensive comparison of shape deformation methods in evolutionary design optimization. In: EngOpt 2012—International Conference on Engineering Optimization (2012)

Simulation-Based Design Optimization by Sequential Multi-criterion Adaptive Sampling and Dynamic Radial Basis Functions

Matteo Diez, Silvia Volpi, Andrea Serani, Frederick Stern and Emilio F. Campana

Abstract The paper presents a global method for simulation-based design optimization (SBDO) which combines a dynamic radial basis function (DRBF) surrogate model with a sequential multi-criterion adaptive sampling (MCAS) technique. Starting from an initial training set, groups of new samples are sequentially selected aiming at both the improvement of the surrogate model global accuracy and the reduction of the objective function. The objective prediction and the associated uncertainty provided by the DRBF model are used by a multi-objective particle swarm optimization algorithm to identify Pareto-optimal solutions. These are used by the MCAS technique, which selects new samples by down-sampling the Pareto front, allowing for a parallel infill of an arbitrary number of points at each iteration. The method is applied to a set of 28 unconstrained global optimization test problems and a six-variable SBDO of the DTMB 5415 hull-form in calm water, based on potential flow simulations. Results show the effectiveness of the method in reducing the computational cost of the SBDO, providing the background for further developments and application to more complex ship hydrodynamic problems.

1 Introduction

Simulation-based design optimization (SBDO) techniques have developed in the last decades in response to the high cost of the build-and-test design paradigm, relying on the increasing accuracy of the simulation tools and availability of computational resources. SBDO requires the automated integration of design modification tools, accurate computer simulations, and optimization algorithms. SBDO typically

M. Diez (✉) · A. Serani · E. F. Campana
CNR-INSEAN, National Research Council-Marine Technology Research Institute,
Via di Vallerano 139, 00128 Rome, Italy
e-mail: matteo.diez@cnr.it

S. Volpi · F. Stern
IIHR–Hydroscience and Engineering, The University of Iowa 100 C. Maxwell
Stanley Hydraulics Laboratory, Iowa City 52242-1585, USA

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_13

requires a large number of computer simulations to identify the global optimal solution to the design problem. The high-fidelity, complexity, and computational expense of the simulation tools is approaching resource saturation, requiring therefore cost-reducing solutions.

The application of surrogate models alleviate the computational cost of SBDO. When performing surrogate-based SBDO, a sampling of the design space by design of experiment (DoE) techniques is used to train a surrogate model of the desired objective function, which is used by the optimization algorithm. Surrogates have been widely used in performing SBDO, including optimization of stochastic black-box functions [14], optimization under uncertainty [15], sampling-based reliability-based design optimization [31], deterministic [5] and stochastic [10] hydrodynamic optimization.

The sampling of the design space needs to be efficient an effective, possibly achieving two competitive goals: an adequate global accuracy of the surrogate model (especially when a global optimum is sought), and a fine investigation of promising design regions [2]. DoEs defined on the basis of a priori methods can hardly achieve these goals. For this reason, *adaptive* sampling techniques have been developed which exploit information that becomes available during the optimization process. The literature proposes a large variety of adaptive sampling criteria. Some examples include: the Kushner's criterion [19], which maximizes the probability of improving the objective; the expected improvement criterion, used in the efficient global optimization (EGO) algorithm [16]; the lower confidence bounding function [7], which minimizes the linear combination of surrogate model prediction and surrogate model uncertainty; locating the threshold-bounded extreme, locating the regional extreme, and minimizing surprises [30].

The objective of the current research is the extension of a dynamic radial basis function (DRBF) surrogate model [29], used in earlier work for uncertainty quantification of ship hydrodynamic problems, to global, derivative free, deterministic design optimization.

The current method implements a sequential *multi-criterion adaptive sampling* (MCAS) technique based on DRBF-predicted objective and associated uncertainty. Starting from an initial training set, groups of new samples are selected from the Pareto front of non-dominated solutions obtained by a multi-objective extension of the deterministic particle swarm optimization (MODPSO) algorithm [4, 18, 22]. An additional single-objective DPSO [26] is performed over the DRBF model to improve the selection of the global minimizer. The procedure is iterated until convergence.

The method is applied to 28 unconstrained global optimization test problems, as well as to the hull-form optimization in calm water and fixed speed of the DTMB 5415, an early concept of the USS Arleigh Burke-class destroyer DDG-51 used as a benchmark for experimental [20, 27] and numerical optimization [12, 13, 25] studies. A potential flow code [1] is used for the simulations. The performance of the DRBF method is assessed by comparison with a direct application of DPSO.

2 Optimization Problem Formulation

Given a design variable vector $\mathbf{x}$ of dimension N and a design objective

$$f(\mathbf{x}) \; : \; \mathbb{R}^N \rightarrow \mathbb{R} \tag{1}$$

the optimization problem is formulated as

$$\min_{\mathbf{x}\in\Omega} f(\mathbf{x}) \tag{2}$$

where

$$\Omega = \left\{\mathbf{x} \in D \subset \mathbb{R}^N \mid c_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, N_c\right\} \tag{3}$$

is the feasible set, D is the design space defined by box constraints, and $c_i(\mathbf{x})$ are inequality constraints. Herein, these are handled by a linearly penalized objective function

$$f_p(\mathbf{x}) = f(\mathbf{x}) + \gamma \sum_{i=1}^{N_c} \max\left[c_i(\mathbf{x}), 0\right] \tag{4}$$

3 Dynamic Radial Basis Function Method for Optimization

3.1 Surrogate Model

Given a set of M training points $\{\mathbf{z}_i\}_{i=1}^{M}$ with associated function evaluations $y_i = g(\mathbf{z}_i)$, a power law RBF provides predictions as

$$h(\mathbf{x}, \varepsilon_j) = \sum_{i=1}^{M} w_i \varphi\left(||\mathbf{x} - \mathbf{z}_i||\right) \quad \text{with} \quad \varphi = ||\mathbf{x} - \mathbf{z}_i||^{\varepsilon_j} \tag{5}$$

where the exponent $\varepsilon_j \in \mathbb{R}$ is a tuning parameter; $\mathbf{w} = \{w_i\}_{i=1}^{M}$ is the solution of the linear system that provides exact prediction at $\mathbf{x} = \mathbf{z}_i$

$$\mathbf{A}\mathbf{w} = \mathbf{y} \quad \text{with} \quad a_{ij} = \varphi\left(||\mathbf{z}_i - \mathbf{z}_j||\right) \quad \text{and} \quad \mathbf{y} = \{y_i\} \tag{6}$$

The DRBF model [29] provides the expected value of a sample of RBF predictions over a stochastic distribution of ε_j. Herein, ε_j is assumed uniformly distributed between $\varepsilon_{\min}$ and $\varepsilon_{\max}$:

$$\hat{g}(\mathbf{x}) = \mathrm{EV}[h(\mathbf{x}, \varepsilon_j)] \quad \text{with} \quad \{\varepsilon_j\}_{j=1}^{N_\varepsilon} \sim \mathrm{unif}[\varepsilon_{\min}, \varepsilon_{\max}] \tag{7}$$

The uncertainty $\hat{U}_g(\mathbf{x})$ associated to the prediction at $\mathbf{x}$ is quantified by the 95%-confidence band of $h(\mathbf{x}, \varepsilon_j)$.

If multiple functions (g_k, $k = 1, \ldots, N_g$) are assessed, multiple surrogate models need to be computed. If these are based on the same training points $\{\mathbf{z}_i\}_{i=1}^{M}$, with corresponding function evaluations $\mathbf{y}_k = \{g_k(\mathbf{z}_i)\}_{k=1}^{N_g}$, a single factorization of the matrix $\mathbf{A}$ may be used. In fact, the system

$$\mathbf{AW} = \mathbf{Y} \tag{8}$$

may be solved at once. In this case, $\mathbf{W} = \left[\mathbf{w}_1 | \ldots | \mathbf{w}_{N_g}\right]$ and $\mathbf{Y} = \left[\mathbf{y}_1 | \ldots | \mathbf{y}_{N_g}\right]$.

When the surrogate model is used for unconstrained optimization, $N_g = 1$, $g(\mathbf{x}) = f(\mathbf{x})$, and the model output is the couple $\{\hat{f}, \hat{U}_f\}$. When the surrogate model is used for constrained optimization, $N_g = N_c + 1$, $g_1(\mathbf{x}) = f(\mathbf{x})$, $g_2(\mathbf{x}) = c_1(\mathbf{x})$, $\ldots$, $g_{N_c+1}(\mathbf{x}) = c_{N_c}(\mathbf{x})$, and the output is the $N_c + 1$ couples $\{\hat{f}, \hat{U}_f\}$, $\{\hat{c}_1, \hat{U}_{c_1}\}$, $\ldots$, $\{\hat{c}_{N_c}, \hat{U}_{c_{N_c}}\}$. The penalized objective function $\hat{f}_p(\mathbf{x})$ is computed using the predictions $\hat{f}$, $\hat{c}_1$, $\ldots$, $\hat{c}_{N_c}$, as per Eq. 4.

3.2 *Multi-criterion Adaptive Sampling*

Starting from an initial training set, the MCAS identifies groups of new samples balancing the surrogate model accuracy and the search for the global minimizer. This is pursued by solving the multi-objective optimization problem

$$\min_{\mathbf{x}\in D} \hat{f}_p(\mathbf{x}) \quad \text{and} \quad \max_{\mathbf{x}\in D} \hat{U}_f(\mathbf{x}) \tag{9}$$

Note that for unconstrained problems $\hat{f}_p(\mathbf{x})$ corresponds to $\hat{f}(\mathbf{x})$, while $\hat{U}_f(\mathbf{x})$ is always the uncertainty of the (non-penalized) objective.

The Pareto front obtained is down-sampled in order to identify m equally spaced points along a curvilinear coordinate ξ (Fig. 1). In view of the fact that: (a) sampling too close to available training points does not add useful information to the analysis, (b) as the distance between training points decreases, the matrix A in Eq. 6 may result ill-conditioned, and (c) the uncertainty at the training points is zero, i.e.

$$\lim_{||\mathbf{x}-\mathbf{z}_i||\to 0} \hat{U}_f(\mathbf{x}) = 0, \tag{10}$$

a constraint is defined such that $\hat{U}_f(\mathbf{x}) \geq u_{\min}$, where $u_{\min} = \beta \hat{U}_{range}$ with $\hat{U}_{range} = \{\max[\hat{U}_f(\mathbf{x})] - \min[\hat{U}_f(\mathbf{x})]\}$.

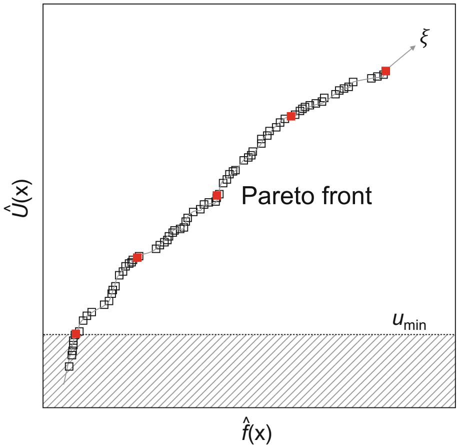


Fig. 1 Pareto solutions of the multi-objective problem with samples

3.3 *Optimization Procedure*

The optimization procedure using DRBF and MCAS is performed as per the following algorithm.

Algorithm DRBF with MCAS

```
Step 1. Define an initial DoE $\{\mathbf{z}_i\}_{i=1}^{M}$ and evaluate objective function and
        constraints $\mathbf{Y} = \left[\mathbf{y}_1 | \ldots | \mathbf{y}_{N_c+1}\right]$.
Step 2. Initialize the current optimum: $y_{\text{opt}} = \min(\mathbf{y}_1)$ and $\mathbf{x}_{\text{opt}} = \text{argmin}(\mathbf{y}_1)$.
Step 3. Build the DRBF model for objective and constraints $\{\hat{f}, \hat{U}_f\}, \ldots, \{\hat{c}_{N_c}, \hat{U}_{c_{N_c}}\}$
        using the training set $T \equiv \{\mathbf{z}_i, \mathbf{Y}_{ik}\}$.
Step 4. Find the minimizer $\mathbf{x}^*$ of $\hat{f}_p(\mathbf{x})$ by DPSO and compute the true objective
        function $y^* = f(\mathbf{x}^*)$.
Step 5. If $y^* < y_{\text{opt}}$ then
            update the current optimum $\mathbf{x}_{\text{opt}} = \mathbf{x}^*$.
Step 6. If $\hat{U}_f(\mathbf{x}^*) \geq u_{\min}$ then
            add $\tilde{\mathbf{z}}_1 = \mathbf{x}^*$ to the new DoE;
            define a new DoE $\{\tilde{\mathbf{z}}_i\}_{i=2}^{m}$ by MCAS;
            evaluate objective and constraints at $\{\tilde{\mathbf{z}}_i\}_{i=2}^{m}$.
        Else
            define a new DoE $\{\tilde{\mathbf{z}}_i\}_{i=1}^{m}$ by MCAS;
            evaluate objective and constraints at $\{\tilde{\mathbf{z}}_i\}_{i=1}^{m}$.
Step 7. Add the new $m$ samples to $T$.
Step 8. Iterate step 3 to 7 until convergence.
```

4 Deterministic Particle Swarm Optimization

Particle Swarm Optimization (PSO) belongs to the class of heuristic algorithms for single-objective evolutionary derivative-free global optimization and was originally introduced by Kennedy and Eberhart [18]. In order to make PSO more efficient for its use within SBDO, a deterministic version of the algorithm (DPSO) was formulated by Campana et al. [4] as follows

$$\begin{cases} \mathbf{v}_i^{k+1} = \chi \left[\mathbf{v}_i^k + c_1(\mathbf{x}_{i,pb} - \mathbf{x}_i^k) + c_2(\mathbf{x}_{gb} - \mathbf{x}_i^k) \right] \\ \mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \mathbf{v}_i^{k+1} \end{cases} \tag{11}$$

Equation 11 represents velocity and position, respectively, of the ith particle at the kth iteration. Particles are attracted by the personal best position $\mathbf{x}_{i,pb}$ ever found by the ith particle and by the global best position $\mathbf{x}_{gb}$ ever found by all particles. The effectiveness of DPSO depends on the constriction factor χ, the cognitive and social learning rate c_1 and c_2, along with the number of individuals N_p and their initial distribution and velocity. Serani et al. [26] investigate the effect of such parameters and propose guidelines for an efficient use of the algorithm in the context of ship hydrodynamic optimization [24].

The extension of DPSO to multi-objective problems can be found, for instance, in Pellegrini et al. [22]. This is based on extending the definition of the personal and global best in the Pareto-optimality sense. Specifically, the personal attractor $\mathbf{x}_{i,pb}$ is the closest point to $\mathbf{x}_i$ of the pesonal Pareto front. The global attractor $\mathbf{x}_{i,gb}$ is different for each particle and defined as the closest point to $\mathbf{x}_i$ of the global Pareto front.

5 Optimization Problems

The DRBF model with the MCAS method is applied to unconstrained global optimization test problems and to the hull-form optimization of the DTMB 5415. The formulation of the problems is presented in the following.

5.1 Unconstrained Global Optimization Test Problems

The study includes the minimization of 28 unconstrained global optimization test problems [3, 29] with a number of independent variables ranging from two to 12. These include multimodal, highly nonlinear, and transcendental functions. Table 1 provides details of the problems, such as their dimension and search domain.

Table 1 Unconstrained global optimization test problems

Function	No. variables	ID	Bounds
Alpine	2, 3, 4, 6, 12	A2, A3, A4, A6, A12	$-10 \le x_i \le 5$
Exponential	2, 3, 4, 6, 12	E2, E3, E4, E6, E12	$-10 \le x_i \le 5$
Griewank	2, 3, 4, 6, 12	G2, G3, G4, G6, G12	$-5 \le x_i \le 5$
Hartmann	3, 6	H3, H6	$0 \le x_i \le 1$
Levy	2, 3, 4, 6, 12	L2, L3, L4, L6, L12	$-10 \le x_i \le 10$
McCormick	2	M2	$-1.5 \le x_1 \le 4$, $-3 \le x_2 \le 4$
Styblinski-Tang	2, 3, 4, 6, 12	S2, S3, S4, S6, S12	$-5 \le x_i \le 5$

5.2 Hull-Form Optimization of the DTMB 5415

The SBDO example is the hull-form optimization of the DTMB 5415 model (Fig. 2). This has been widely investigated by towing tank experiments [20, 27] and SBDO studies [12, 17, 25, 28]. In the present work, a single-objective SBDO is shown, aiming at the reduction of the total resistance R_T in calm water at 18 kn, corresponding to a Froude number (Fr) equal to 0.25. Main particulars and design conditions are summarized in Table 2.

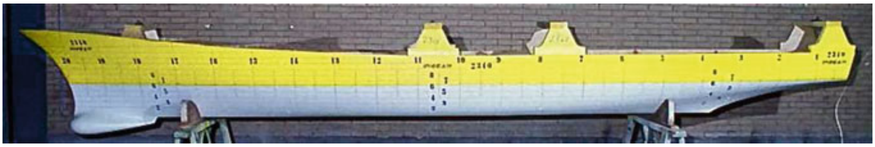


Fig. 2 A 5.720 m length model of the DTMB 5415 (CNR-INSEAN model 2340)

Table 2 DTMB 5415 model main particulars and test conditions (full scale)

Description	Symbol	Unit	Value
Displacement	∇	Tonnes	8,636
Length between perpendiculars	LBP	m	142
Beam	B	m	18.9
Draft	T	m	6.16
Longitudinal center of gravity	LCG	m	71.6
Vertical center of gravity	VCG	m	1.39
Speed	V	kn	18
Water density	ρ	kg/m^3	998.5
Kinematic viscosity	ν	m^2/s	1.09×10^{-6}
Gravity acceleration	g	m/s^2	9.803

An orthogonal representation of the shape modification is used, since more efficient in the context of shape design optimization [3, 9]. Specifically, six orthogonal functions $\Psi_{1,\ldots,6}$ are applied for the modification of the hull shape, controlled by six design variables $\alpha_{1,\ldots,6}$:

$$\begin{cases} \Psi_j(u,v) := \alpha_j \sin\left(\dfrac{p_j \pi u}{A_j} + \phi_j\right) \sin\left(\dfrac{q_j \pi v}{B_j} + \chi_j\right) \mathbf{e}_{k(j)} \\ (u,v) \in [0; A_j] \times [0; B_j] \end{cases} \tag{12}$$

where u and v are curvilinear coordinates; p_j and q_j define the order of the function in u and v direction, respectively; ϕ_j and χ_j are the corresponding spatial phases; A_j and B_j define the modification domain size; $\mathbf{e}_{k(j)}$ is a unit vector. Table 3 summarizes the parameters used here, including upper and lower bounds of α_j. The results will be presented in terms of non-dimensional design variables $x_j \in [-1, 1]$ given by $x_j = 2(\alpha_j - \alpha_{j,min})/(\alpha_{j,max} - \alpha_{j,min}) - 1$. Geometrical constraints include fixed displacement and length between perpendiculars (automatically satisfied by the geometry modification tool), and ±5% maximum variation of beam and draft.

The solver used is the potential flow code WARP [1] based on the double model linearization [8]. The wave resistance is estimated by integrating the pressure over the hull, whereas the friction resistance is estimated by a local approximation based on flat-plate theory [23]. Simulations are performed for the right demi-hull taking advantage of the symmetry about the xz plane. The computational domain for the free surface is defined within 1 *LBP* upstream, 3 *LBP* downstream and 1.5 *LBP* sideways. The associated panel grid used can be found in Serani et al. [25]. The validation of the computations for the original hull is shown in Fig. 3 versus experimental data collected at CNR-INSEAN [21] showing a reasonable agreement especially for low speeds. $C_T = R_T/0.5\rho V^2 S_{w,\text{stat}}$, δ, and τ are shown, where R_T is the total resistance, $S_{w,\text{stat}}$ is the static wetted surface area, δ is the sinkage (positive if the center of gravity sinks), and τ is the trim (positive if the bow sinks).

Table 3 Orthogonal function parameters for shape modification

Description	j	p_j	ϕ_j	q_j	χ_j	$k(j)$	$\alpha_{j,min}$ (m)	$\alpha_{j,max}$ (m)
Hull modification	1	2.0	0	1.0	0	2	−1.0	1.0
	2	3.0	0	1.0	0	2	−1.0	1.0
	3	1.0	0	2.0	0	2	−0.5	0.5
	4	1.0	0	3.0	0	2	−0.5	0.5
Sonar dome modification	5	1.0	0	1.0	0	2	−0.3	0.3
	6	0.5	$\pi/2$	0.5	0	3	−0.5	0.5

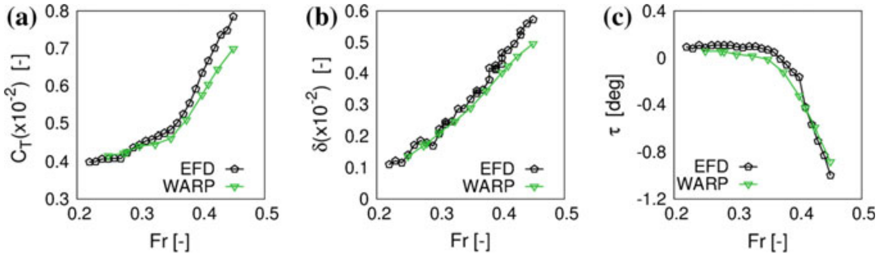


Fig. 3 Total resistance coefficient (**a**), non-dimensional sinkage (**b**), and trim (**c**) in calm water versus Fr, for the model scale DTMB 5415 ($LBP = 5.72$ m)

6 Numerical Results

The test problems are solved using the following setup. The initial DoE is a Hammersley sequence sampling (HSS), $N_\varepsilon = 513$, $\varepsilon \in [0.75, 2.5]$, $m = 5$, $\beta = 0.01$, and the maximum number of function evaluations N_{eval} is 1000. The hull-form optimization is solved using the following setup. The initial DoE is a HSS, $N_\varepsilon = 200$, $\varepsilon \in [0.75, 2.5]$, $m = 8$, $\beta = 0.01$, and $N_{eval} = 1000$. DPSO parameters are given in Table 4.

6.1 *Unconstrained Global Optimization Test Problems*

For the assessment of the test problems, the normalized difference between the current optimum and the true minimum Δ_f is used as a metric [3]. DRBF and DPSO are iterated until $\Delta_f \leq 0.1\%$ or until the number of function evaluations reaches N_{eval}.

Figure 4 shows the convergence of DRBF and DPSO algorithms for the test problems E2, H6, and G12 as an example. The figure displays the value of the metric versus the number of functions evaluations M. DRBF is found significantly more effective than DPSO for E2 and G12, and slightly more efficient for H6. The average performance of DRBF and DPSO is summarized in Fig. 5, taking into account all the test problems. The average number of function evaluations needed to achieve $\Delta_f \leq 0.1\%$ is shown versus the number of design variables N. On average, DRBF

Table 4 DPSO parameters

Objective function	N_p	N_{eval}	Initialization	$[\chi \quad c_1 \quad c_2]$
f_p (no surrogate)	$4N$	1000	HSS $\mathbf{v} \neq 0$ [5]	[0.721 1.655 1.655] [6]
$\hat{f}_p$ (surrogate, single-objective)	$4N$	$256N$	HSS $\mathbf{v} \neq 0$ [5]	[0.721 1.655 1.655] [6]
$\hat{f}_p$, $\hat{U}_f$ (MCAS, multi-objective)	$32N$	$1024N$	HSS $\mathbf{v} = 0$	[0.9801 0.3333 0.6767] [11]

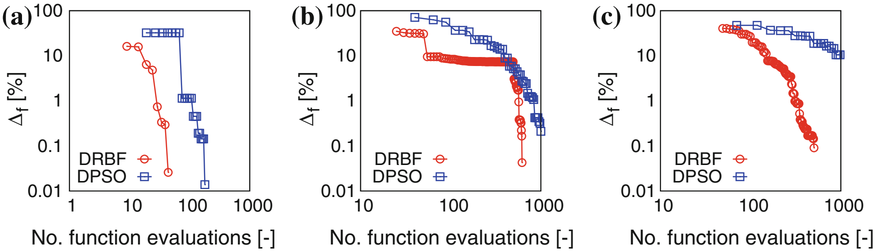


Fig. 4 Convergence of DRBF and DPSO for test problems E2 (**a**), H6 (**b**), G12 (**c**)

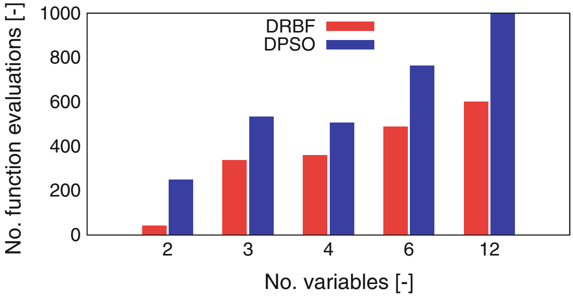


Fig. 5 Average number of function evaluations needed to achieve $\Delta_f \leq 0.1\%$

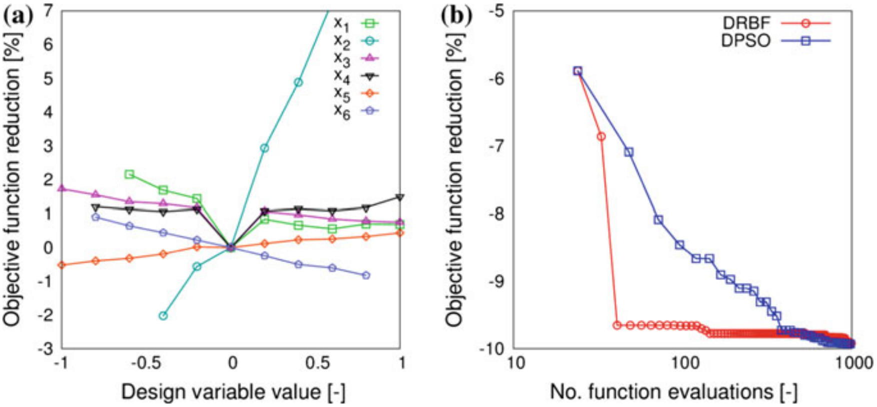
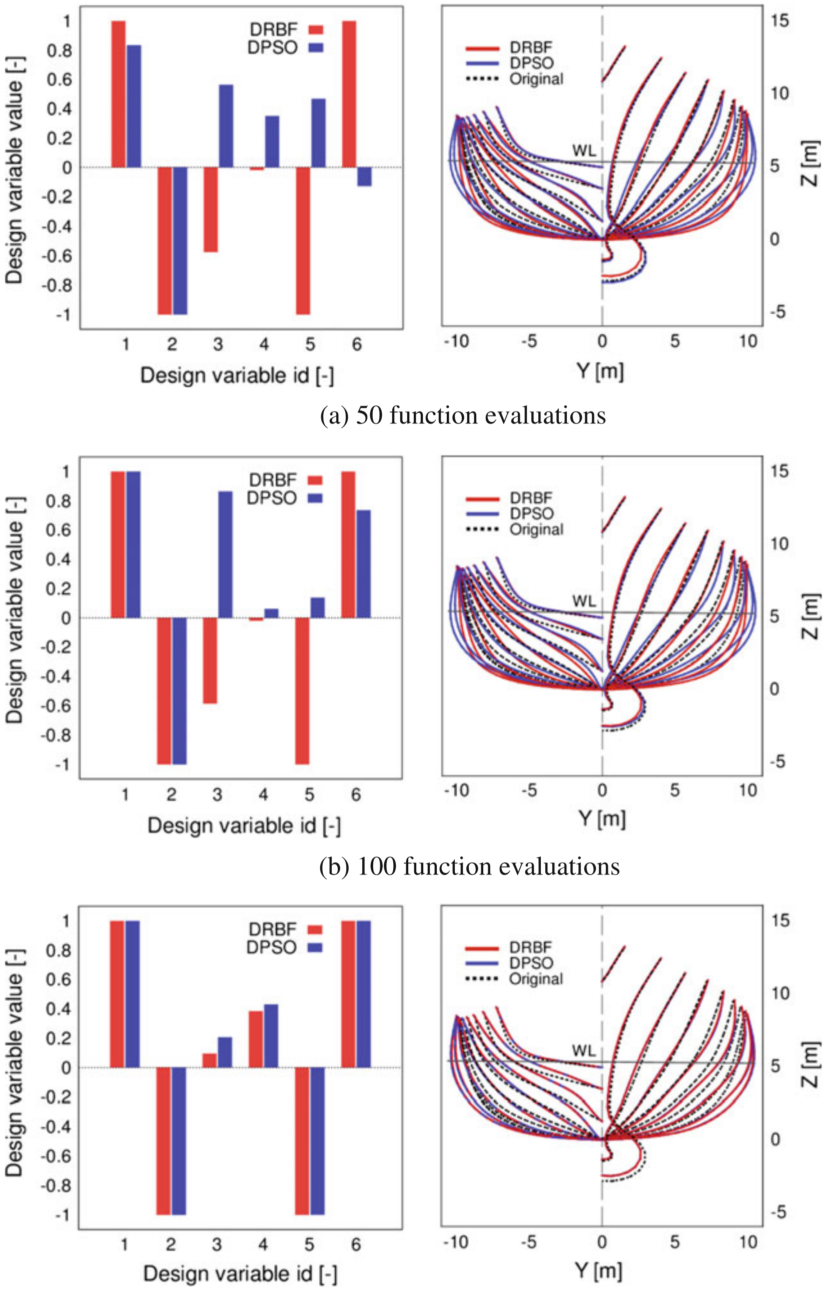


Fig. 6 Sensitivity analysis of the design variables (**a**), and convergence of DRBF and DPSO for the hull-form optimization of the DTMB 5415 (**b**)

requires fewer function evaluations to achieve the optimal solution. Moreover, DRBF outperforms DPSO for $N = 12$ since the latter does not achieve $\Delta_f \leq 0.1\%$ for any of the problems within the prescribed budget N_{eval}.



(a) 50 function evaluations

(b) 100 function evaluations

(c) 1000 function evaluations

Fig. 7 Optimal design variables (left), and sections of the DTMB 5415 original and optimized hulls, comparing DRBF and DPSO solutions (right)

Table 5 Design variables and objective function values for the DTMB 5415 hull-form

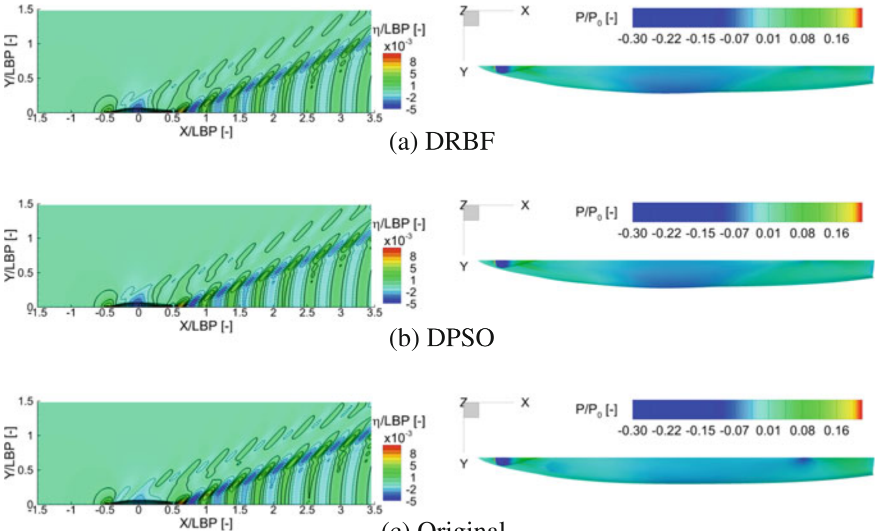
No. function evaluations		Design variables [–]						$R_T \times 10^5$ [N]	
		x_1	x_2	x_3	x_4	x_5	x_6	Value	Δ%
50	DRBF	1.00	−1.00	−0.57	−0.02	−1.00	1.00	3.07	−9.65
	DPSO	0.83	−1.00	0.56	0.35	0.47	−0.13	3.16	−7.09
100	DRBF	1.00	−1.00	−0.59	−0.02	−1.00	1.00	3.07	−9.66
	DPSO	1.00	−1.00	0.86	0.06	0.14	0.74	3.11	−8.46
1000	DRBF	1.00	−1.00	0.10	0.38	−1.00	1.00	3.06	−9.92
	DPSO	1.00	−1.00	0.21	0.43	−1.00	1.00	3.06	−9.93

6.2 *Hull-Form Optimization of the DTMB 5415*

The penalized objective of Eq. 4 is computed using a penalty coefficient $\gamma = 100$.

A preliminary sensitivity analysis for each design variable is presented in Fig. 6a. Unfeasible designs are not reported. Changes in f reveal a potential reduction of the total resistance (at Fr = 0.25) close to 2%.

The optimization process by DRBF is shown in Fig. 6b, including the comparison with DPSO. The figure displays the convergence of f versus M. Note that all the solutions correspond to feasible designs. Both methods reach a total resistance reduction



(a) DRBF

(b) DPSO

(c) Original

Fig. 8 Wave elevation pattern (left) and pressure field distribution (right) at Fr = 0.25 of the optimized DRBF (**a**) and DPSO (**b**) hulls compared to the original (**c**), with 1000 function evaluations

Table 6 Summary of optimization results for DTMB 5415 hull-form

Parameter	Unit	Original	Optimized $\Delta\%_{\text{orig}}$	
			DRBF	DPSO
C_w	–	1.00E−03	−26.36	−26.77
C_f	–	1.61E−03	−0.04	−0.05
C_T	–	2.62E−03	−10.15	−10.30
δ/LBP	–	1.37E−03	5.45	5.27
τ	rad	9.90E−04	91.54	89.99
$S_{w,\text{stat}}/LBP^2$	–	1.48E−01	0.25	0.42
$S_{w,\text{dyn}}/LBP^2$	–	1.51E−01	0.36	0.52

close to 10%. DRBF provides a quite sudden convergence achieving 9.6% reduction with 41 function evaluations, whereas DPSO requires nearly 400 evaluations to reach the same improvement.

Figure 7a, b, and c show the solutions for 50, 100, and 1000 function evaluations, respectively, in terms of variable values and hull sections of the corresponding designs. Table 5 gives the design variables and the associated objective function values. The solutions provided by DRBF and DPSO differ in the objective reduction by 2.66%, 1.20%, and 0.01%, for 50, 100, and 1000 evaluations, respectively.

Figure 8 shows the non-dimensional wave elevation pattern and the associated non-dimensional pressure distributions on the hull comparing DRBF and DPSO final designs to the original hull. The transverse wave is reduced and the pressure shows a better recovery towards the stern.

Finally, Table 6 summarizes the main parameters associated with the optimal DRBF and DPSO designs. The resistance coefficients are defined as $C_x = R_x/0.5\,\rho V^2 S_{w,\text{stat}}$, with R_w, R_f, R_T being wave, frictional, and total resistance, respectively; $S_{w,\text{stat}}$ and $S_{w,\text{dyn}}$ are static and dynamic wetted surface areas.

7 Conclusions and Future Work

The current study investigates the performance of a novel method for simulation-based design optimization (SBDO), which combines a dynamic radial basis function (DRBF) surrogate model with a sequential multi-criterion adaptive sampling (MCAS) technique. The MCAS selects groups of new samples sequentially, starting from an initial deterministic DoE and using the function prediction and its associated uncertainty as provided by the surrogate model. Function value and uncertainty of the surrogate are the two objectives of a multi-objective deterministic particle swarm optimization (MODPSO) algorithm, which is used to obtain Pareto-optimal solutions. The MCAS performs the parallel infill of an arbitrary number of new training points by down-sampling of the Pareto front. Therefore, this sampling method pursues simultaneously the global accuracy of the surrogate and the refinement of optimal regions, also exploiting the availability of parallel computing architectures.

Numerical results for a set of 28 unconstrained global optimization test problems show that DRBF outperforms a direct application of DPSO, requiring on average approximately 80% fewer function evaluations for two-dimensional problems and 35% for higher dimensions.

The application of DRBF to the six-variable hull-form optimization of the DTMB 5415 shows the potential of the method in performing constrained SBDO problems. The hull is optimized using a potential flow solver and the total resistance is reduced by nearly 10%. For a large number of simulations, DRBF and DPSO converge approximately to the same solution. DRBF is found more efficient than the direct application of DPSO, showing a quite sudden convergence. Specifically, 9% resistance reduction is achieved by DRBF requiring nearly ten times fewer simulations than DPSO.

Future developments include the assessment of optimal initial DoEs, in terms of number of samples and distribution, along with the optimal number of samples selected by the MCAS technique at each iteration. The promising result of DRBF lays the groundwork for further investigations, including SBDO with larger design spaces and the use of high-fidelity CFD methods, such as RANS solvers.

Acknowledgements The work was performed within NATO RTO Task Group AVT-204 "Assess the Ability to Optimize Hull Forms of Sea Vehicles for Best Performance in a Sea Environment." The authors are grateful to Dr Woei-Min Lin, Dr Ki-Han Kim, and Dr Salahuddin Ahmed of the US Office of Naval Research, for their support through NICOP grant N62909-15-1-2016 and grant N00014-14-1-0195. The authors are also grateful to the Italian Flagship Project RITMARE, coordinated by the Italian National Research Council and founded by the Italian Ministry of Education.

References

1. Bassanini, P., Bulgarelli, U., Campana, E., Lalli, F.: The wave resistance problem in a boundary integral formulation. Surv. Math. Ind. **4**, 151–194 (1994)
2. Booker, A.J., Dennis Jr., J., Frank, P.D., Serafini, D.B., Torczon, V., Trosset, M.W.: A rigorous framework for optimization of expensive functions by surrogates. Struct. Optim. **17**(1), 1–13 (1999). https://doi.org/10.1007/BF01197708
3. Campana, E.F., Diez, M., Iemma, U., Liuzzi, G., Lucidi, S., Rinaldi, F., Serani, A.: Derivative-free global ship design optimization using global/local hybridization of the DIRECT algorithm. Optim. Eng. **17**(1), 127–156 (2015)
4. Campana, E.F., Liuzzi, G., Lucidi, S., Peri, D., Piccialli, A., Pinto, A.: New global optimization methods for ship design problems. Optim. Eng. **10**(4), 533–555 (2009). https://doi.org/10.1007/s11081-009-9085-3
5. Chen, X., Diez, M., Kandasamy, M., Zhang, Z., Campana, E.F., Stern, F.: High-fidelity global optimization of shape design by dimensionality reduction, metamodels and deterministic particle swarm. Eng. Optim. **47**(8), 473–494 (2015)
6. Clerc, M.: Stagnation analysis in particle swarm optimization or what happens when nothing happens (2006). http://clerc.maurice.free.fr/pso
7. Cox, D.D., John, S.: SDO: a statistical method for global optimization. In: Multidisciplinary Design Optimization: State of the Art, pp. 315–329 (1997)

8. Dawson, C.W.: A practical computer method for solving ship-wave problems. In: Proceedings of the 2nd International Conference on Numerical Ship Hydrodynamics, pp. 30–38. Berkeley (1977)
9. Diez, M., Campana, E.F., Stern, F.: Design-space dimensionality reduction in shape optimization by Karhunen-Loève expansion. Comput. Methods Appl. Mech. Eng. **283**, 1525–1544 (2015)
10. Diez, M., Chen, X., Campana, E.F., Stern, F.: Reliability-based robust design optimization for ships in real ocean environment. In: Proceedings of 12th International Conference on Fast Sea Transportation, FAST2013, Amsterdam, The Netherlands (2013)
11. Diez, M., Peri, D.: Robust optimization for ship conceptual design. Ocean Eng. **37**(11), 966–977 (2010)
12. Diez, M., Serani, A., Campana, E.F., Goren, O., Sarioz, K., Danisman, D.B., Grigoropoulos, G., Aloniati, E., Visonneau, M., Queutey, P., Stern, F.: Multi-objective hydrodynamic optimization of the DTMB 5415 for resistance and seakeeping. In: Proceedings of the 13th International Conference on Fast Sea Transportation, FAST 2015. Washington, D.C., USA (2015)
13. Grigoropoulos, G., Campana, E., Diez, M., Serani, A., Goren, O., Sarioz, K., Danisman, D., Visonneau, M., Queutey, P., Abdel-Maksoud, M., et al.: Mission-based hull form and propeller optimization of a transom stern destroyer for best performance in the sea environment. In: Michel Visonneau, P.Q., Touzé, D.L. (eds.) VII International Conference on Computer Methods in Marine Engineering, pp. 83–94 (2017)
14. Huang, D., Allen, T., Notz, W., Zeng, N.: Global optimization of stochastic black-box systems via sequential Kriging meta-models. J. Global Optim. **34**(3), 441–466. https://doi.org/10.1007/s10898-005-2454-3, http://dx.doi.org/10.1007/s10898-005-2454-3
15. Jin, R., Du, X., Chen, W.: The use of metamodeling techniques for optimization under uncertainty. Struct. Multidiscip. Optim. **25**(2), 99–116 (2003)
16. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. J. Global Optim. **13**(4), 455–492 (1998)
17. Kandasamy, M., Wu, P.C., Zalek, S., Karr, D., Bartlett, S., Nguyen, L., Stern, F.: CFD based hydrodynamic optimization and structural analysis of the hybrid ship hull. SNAME (2014)
18. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE Conference on Neural Networks, IV, Piscataway, NJ, pp. 1942–1948 (1995)
19. Kushner, H.J.: A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. J. Fluids Eng. **86**(1), 97–106 (1964)
20. Longo, J., Stern, F.: Uncertainty assessment for towing tank tests with example for surface combatant DTMB model 5415. J. Ship Res. **49**(1), 55–68 (2005)
21. Olivieri, A., Pistani, F., Avanzini, A., Stern, F., Penna, R.: Towing tank experiments of resistance, sinkage and trim, boundary layer, wake, and free surface flow around a naval combatant INSEAN 2340 model. Technical report, DTIC Document (2001)
22. Pellegrini, R., Serani, A., Leotardi, C., Iemma, U., Campana, E.F., Diez, M.: Formulation and parameter selection of multi-objective deterministic particle swarm for simulation-based optimization. Appl. Soft Comput. **58**, 714–731 (2017)
23. Schlichting, H., Gersten, K.: Boundary-Layer Theory. Springer, Berlin (2000)
24. Serani, A., Diez, M.: Are random coefficients needed in particle swarm optimization for simulation-based ship design? In: Proceedings of the 7th International Conference on Computational Methods in Marine Engineering (MARINE 2017), pp. 48–59. Nantes, France (2017)
25. Serani, A., Fasano, G., Liuzzi, G., Lucidi, S., Iemma, U., Campana, E.F., Stern, F., Diez, M.: Ship hydrodynamic optimization by local hybridization of deterministic derivative-free global algorithms. Appl. Ocean Res. **59**, 115–128 (2016)
26. Serani, A., Leotardi, C., Iemma, U., Campana, E.F., Fasano, G., Diez, M.: Parameter selection in synchronous and asynchronous deterministic particle swarm optimization for ship hydrodynamics problems. Appl. Soft Comput. **49**, 313–334 (2016)
27. Stern, F., Longo, J., Penna, R., Olivieri, A., Ratcliffe, T., Coleman, H.: International collaboration on benchmark CFD validation data for surface combatant DTMB model 5415. In: Proceedings of Twenty-Third Symposium on Naval Hydrodynamics (2001)

28. Tahara, Y., Peri, D., Campana, E.F., Stern, F.: Computational fluid dynamics-based multiobjective optimization of a surface combatant using a global optimization method. J. Mar. Sci. Technol. **13**(2), 95–116 (2008)
29. Volpi, S., Diez, M., Gaul, N.J., Song, H., Iemma, U., Choi, K.K., Campana, E.F., Stern, F.: Development and validation of a dynamic metamodel based on stochastic radial basis functions and uncertainty quantification. Struct. Multidiscip. Optim. **51**(2), 347–368 (2015). https://doi.org/10.1007/s00158-014-1128-5
30. Watson, A.G., Barnes, R.J.: Infill sampling criteria to locate extremes. Math. Geol. **27**(5), 589–608 (1995)
31. Zhao, L., Choi, K.K., Lee, I., Gorsich, D.: Conservative surrogate model using weighted Kriging variance for sampling-based RBDO. J. Mech. Des. **135**(9), 091,003 (2013)

Application of Holistic Ship Optimization in Bulkcarrier Design and Operation

Lampros Nikolopoulos and Evangelos Boulougouris

Abstract The recent years have seen an evolution of traditional approaches in ship design. Raising fuel costs, tough and volatile market conditions, the constant societal pressure for a «green» environmental footprint combined with ever demanding international safety regulations pose a new challenge for today's Naval Architect. As a result of this current status of shipping commercial ship design is shifting towards new approaches where holistic approaches are deemed necessary. Apart from considering all the interrelationships between the subsystems that consist the vessel, lifecycle and supply chain considerations are the key in successful and «operator-oriented» designs. The paper presents a methodology within the parametric design software CAESES® for the optimization of the basic design of a new vessel and the operation of an existing one with regards to the maximization of the efficiency, safety and competitiveness of the final design. A case study with the design optimization was undertaken based on the simulation of the anticipated operation of a vessel engaged in the supply chain of Iron Ore. The target was the minimization of costs, fuel consumptions as well as of the Energy Efficiency Operating Index (EEOI) under conditions of uncertainty.

1 Introduction

For centuries the backbone of global trade and prosperity has been international shipping, with the vast majority of transportation of raw material as well as manufactured goods being transported by ships. While the 20th century saw the expansion of shipping in parallel with the industrial revolution, the first decade of the 21st posed a series of challenges for commercial shipping. The economic recession combined

L. Nikolopoulos (✉) · E. Boulougouris
School of Naval Architecture, Ocean and Marine Engineering, University
of Strathclyde, 100 Montrose Street, Glasgow G4 0LZ, UK
e-mail: lampros.nikolopoulos@strath.ac.uk

E. Boulougouris
e-mail: evangelos.boulougouris@strath.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_14

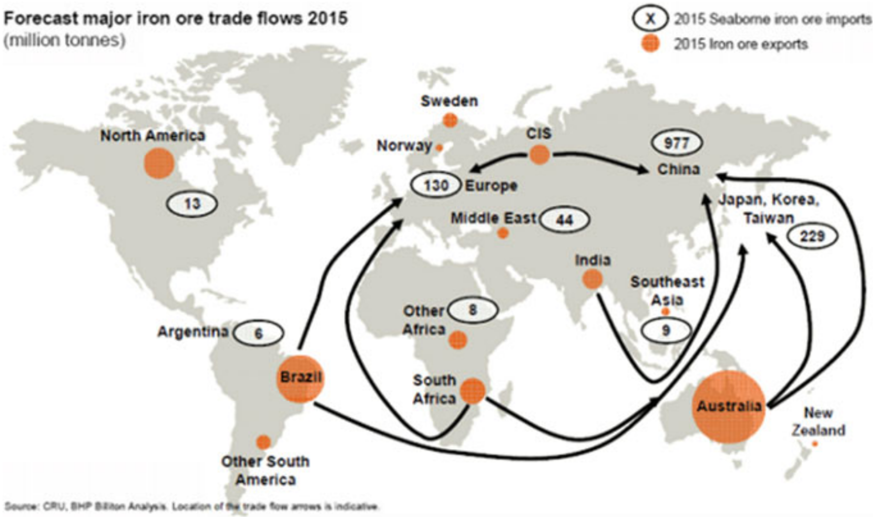


Fig. 1 Major iron ore trades

with a fall in freight rates (due to tonnage overcapacity as well as a global economic slowdown in terms of growth per capita) has threatened the financial sustainability of numerous companies. At the meantime, the Kyoto [1] protocol and the societal pressure for greener shipping, gave birth to a number of international environmental regulations that set the scheme for future ship designs. These are required to have a small carbon footprint and also incorporate ballast treatment systems to mitigate the risk of reducing biodiversity (especially in sensitive ecosystems such as reefs) due to the involuntary carriage of evasive species in the ballast water tanks.

Different cargoes have different main routes. Focusing on the seaborne trade of major bulk commodities such as iron ore or coal, the trade routes are very specific and shown in Fig. 1.

The rapid expansion of the Chinese economy created a constant demand for both iron and coal. The major iron ore exporters are located in South America (primarily Brazil) and Australia with million tons of exports per annum. The coal production is concentrated in Indonesia, Australia and Russia with 383, 301 and 314 million tons accordingly. The coal consumers are the Atlantic market consisted by Western European countries (mainly Germany and the UK) and the Pacific market, which consists of developing and OECD Asian importers, notably Japan, Korea and Chinese Taipei. The Pacific market currently accounts for about 57% of world seaborne steam coal trade. For the past half century global bulk shipping has focused on providing tonnage to serve the above trade with vessels of considerable size due to absence of significant size restrictions. The latter being the outcome of the ever expanding port terminals and the absence of physical restrictions (e.g. Panama Canal) on these routes. The present paper focuses on vessels intended for this trade which belong in the Capesize/Very Large Ore Carrier (VLOC) segment of the shipping market.

The design of bulkcarriers was focused during the last 7 years on the increase of efficiency by two means: increase of cargo carrying capacity and decrease of energy demands. In most cases the optimization is evolved around a single design point in terms of both speed and loading condition (draft and thus displacement). This paper provides a holistic methodology [2] intended for the optimization of the basic design of large bulkcarriers for their entire lifecycle, operational profile and supply chain. The speed and trading profile is simulated for the entire economic life of the vessel and the optimization focuses on the minimization of operating costs, maximization of income, minimization of internal rate of return (IRR) summarized by the Required Freight Rate (RFR) from one hand and from the other the minimization of the energy footprint of the vessel expressed by the Energy Efficiency Design Index (EEDI) and the simulated Energy Efficiency Operating Index (EEOI). In order to make sure that the produced designs will be also safe, the optimization targets on the minimization of the risk of structural failure without unnecessary increases the lightship weight.

2 Overview of the Holistic Methodology

Holism (from ὂλος holos, a Greek word meaning all, whole, entire, total) is the idea that natural systems (physical, biological, chemical, social, economic, mental, linguistic, etc.) and their properties, should be viewed as wholes, not as collections of parts. This often includes the view that systems somehow function as wholes and that their functioning cannot be fully understood solely in terms of their component parts. Within this context the authors have developed such methodologies in the Ship Design Laboratory of NTUA with the use of CAESES® [3] parametric design software or other similar tools [4, 5]. This approach has been applied in a variety of cases, e.g. to tanker design optimization [6] as well as to containership design [7].

The methodology is holistic in the sense that all the critical aspects of the design are addressed under a common framework that takes into account the lifecycle performance of the ship in terms of safety, efficiency and economic performance, the internal system interactions as well as the trade-offs and sensitivities. The workflow of the methodology has the same tasks as the traditional design spiral with the difference that the approach is not sequential but concurrent.

2.1 Design and Simulation Environment

The environment in which the methodology is programmed and is responsible for the generation of the fully parametric hullform is CAESES® which stands for “CAE system empowering simulation” [3]. It is a CAD-CFD integration platform which was developed for simulation-driven design of functional surfaces like ship hulls,

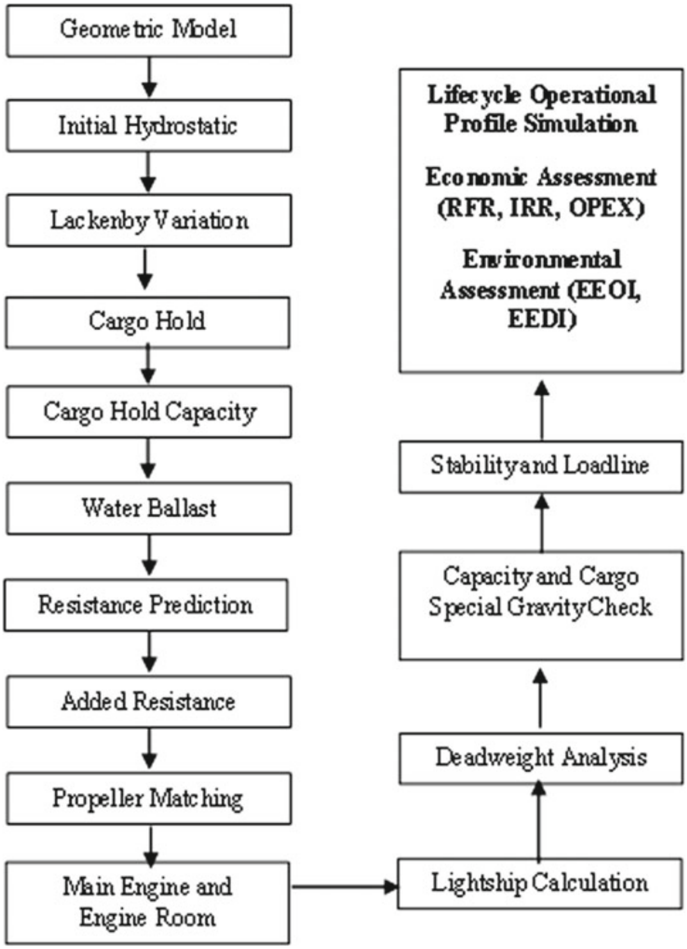


Fig. 2 Workflow of the proposed methodology

propellers and appendages. It may also be used for other applications like designing turbine blades and pump casings. It provides a wide range of functionalities like parametric modelling, integration of simulation codes, algorithms for systematic variation and formal optimization. These capabilities make it an ideal tool for the holistic ship design optimisation problem, where a parametric hullform should be generated and its performances should be assessed by different software tools. The holistic methodology proposed herein is depicted in Fig. 2 and will be analysed in the next paragraphs.

2.2 Geometric Core

The core of any holistic design optimisation method developed in a CAD/CAE system is the geometric model. For the ship hullform this poses unique challenges due to the fairness and shaping requirements for both the forward and the aft area e.g. the shape of the bulbous bow and the size of the transom respectively. In that respect, the more flexible is the modelling environment, the better and higher the resolution of the design space exploration would be. CAESES® offers such flexibility and this is why it was selected for the definition of the original hull. The surface of the hull was modelled as a group of parametric sub-surfaces.

2.3 Initial Hydrostatic Properties

The calculation of the hydrostatic properties is important for the verification of integrity of the design by its displacement, the block coefficient and the centre of buoyancy of the design. It is performed by an internal computation of CAESES®. For its execution a dense set of offsets (sections) is required as well as a plane and a mirror plane, defined by the user.

2.4 Lackenby Variation

In order to be able to generate the lines with the desired geometrical properties, the Lackenby [8] variation is applied. This variation is a transformation that is able to change the distribution of the enclosed volume longitudinally. Instead of applying quadratic polynomials as shift functions, fairness optimized B-Splines are used. This allows a better selection of the region influenced as well as smoother transition. The required input for the transformation is its extent and the target values for the block coefficient (Cb) and the longitudinal centre of buoyancy (LCB). In this case the extent was from the propeller's position to the fore peak. An example is shown in Fig. 3.

2.5 Cargo Hold Modelling

The cargo hold arrangement was generated on that resulting surface using a feature of CAESES® and the capacity of the various holds was calculated. The cargo hold surfaces and their respective parametric entity were realized within the CAESES®. The parameters/variables controlling this area were the positions of the bulkheads, the position of the Engine Room bulkhead, the frame spacing as well as some local

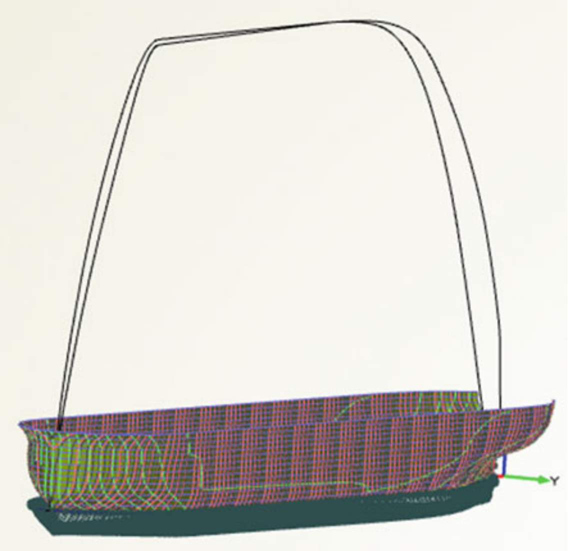


Fig. 3 Hullform after the application of Lackenby's variation

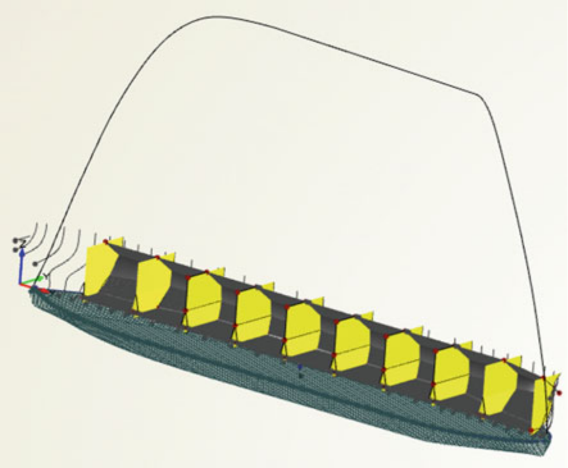


Fig. 4 Parametric cargo hold surfaces

variables such as the hopper width and angle, the topside tank dimensions (width and height), the lower stool height and length and double bottom height.

The capacity of each tank was calculated by creating offsets for each one of the tank surfaces and joining them together. The calculation of the tanks' hydrostatics was performed then and the total capacity was checked. A calibration factor which derived from the parent hull was applied to account for the volume losses due to the structural frames inside the cargo holds. A similar factor was used for the estimation of the Bale and Grain capacities. The result of the parametric tank modelling is shown in Fig. 4.

2.6 Resistance Prediction

2.6.1 Calm Water Resistance

The resistance prediction of this model uses a hybrid method and two different approaches, depending on the optimization stage.

Initially, during the design of experiment (DoE) and the global optimization phase, where a large number of variants is created there is a need for a fast procedure. For this particular reason Holtrop's [9] approximate powering prediction method is used. It derives from the statistical analysis of model tests and is well-known for its very good accuracy-to-computational cost ratio. Especially for bulkcarriers it is more accurate as the wave making and the viscous pressure resistance are very small fractions of the total resistance. It is the frictional resistance (directly related to the wetted surface) that dominates the total resistance due to their small Froude number. The entire Holtrop's method was programmed within CAESES®. Thus, the actual data from the geometric model (e.g. entrance angle, prismatic coefficients etc.) are used, making the process more precise for the specific design.

An innovative feature of the methodology developed is that the parameters from Holtrop's statistical method were systematically calibrated in order for the programmed methodology to match the speed-resistance and speed-power curves of the parent vessel as derived from its model tests. The calibration was performed by a systematic optimization approach. The optimization variables were the coefficients used in Holtrop's methodology with a small margin of variance. Then the methodology would be applied for each speed/point of the model tests and the difference in powering would derive. The minimization of this difference is the optimization target of this particular sub-problem. As 9 different speeds (from 12.5 to 16.5 knots) were assessed the applied algorithm for the optimization was the NSGA II [10], while 900 variants were produced. The result was an average difference of 1.5% with the Holtrop results being more conservative (over estimation) than the model tests.

At a later stage, in the optimization post processing, where local hullform parameters are considered, the CFD code package STAR CCM+ is used in order to validate the trends in terms of propulsion efficiency for the Pareto front designs. However, the results of Holtrop are generally conservative and on the safe side compared with CFD analysis while there are no discrepancies regarding the ranking of the designs in terms of hull efficiency. Under these assumptions, the use of Holtrop's method at the preliminary design stage can be considered a prudent choice since the results cannot be considered to be distorted and any errors are systematic and they do not bias the results. Given the systematic calibration for the same «family» of hullforms it argued that this is a sound strategy for both accuracy and computational efficiency.

2.6.2 Main Engine and Engine Room Dimensioning

With the propeller dimensioned, the RPM and required power of the main engine are determined. A margin for adverse weather conditions and fouling is considered on the basis of 15% as per industry standard. A further 5% is also considered for derating the main engine and ensuring smaller SFOC.

For the final requirements the main engine is matched with the existing G-Type, ultra-long stroke, engines available from MAN [11]. An internal iterative procedure ensures that the engine will have sufficient light running margin and that the layout point on the diagram is close to the L2L4 line corresponding to larger torque/MEP margins and smaller SFOC values.

From the above, the final SFOC curve from 50 to 100% is produced and corrected for the actual engine layout.

The Diesel Generator output is calculated from an electrical balance while the boiler output is based on the exhaust gas amount of the main engine in order to be also sufficient for the steam production for the onboard heating of the fuel tanks.

2.6.3 Lightship Weight Prediction

The lightship calculation follows the traditional categorization in three weight groups, the machinery weight, the outfitting weight and the steel weight.

Machinery Weight

The machinery weight calculation is based on the average of two methods: the Watson-Gilfillan formula and the calculation based on the Main Engines weight respectively.

The machinery weight estimation is based on a empirical formula due to Watson-Gilfillan [12]:

$$Wm = Cmd * Pb^{0.89} \tag{1}$$

The average is used to balance out any extreme differences, and the coefficients of the Watson-Gilfillan formula are calibrated for low speed, two stroke engines based on statistic data available for a fleet of bulkers.

Outfitting Weight

The outfitting weight is also based on the average of two independent calculations. The Schneekluth method [12] is the first one and the use of empirical coefficients for sub-groups of that particular weight group is the other.

Steel Weight

During the initial design stages, and the selection of optimal main dimensions, it is necessary to identify the effect of the change of the principal dimensions of a reference ship on the structural steel weight. Thus, at first, an accurate calculation of the steel weight of the reference ship is conducted. Following this, the "Schneekluth Lightship Weight Method" was applied [12]. Given that the steel weight for the parent vessel was available as derived from summing the individual steel block weights (from the shipbuilding process) a TSearch algorithm [3] was employed in order to vary the values of the statistical coefficients and constants of subject methodology with the objective of the minimization of the difference between the actual and calculated values for the steel weight. The results have an accuracy of 0.3% which is more than acceptable within the scope of basic/preliminary design.

2.7 Deadweight Analysis

The deadweight of the vessel is comprised by subgroups such as the consumables, the crew weight and the deadweight constant. The deadweight analysis predicts the payload of the vessel based on the calculation of the consumables.

As mentioned before, the consumables for the machinery is calculated, namely the Heavy Fuel Oil for the main engines, and diesel generators, the Lubricating Oils of the engines and generators.

Furthermore, based on the number of the crew members (30), the fresh water onboard is calculated as well as the supplies and the stores of the vessel.

2.8 Stability and Loadline Check

The initial intact stability is assessed by means of the metacentric height of the vessel (GM). The centre of gravity of the cargo is determined from the capacity calculation within the framework while the centre of gravity for the lightship and consumables is determined from non-dimensioned coefficients (functions of the deck height) that derive from the information found in the trim and stability booklet of the parent vessel. All the above are calculated according to the requirements of the IMO Intact Stability Code [13].

2.9 Operational Profile Simulation

This module is an integrated code within the methodology that simulates the actual operating conditions of the vessel for its entire lifecycle. Two trade routes are considered, the Brazil to China and the Australia to China roundtrips. Each voyage is split into legs depending on distinctive sea areas.

Input Data

For each one of the legs (given distance in nautical miles) the average speed and added resistance curves are input as well as the loading of the generators and the manoeuvring time. If the leg includes discharging, loading or bunkering port the corresponding time in hours is also used. Based on this profile, the voyage associated costs together with the fuel costs are calculated on a much more accurate and realistic basis. The predictions of this module have been verified by actual data from real ships.

Added Resistance

In order to be consistent with the need for the simulation driven design it is necessary to include a consideration for the added resistance in waves. Thus, a module has been herein developed that utilizes Kwon's method for the calculation of added resistance in waves [14, 15].

Kwon's added resistance modelling is an approximate method for the prediction of loss of speed due to added resistance in rough weather condition (irregular waves and wind). The advantage of this method is the prediction of the involuntary loss of speed due to the effect of weather loading on an advancing displacement type of ship with a limited number of input data. The module is described by Eqs. (2) and (3).

$$\frac{\Delta V}{V_1} * 100\% = C_\beta * C_U * C_{Form} \tag{2}$$

$$V_2 = V_1 - \left(\frac{\Delta V}{V_1} * 100\%\right) * \frac{1}{100\%} * V_1 = V_1 - \left(C_\beta * C_U * C_{Form}\right)\frac{1}{100\%} * V_1 \tag{3}$$

where:

V_1 Design (nominal) operating ship speed in calm water conditions (no wind, no waves), given in m/s

V_2 Ship speed in the selected weather (wind and irregular waves) conditions, given in m/s

$\Delta V = V_2 - V_1$, Speed difference, given in m/s.

C_β Direction reduction coefficient, dependent on the weather direction angle (with respect to the ship's bow) and the Beaufort number BN (Bft), as shown in Table 1

C_U Speed reduction coefficient, dependent on the ship's block coefficient b. The loading condition and the Froude number n, as shown in Table 2

Table 1 Direction reduction coefficient CB due to weather direction

Weather direction	Direction angle (with respect to the ship's bow) (deg)	Direction reduction coefficient Cß
Head sea (irregular waves) and wind	0	2Cß = 2.0
Bow sea (irregular waves) and wind	30–60	2Cß = 1.7 – 0.03 * (BN-4)^2
Beam sea (irregular waves) and wind	60–150	2Cß = 0.9 – 0.06 * (BN-6)^2
Following sea (irregular waves) and wind	150–180	2Cß = 0.4 – 0.03 * (BN-8)^2

Table 2 Speed reduction coefficient CU due to Block coefficient Cb

Block coefficient Cb	Ship loading conditions	Speed reduction coefficient C_U
0.8	Loaded or normal	$2.6–13.1 \cdot \mathrm{Fn}-15.1 \cdot \mathrm{Fn}^2$
0.85	Loaded or normal	$3.1–18.7 \cdot \mathrm{Fn}+28 \cdot \mathrm{Fn}^2$
0.8	Ballast	$3.0–16.3 \cdot \mathrm{Fn}\text{-}21.6 \cdot \mathrm{Fn}^2$
0.85	Ballast	$3.4–20.9 \cdot \mathrm{Fn}+31.8 \cdot \mathrm{Fn}^2$

Table 3 Ship form coefficient CForm due to ship categories and loading condition

Type of ship	Ship form coefficient C_{Form}
Full hull in laden condition	$0.5 \cdot \mathrm{BN}+(\mathrm{BN}^{6.5})/(2.7 \cdot \nabla^{2/3})$
Full hull in ballast condition	$0.7 \cdot \mathrm{BN}+(\mathrm{BN}^{6.5})/(2.7 \cdot \nabla^{2/3})$

C_{Form} Ship form coefficient, as shown in Table 3.

The above formulas for speed loss need to be combined for all the sea states and weather angles of each of the stages of the determined voyage legs (see Sect. 2.10) in order to include all the in service considerations. The derived reduced speed from the Kwon calculation is in combined with Holtrop's resistance for the powering prediction. This results to four different Added Resistance–Speed curves, depending on the weather angle (0–30, 30–60, 60–150 and 150–180 as in Fig. 5). Then, in the operational simulation module (Sect. 2.10) for each stage and voyage leg, the computation of these four curves is performed for Beaufort numbers of the following groups: (0,2], (2,4], (4,6], (6,8].

For each stage of each leg, the probability of both the weather encountering heading as well as the Beaufort number range is set as input. At the end, a probabilistic additional Propulsion Power given the known stage/leg average speed is derived. Using this requirement, the engine load is estimated.

The developed methodology calculates the Operational Expenditure (OPEX), the Capital Expenditure (CAPEX), the Required Freight Rate (RFR), the Internal Rate of Return (IRR) as well as the IMO Energy Efficiency Operational Index (EEOI). All

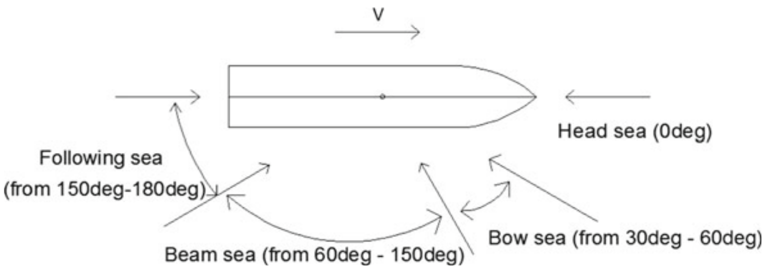


Fig. 5 Vessel heading directions

the calculations are made under two uncertainties: Fuel Price and Market Condition (expressed by the Baltic Dry Index and USD per ton of cargo paid as charter that is translated into TCE afterwards). This is another key point of this methodology, as it allows the optimization of the vessel's design under uncertainty as the produced designs correspond to a more realistic scenario and the dominant variants of the optimization have a more robust behaviour over a variety of exogenous governing market factors. The derived probabilistic values of RFR and the deterministic value of the EEOI are the functions/targets used in the optimization sequence later.

2.10 Energy Efficiency Design Index Calculation

The Energy Efficiency Design Index (EEDI) is calculated according to the formula proposed in the IMO resolution MEPC.212(63) [16], using the values of 70% deadweight and 75% of the MCR of the engines and the corresponding reference speed:

$$EEDI = \frac{\left(\prod_{j=1}^{M} fj\right)\left(\sum_{i=1}^{nME} PME(i) * CFME(i) * SFCME(i)\right) + (PAE * CFAE * SFCAE)}{fi * Capacity * Vref * fw} + \frac{\left\{\left(\prod_{j=1}^{M} fj * \sum_{i=1}^{nPTI} PPTI(i) - \sum_{i=1}^{neff} feff(i) * PAEeff(i)\right) * CFAE * SFCAE\right\} - \left(\sum_{i=1}^{neff} feff(i) * Peff(i) * CFME * SFCME\right)}{fi * Capacity * Vref * fw} \tag{4}$$

The minimization of this index is one of the primary objectives of the conducted optimization. The engine power is directly related to the resistance of the hullform, while the deadweight is also related to both the hullform in terms of displacement and to ship's lightship weight.

3 Design Concept

3.1 Large Bulkcarrier Market

The focus of the present study lies within the large bulkcarrier segment dominated in numbers by Capesize ships as well as Very Large Ore Carriers (VLOCs). During the last decade a new class of vessels has emerged, known as Newcastlemax as they are the largest vessels that can enter and load in the Coal Terminal of Newcastle in Australia.

3.2 Baseline Vessel—208k Newcastlemax

It is imperative in a ship design optimization case study that a baseline is set in the form of a parent vessel used as a primary source of reference as well as calibration for the methodology and all the formulas/computations applied in the latter. For this particular reason it is necessary to have as complete data as possible for the parent vessel in order to achieve a better degree of accuracy as well as being able to make proper comparison during the analysis of the dominant variants of the optimization front.

The vessel chosen for this study belongs to the new category segment of Newcastlemax Bulkers and is a newly delivered vessel. The baseline parametric geometry has been adapted to fit the hull lines available. Its model test results were used to calibrate Holtrop's prediction for resistance and powering. The principal particulars of the vessel can be found in Table 4.

3.3 Proposed Design Concept Characteristics

A low Froude number (slow speed) and full hullform is herein proposed as the base hull for the global optimization. The absence of a bulbous bow is evident as it is a recent trend in bulkcarrier design. It results from the understanding that such a geometry assists in the reduction of the vessel frictional resistance (primary resistance component) while the wave making resistance is not increased. The effect of the bulbous bow on the above as well as the added resistance were investigated in depth in a separate study. Furthermore, the decision to limit the selection of the Main Engine to only electronically controlled types was taken and no Energy Saving Devices (wake equalizing duct, pre-swirl fin, bulbous rudder etc.) are considered since there

Table 4 Baseline vessel principal particulars

Baseline vessel principal particulars	
Length over all	299.98
Length between perpendiculars	294
Beam	50
Scantling draft	18.5
Deck height	25
Cb	0.8521
Main engine specified MCR (kW)	17494 @ 78.7 RPM/MAN B&W 6G70ME-C9.2
Deadweight (tons)	Abt 208,000
Lightship weight (tons)	26,120
Cargo hold capacity (m^3)	224,712.1

is no such device installed on the parent vessel. The improvements achieved by such devices can be considered at a later stage.

3.3.1 Simulation Driven Design, Choice of Hullform Parameters

The assessment of the design is derived from the simulation of the operational, economic and trading profile. In other words, instead of using only one design point (in terms of draft and speed) multiple points are used derived from actual operating data of a the baseline vessel.

3.3.2 Newcastlemax Design Concept

The maximum allowable dimensions (Length Over All and Breadth) in order to load in the port Newcastle in Australia set the constraints for this optimization case study.

4 Optimization Studies

4.1 Optimization Target/Goals

The generic targets or objectives in this optimization problem were:

4.1.1 Competitiveness

The market and economic competitiveness of a design is the core of any optimization as a vessel will always be an asset (of high capital value). This can be expressed by the following indices:

1. *Required Freight Rate*

The required freight rate is the hypothetical freight which will ensure a break even for the hypothetical shipowner between the operating costs, capital costs and its income based on the annual voyages as well as collective cargo capacity and is such expressed in USD per ton of cargo.

2. *Operating Expenditure (OPEX)*

The operating expenditure expressed on a daily cost includes the cost for crewing, insurance, spares, stores, lubricants, administration etc. It can indicate apart from the operator's ability to work in a cost effective structure, how the vessel's design characteristics can affect. The lubricant cost is based on actual feed rates used for subject engines as per the relevant service letter SL2014-537 of MAN [17].

3. *Capital Expenditure (CAPEX)*

The CAPEX is a clear indication of the cost of capital for investing and acquisition of each individual design variant. The acquisition cost is calculated from a function derived from actual market values and the lightship weight for vessels built in Asian shipyards, and more specifically in China.

4.1.2 Efficiency

The merit of efficiency is herein expressed by the IMO EEOI index. Although on the design basis in practice the IMO Energy Efficiency Design Index is used as a KPI and measure of the merit of efficiency in new design concepts as well as for any newbuild vessel, in this study the calculated Energy Efficiency Operating Index is used instead. The reason for this change is the use of the Operational Profile simulation module which contains from a wide statistical database of a bulker operator the daily average speed per each stage of each voyage leg (see Sect. 2.10) thus given the cargo capacity calculation (see Sect. 2.4) the EEOI can be accurately derived, which can depict more accurately the efficiency of the design given the fact that it takes into account all operating speeds (instead of one design speeds) and all operating drafts (instead of the design draft) thus expressing the actual transport efficiency of each variant by a simple ratio of tons of CO_2 emitted (direct function of the tons of fuel consumed) to the tons of cargo multiplied by the actual distance covered (in nautical miles). In addition to the above, each operational practice such as slow steaming is taken into account, also considering side implications (for example the use of two diesel generators in the normal sea going condition instead of one in order to cover the blower's electrical load).

4.2 Design Variables

The design variables used are shown in Table 5. They can be categorized in three groups; principal dimensions, hullform characteristics (Cb, LCB, Parallel Middlebody) and cargo hold arrangement parameters. The more detailed design variables of the hullform arrangement for the detailed shape of the bulbous bow (if any), flair and stem shape as well as stern shape are going to be assessed in a separate optimization study with the use of integrated CFD codes.

4.3 Optimization Procedure

The optimization procedure applied for this study is depicted in Fig. 6.

Table 5 List and range of design variables of the optimization problem

Design variable	Lower boundary	Upper boundary
Length between perpendiculars	290	299
Length overall	298	300
Beam	48	50
Draft	18	19
Deck height	24	27
Hopper length	8	11
Hopper breadth (m)	3	6
Topside height (m)	8	14
Topside breadth (m)	9	13
Inner bottom height (m)	2.4	3
Block coefficient Cb	0.84	0.87
LCB (% Lbp)	0.49	0.53
Beginning of parallel midbody (Aft % Lbp)	0.35	0.45
End of parallel midbody (Fore % Lbp)	0.65	0.8
Stem overhang (% Lbp)	0	0.02

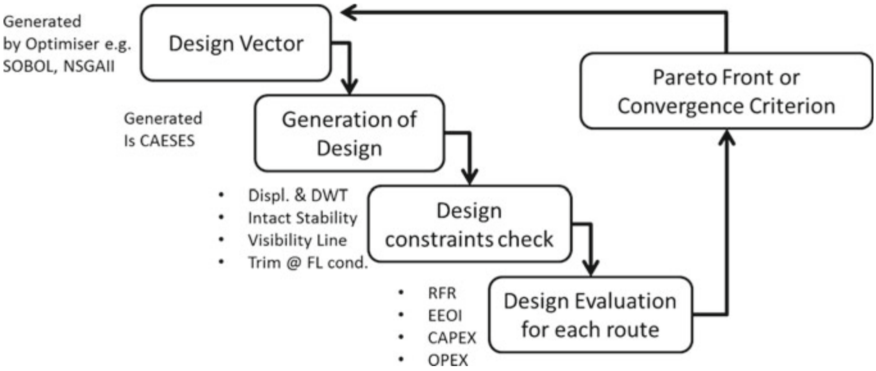


Fig. 6 The optimization loop applied

In each iteration the design variables receive their input values from the "design engine" i.e. CAESES®. The design engine can either be a random number generator, a design of experiments procedure or an optimization algorithm depending on the optimization stage. The generated values trigger the creation of a new design from the parametric model. The design's performance, in the form of the calculated values of the Design Objectives, is logged and assessed accordingly and the Design Constraints imposed are checked for compliance. The Design constraints chosen for this study were the calculated values for Deadweight, Cargo Specific Gravity and the Stability Criteria of the 2008 Intact Stability Code. The size restrictions (in terms of vessel's dimensions) were not used in constraints given the fact they were taken into account in the applied range of the Design Variables.

The optimization procedure described in this paper can be described as a multi stage one. At first, it is necessary to explore and fully understand both the design space (potential for improvement with given constraints) as well as the sensitivity of the methodology by a Design of Experiments (SOBOL) procedure. The sensitivity analysis is a very important, preparatory step in which it is ensured that no major, unreasonable manipulations occur. Furthermore, it is important to see that the results are realistic both on a quantitative and qualitative basis, with the latter in need of particular attention since the design ranking and selection is the essence of optimization (the absolute value of a design is not important than the relationship with all the other produced designs).

During the next stage a formal optimization is performed using a genetic algorithm technique (NSGA II algorithm). The formal optimization runs involve the determination of the number of generations and the definition of population of each generation to be explored. The generated designs are ranked according to a number of scenarios regarding the preference of the decision maker. One favoured design is picked to be the baseline design of the next optimization run, where the same procedure is followed. When it is evident that there little more potential for improvement the best designs are picked using the same ranking principles with utility functions, and are exported for further analysis.

Both the SOBOL and NSGA II algorithms as well as a plethora of other variant generation and optimization algorithms are fully integrated and available within the CAESES®.

4.4 Design of Experiment

The Design of Experiment has the primary purpose of calibration, test and sensitivity check of the methodology from one hand as well as the investigation for the optimization margin. In this case study, it was evident that there is a strong scale effect that dominates this particular optimization problem. This effect is very common in ship design were the largest vessels usually dominate the smaller since the increase of cargo capacity does not trigger an equivalent increase in the powering requirements or the vessel's weight.

In addition to the scaling effect it was observed as in the formal optimization algorithm that there was a strong linear correlation between the Required Freight Rate (RFR) and the EEOI, which was expected since both functions use cargo capacity.

The feasibility index was in a very high level (above 90%). In total 250 designs were created.

4.5 Global Optimization Studies

In this stage the formal, global design optimization with the NSGA II algorithm was utilized. The latter is a genetic, evolutionary algorithm that is based on the principles of biological evolution [18]. As in the biological evolution each design variant is an individual member of a population of a generation. Each individual of the population is assessed in terms of the Optimization Objectives, as well as its relation to the desired merits. For the application in ship design optimization it is usual to apply a large population for each generation with an adequate number of generations. The large population combined with a high mutation probability ensures that the design space is properly covered, while the number of generations ensures that there is a push towards the Pareto frontier for each case of objective combination. For this particular application a combination of 10 generations with 100 variants population each was selected.

The results of this run can be seen in Figs. 7, 8 and 9. In Fig. 7 the relation of the RFR to the EEOI is depicted and is quite evident that their relationship as already explained is strongly linear. The reason is the direct correlation to the cargo capacity for both indices. It is interesting to note that the baseline vessel is in the middle and towards the lower part of the range meaning that although it belongs to the better performers it is away from dominant variants.

When it comes to the relationship between the CAPEX and RFR (see Fig. 9) we can see that there is a contradicting requirement since the acquisition cost is calculated with a linear function of the lightship weight, while the larger vessels

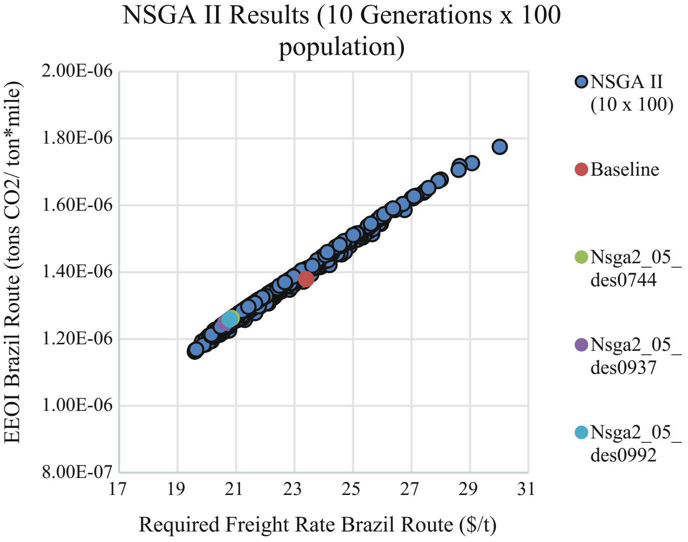


Fig. 7 NSGA II run: RFR versus EEOI

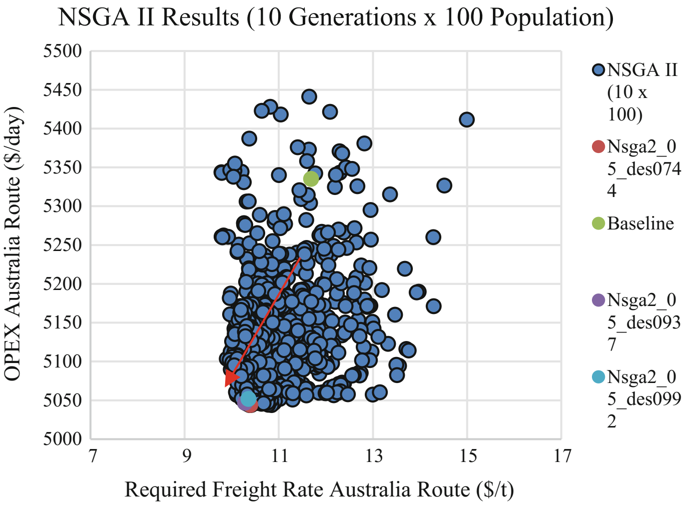


Fig. 8 NSGA II results: OPEX versus RFR

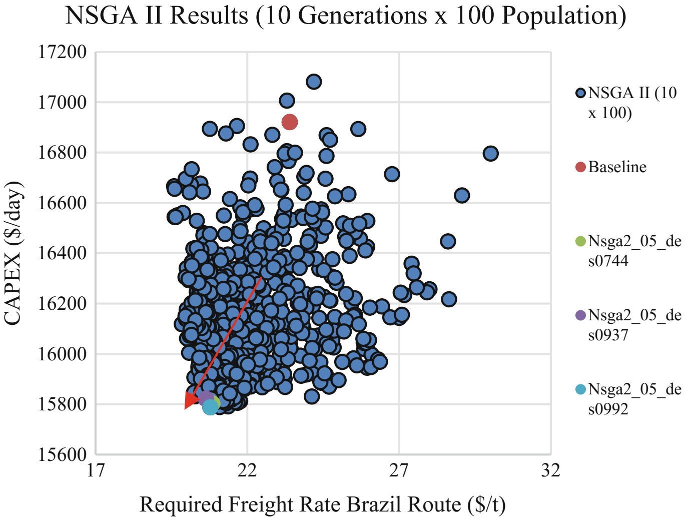


Fig. 9 NSGA II results: RFR versus CAPEX

boast a greater profitability and thus better RFR. A small area like a Pareto front is created, however again there is a localized peak that dominates the majority of the generated designs. The same relationship is also observed between the OPEX and RFR values of the generated design (see Fig. 8).

4.6 Dominant Variant Ranking

One of the most critical steps during optimization of any system is the selection and the sorting of the dominant variants. For this particular reason it is necessary to follow a rational, rather than an intuitive, approach in order to consider in an unbiased way all trade-offs that exist. One such method is utility functions technique.

The optimum solution in our case would dispose the minimum EEOI, RFR, OPEX and CAPEX values. Instead of using fixed weights for the set criteria in the evaluation of the variants, we rather assume a utility function as following

$$U = w_{EEOI} \cdot u_{(EEOI)} + w_{RFR} \cdot u_{(RFR)} + w_{CAPEX} \cdot u_{(CAPEX)} + w_{OPEX} \cdot u_{(OPEX)} \quad (5)$$

The maximization of this utility function is the objective now, and the dominant variants of those 10 most favourable with respect to the 4 defined utility scenarios (Table 6) resulting in the identification and sorting of 40 designs with best performance according to each utility scenario.

From the above ranking (Figs. 10, 11, 12 and 13) it is very interesting to observe that there is a certain repetition in the top three dominant variants from the ranking procedure. Furthermore, for scenario U3 where there is an equal weight for all objectives, the three top dominant variants are the ones from scenario's U1 and U2. All the above illustrate that the peak on the observed Pareto front is strong and apart from that, the dominant variants that can be selected (e.g. 744, 937, 992) perform better in a robust way under different assumptions and weights from the decision maker point of view. The characteristics of these three variants can be found in Table 7.

5 Discussion of the Results—Future Perspectives

From Table 8 it is apparent that a 10–11% average improvement in the required Freight Rate has been achieved, while the OPEX and CAPEX values have been

Table 6 Weights used for the utility functions

Maximum objective weight	U1	U2	U3	U4
RFR_Brazil	0.2	0.1	0.125	0.1
RFR_NMAX	0.2	0.1	0.125	0.1
EEOI_Brazil	0.1	0.1	0.125	0.1
EEOI_NMAX	0.1	0.1	0.125	0.1
OPEX_Brazil	0.1	0.1	0.125	0.2
OPEX_NMAX	0.1	0.1	0.125	0.2
CAPEX_Brazil	0.1	0.2	0.125	0.1
CAPEX_NMAX	0.1	0.2	0.125	0.1

Fig. 10 Ranking of dominant variants with U1 scenario

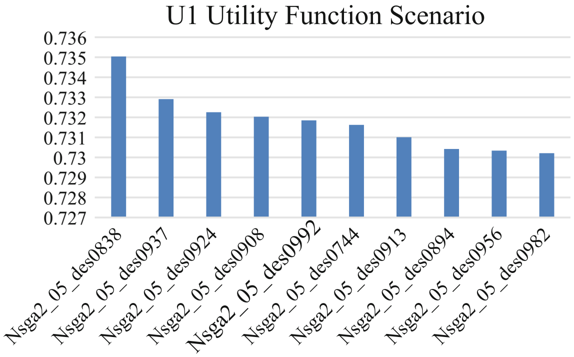


Fig. 11 Ranking of dominant variants with U2 scenario

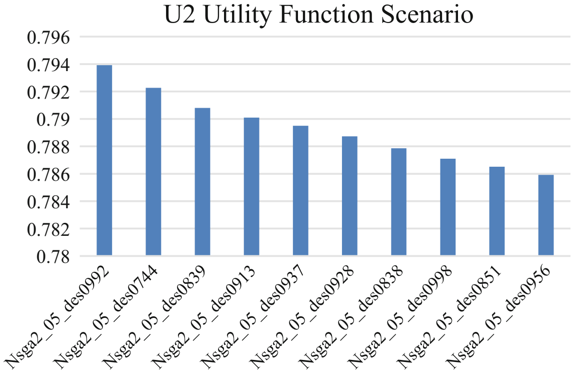
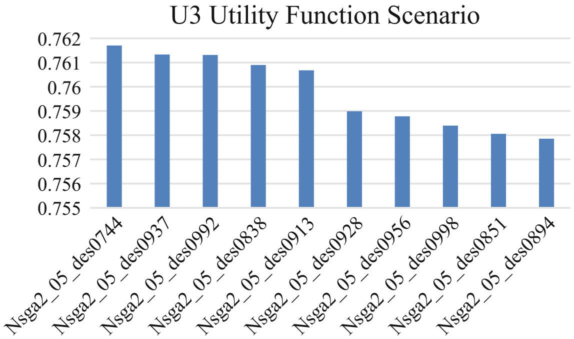


Fig. 12 Ranking of dominant variants with U3 scenario



reduced in a lesser extent by approx. 6.5%. This can be justified by the reduction of generally vessel size primarily in terms of beam and length (beam given the fact that these vessels are not stability limited) and thus the reduction of the initial capital cost, while in the meantime the cargo capacity has increased, boosting in this way the Required Freight Rate. It is also interesting to observe that although beam has

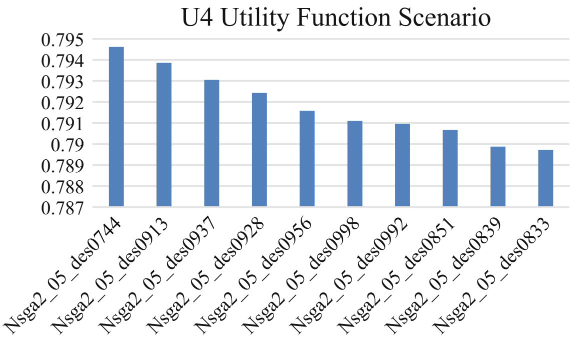


Fig. 13 Ranking of dominant variants with U4 scenario

Table 7 Comparison between the baseline and the three variants geometric properties

Particulars	Baseline	ID744	ID937	ID992
Lbp (m)	294	290.24926	290.26683	290.26464
Beam (m)	50	48.01819	48.07337	48.09241
Deck height (m)	25	26.98824	26.87828	26.98750
Cb	0.8538	0.86535	0.86533	0.86301
LCB	0.51986	0.52203	0.51169	0.51145
LOA (m)	299.98	299.15743	299.04591	298.07306
Draft (m)	18.5	18.00232	18.00220	18.03555
Topside breadth (m)	12	9.11792	11.36893	12.87433
Topside height (m)	9	9.09700	8.30011	8.20636
Hopper height (m)	10	8.56892	8.53816	9.56466
Hopper breadth (m)	4	3.30607	3.22715	3.08890
Double bottom height (m)	2.5	2.82176	2.82140	2.51971
Bow overhang (% Lbp)	0.01	0.00098	0.00120	0.00107
Beggining parallel midbody (% Lbp)	0.42	0.43373	0.40859	0.36219
End parallel midbody (% Lbp)	0.72	0.73976	0.74282	0.76179

reduced the draft has been increased in order to facilitate and balance the decrease in deadweight.

From the above discussion we can conclude that the novel methodology herein proposed for the simulation driven design with lifecycle, supply chain and the actual

Table 8 Design objectives of the baseline versus the dominant variants

Particulars	Baseline	ID 744	Diff%	ID 937	Diff%	ID 992	Diff%
RFR_Brazil	23.40	20.86	−10.86	20.64	−11.80	20.78	−11.17
RFR_Australia	11.69	10.40	−11.07	10.29	−11.99	10.36	−11.38
EEOI_Brazil	0.00	1.26E−06	−8.46	1.25E−06	−9.46	1.26E−06	−8.74
EEOI_Australia	0.00	1.16E−06	−8.49	1.15E−06	−9.49	1.16E−06	−8.78
OPEX_Brazil	5198.09	4911.06	−5.52	4913.97	−5.47	4918.75	−5.37
OPEX_Australia	5335.02	5043.68	−5.46	5046.64	−5.41	5051.42	−5.32
CAPEX	16920.61	15802.94	−6.61	15821.74	−6.49	15788.05	−6.69

operating in service parameters can successfully trigger a reduction in the RFR and EEOI via systematic variation and advanced optimization techniques. However, this is a preliminary work restricted only into illustrating the applicability and potential of this method. The following work is planned for the next steps:

1. Integration of the STAWAVE 2 methodology for added resistance prediction.
2. Refinement of the statistical data for in service conditions.
3. Investigation of larger bulker concepts for the iron ore supply chain by waving the restrictions of the Port of Newcastle and thus scaling up to the VLOC segment while also utilizing dual loadline characteristics.
4. Optimization of the in service operating profile of the vessel. For the baseline vessel, given the results of a pending trim optimization study (with use of STAR CCM+) the in service speeds and trims (trims only for the ballast leg) are going to be re-assessed in a rational way with systematic variation while taking into account exogenous factors such as Port Congestion as well as trade route supply and demand functions.
5. Local Hullform optimization of Bow and Stern Area. Three different bow types (ledge bow, bulbous and semi bulbous) are considered and further optimized for the baseline vessel.

Finally, given the developed library of modules for several calculations in «feature format» in the CAESES® the operational, supply chain and lifecycle simulation is going to be expanded for the case of a containership where the optimization problem is far more challenging due to the slender hullform and the inherent stability limitations.

Acknowledgements The authors would like to express their deep gratitude to their mentor and teacher, Professor Apostolos Papanikolaou, Director of the Ship Design Laboratory at the National Technical University of Athens, who has been the inspiration for this paper. Furthermore, the authors would like to acknowledge the help of Star Bulkcarriers Corporation in providing valuable data from their operating fleet and reference drawings.

References

1. UN Doc FCCC/CP/1997/7/Add.1: Kyoto Protocol to the United Nations Framework Convention on Climate Change, Dec. 10, 1997; 37 ILM 22 (1998)
2. Papanikolaou, A.: Holistic ship design optimization. Comput. Aided Des. (2009). https://doi.org/10.1016/j.cad.2009.07.002
3. FRIENDSHIP SYSTEMS, CAESES® (2015). www.caeses.com
4. Boulougouris, E.K., Papanikolaou, A., Zaraphonitis, G.: Optimisation of arrangements of Ro-Ro passenger ships with genetic algorithms. Ship Technol. Res. **51**(3), 99–105 (2004)
5. Papanikolaou, A., Zaraphonitis, G., Boulougouris, E., Langbecker, U., Matho, S., Sames, P.: Multi-objective optimization of oil tanker design. J. Mar. Sci. Technol. **15**(4), 359–373 (2010)
6. Nikolopoulos, L.: A holistic methodology for the optimization of tanker design and operation and it's applications. Diploma Thesis NTUA, July 2012
7. Koutroukis, G.: Parametric design and multiobjective optimization-study of an ellipsoidal containership. Diploma Thesis NTUA, January 2012
8. Lackenby, H.: On the systematic geometrical variation of ship forms. Trans INA **92**, 289–316 (1950)
9. Holtrop, J., Mennen, G.G.J.: An approximate power prediction method. International Shipbuilding Progress (1982)
10. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
11. MAN B&W: Marine Engine Program (2015)
12. Papanikolaou, A.: Ship Design: Methodologies of Preliminary Design. Springer. ISBN 978-9401787505
13. IMO, Intact Stability Code (2008)
14. Kwon, Y.J.: Speed loss due to added resistance in wind and waves. In: The Naval Architect, vol. 3, pp. 14–16 (2008)
15. Lu, R., Turan, O., Boulougouris, E.: Voyage optimisation: prediction of ship specific fuel consumption for energy efficient shipping. In: Low Carbon Shipping Conference, London (2013)
16. IMO Resolution MEPC.212(63): Guidelines on the method of calculation of the attained energy efficiency design index (EEDI) for new ships, March 2012
17. MAN Diesel and Turbo, SL2014-587 «Guidelines of Cylinder Lubrication», Copenhagen (2014)
18. Darwin, C.: On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life, 1st edn. John Murray, London (1859)

Part V
Game Strategies Combined with Evolutionary Computation

Designing Networks in Cooperation with ACO

E. D'Amato, E. Daniele and L. Mallozzi

Abstract In this paper we present a cooperative game for a network design. The game model adopts for the cooperating players the profit maximizing requirement. Since the players may use different paths, there is the possibility to cooperate and design the optimal network satisfying the requests of all the players and minimizing the cost. The solution of the game is determined by the core concept, well known in cooperative game literature. By means of several examples, both analytical and numerical solutions are proposed. Concerning the computational procedure, in this work an algorithmic approach based on ant colony model is employed. Finally, an application to the airline network design is discussed, providing a numerical example for intercontinental air traffic routes.

1 Introduction

A particularly interesting class of problems is the study of optimization problems on networks, closely related to cooperative and non-cooperative game theory, in which networks and related graph theoretic concepts play a prominent role. The subject has been intensively studied in literature for the computational complexity and the

E. D'Amato (✉)
Dipartimento di Ingegneria, Università della Campania "L. Vanvitelli",
Via Roma 29, 80039 Aversa, Italy
e-mail: egidio.damato@unina.it

E. Daniele
Fraunhofer Institut für Windenergie und Energiesystemtechnik - IWES,
Ammerländer Heerstraße 136, 26129 Oldenburg, Germany
e-mail: elia.daniele@iwes.fraunhofer.de

L. Mallozzi
Dipartimento di Matematica ed Applicazioni, Università degli Studi di Napoli "Federico II",
Via Claudio 21, 80125 Napoli, Italy
e-mail: mallozzi@unina.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_15

design of efficient algorithms, as well as from a theoretical and economics point of view [2, 6, 10, 19].

In the economics literature, several analytical contributions have considered the effect of network topology on competition within the airline industry. In [12] the competion is modelled as a noncooperative game where airlines select network design and prices for transportation between any two nodes. In [13] the network is analyzed and the schedule is chosen by a profit maximizing airline, using four network types, finding that it is optimal for airlines to design networks and schedules to minimize the sum of airline and passenger costs.

In [21] the evolution of several airline networks from linear to hub system is demonstrated, and a description of hub's effect on airline cost and passenger travel time is reported. Hub and spoke systems could be regarded as one of the major change within the airline industry after the introduction in 1978 of the Airline Deregulation Act in U.S.A [11].

The equilibria for mergers and alliances under competition within a game theoretic framework is introduced by a network design in a non-competitive environment [1]. Their computation is pursued by adapting the non-competitive network design model to one of profit maximization, given the best responses of competitors in the field. The integration of near-surface temperature change as climate target in airline network design for a single airline is proposed in [4], in addition to economical targets.

The assessment of airline network design policy is an important goal for many countries. In order to achieve a high-performance Air Traffic Management infrastructure which will enable the safe and environmentally friendly development of air transport, Europan Commission is funding the SESAR Joint Undertaking partnership [22]. In 2011, the European Commission nominated EUROCONTROL as European Network Manager [9]. The main aim of these activities is reducing travel time, fuel consumption and CO2 emission.

A network design model could be regarded as a cooperative game model [23] which can describe decision-making processes and economic interactions of players.

We focus on network design games, namely cooperative games, where players share the profit of shipping some commodity from a given origin to a given destination. The profit is the revenue minus the cost of installing infrastructures on edges, in order to ship the commodity. Since the players may use different paths, there is the possibility to cooperate and design the optimal network satisfying the requests of all the players and minimizing the cost.

We present the network design game and study core solutions as a possible solution concept. A numerical procedure based on the ant colony approach is illustrated. The paper is organized as follows: in Sect. 2 we present the network design cooperative game; in Sect. 3 we define approximated core solutions for the game and a numerical procedure based on an ant colony approach; in Sect. 4 we discuss a test case for the airline network design game. Some concluding remarks are in Sect. 5.

2 A Network Design Model

Let us consider a set of players $N = \{1, \dots, n\}$ $(n \in N)$ and a graph $G = (V, E)$ where $V = \{1, \dots, k\}$ is the finite set of vertexes or nodes and $E = \{1, \dots, m\}$ the set of directed edges (k, m are natural numbers). Each player $i \in N$ has to ship $h_i > 0$ units of a commodity i between a given ordered pair of nodes (o_i, d_i) with $o_i, d_i \in V$, for any $i \in N$. We denote by $h = (h_1, \dots, h_n)$ vectors of R^n and by $OD = \{(o_1, d_1), \dots, (o_n, d_n)\}$ the set of origin/destination for any player.

From the shipment player i receives a return r_i. The initial capacity of each edge of E for accomodating shipments of the players' commodities is set at zero, and there is an investment cost $c_j(x)$ for installing x units of capacity on edge $j \in E$. Any coalition $S \subseteq N$ of players could construct capacities on the edges of E to create a capacitated network in which the requirements of any player of S are satisfied (admissible network). Coalition S chooses the admissible network of minimum cost. Define for any player $i \in N$ the set

$$P_i = \{\text{path connecting } o_i \text{ and } d_i\} \tag{1}$$

and for any edge $j \in E$ the set

$$Q_j = \{\text{path of edges from E including } j\}. \tag{2}$$

A path is the union of consecutive edges ($i\ j\ k$ is the path given by edge i, then edge j, then edge k).

For each player i in the coalition S, fix a path $p_i \in P_i$; then we consider the quantity

$$\sum_{j \in E} c_j\Big(\sum_{i: i \in S, p_i \in Q_j} h_i \Big) \tag{3}$$

that represents the sum of the costs of each edge j by considering all the players of coalition S that are using that edge j when they choose the paths $p_i \in P_i, \forall i \in S$.

We denote by $r = (r_1, \dots, r_n)$ the revenue profile vector $(r_i > 0)$ and $IC = (c_1, \dots, c_m)$ the installing cost functions ($c_j : [0, +\infty[\to [0, +\infty[$, $c_j(0) = 0$, c_j strictly increasing in $[0, +\infty[$). We call the tuple (N, G, h, OD, r, IC) a network design situation.

Definition 1 Given a network design situation (N, G, h, OD, r, IC), we define the network design cooperative game $< N, v >$ where $N = \{1, \dots, n\}$ is the set of the players and $v : 2^N \to R$ is the characteristic function such that $v(\emptyset) = 0$ and for each coalition $S \subseteq N$ the worth of the coalition is given by

$$v(S) = \sum_{i \in S} r_i - c(S) \tag{4}$$

being $c(S)$ the cost of the coalition S defined as

$$c(S) = \min_{p_i : p_i \in P_i, \forall i \in S} \sum_{j \in E} c_j \Big(\sum_{i : i \in S, p_i \in Q_j} h_i \Big) \tag{5}$$

A natural solution concept for this cooperative game is the core [17]. The core $\mathscr{C}(v)$ of the cooperative game $< N, v >$ is defined by

$$\mathscr{C}(v) = \{(x_1, \ldots, x_n) \in R^n : \sum_{i \in N} x_i = v(N), \tag{6}$$

$$\sum_{i \in S} x_i \geq v(S), \forall\ S \subseteq N\}, \tag{7}$$

i.e. the set of vectors in R^n satisfying the two properties:

$$\textbf{Budget balance} \qquad \sum_{i \in N} x_i = v(N) \tag{8}$$

$$\textbf{Core property} \qquad \sum_{i \in S} x_i \geq v(S), \forall\ S \subseteq N \tag{9}$$

The core of a network design cooperative game may be empty [24].

Example 1 Let us consider a network design situation (N, G, h, OD, r, IC) where $N = \{1, 2, 3\}$, $V = \{1, 2, 3, 4, 5, 6, 7\}$ and $E = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ (with $G = (V, E)), h = (1, 1, 1), OD = \big\{(1, 7), (3, 7), (5, 7)\big\}, r = (13, 12, 14), c_j(x) = \sqrt{x}$, $j \in E$. In this case (see Fig. 1)

$$P_1 = \big\{18, 67, 123457, 1239\big\},\ P_2 = \big\{2167, 28, 3457, 39\big\},\ P_3 = \big\{57, 49, 432167, 4328\big\} \tag{10}$$

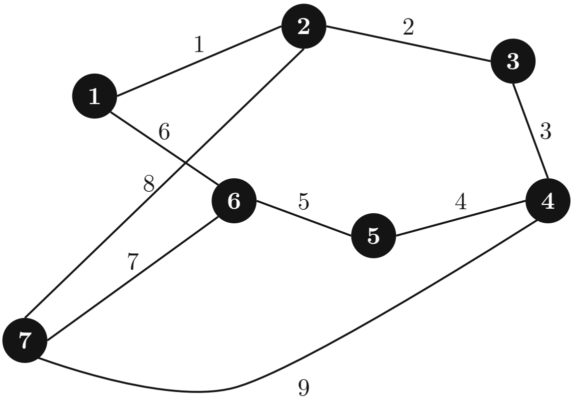


Fig. 1 Scheme for the network described into Example 1

$$
\begin{gathered}
Q_1 = \{18, 123457, 1239, 2167, 432167\},\ Q_2 = \{123457, 1239, 2167, 28, 432167, 4328\},\\
Q_3 = \{123457, 1239, 3457, 39, 432167, 4328\},\ Q_4 = \{123457, 3457, 49, 432167, 4328\},\\
Q_5 = \{123457, 3457, 57\},\ Q_6 = \{67, 2167, 432167\},\\
Q_7 = \{67, 123457, 2167, 3457, 57, 432167\},\ Q_8 = \{18, 28, 4328\},\ Q_9 = \{1239, 39, 49\}
\end{gathered} \tag{11}
$$

and it is easy to compute

$$
\begin{gathered}
c(\{1\}) = c(\{2\}) = c(\{3\}) = 2\\
c(\{1,2\}) = c(\{2,3\}) = c(\{1,3\}) = 2 + \sqrt{2}\\
c(\{1,2,3\}) = 4 + \sqrt{2}.
\end{gathered} \tag{12}
$$

The characteristic function is

$$
\begin{gathered}
v(\{1\}) = 11,\ v(\{2\}) = 10,\ v(\{3\}) = 12\\
v(\{1,2\}) = 23 - \sqrt{2},\ v(\{2,3\}) = 24 - \sqrt{2},\ v(\{1,3\}) = 25 - \sqrt{2}\\
v(\{1,2,3\}) = 35 - \sqrt{2}
\end{gathered} \tag{13}
$$

and the core of this game is empty.

If we consider $c_j(x) = x^2,\ j \in E$, the characteristic function is

$$
\begin{gathered}
v(\{1\}) = 11,\ v(\{2\}) = 10,\ v(\{3\}) = 12\\
v(\{1,2\}) = 21,\ v(\{2,3\}) = 22,\ v(\{1,3\}) = 23\\
v(\{1,2,3\}) = 33
\end{gathered} \tag{14}
$$

and the vector (11, 10, 12) is in the core.

We call minimum cost network any tuple $(\bar{p}_1, \ldots, \bar{p}_n)$ with $\bar{p}_i \in P_i$ for any $i \in N$ such that

$$
\min_{p_i : p_i \in P_i, \forall i \in N} \sum_{j \in E} c_j\Big(\sum_{i : i \in N, p_i \in Q_j} h_i \Big) = \sum_{j \in E} c_j\Big(\sum_{i : i \in N, \bar{p}_i \in Q_j} h_i \Big) \tag{15}
$$

Of course, there are different minimal cost network with cost $c(N)$, depending on the chosen path. In Example 1, the procedure gives as core solution the vector (11, 10, 12) and there are three optimal path giving the minimum cost (Fig. 2, red for player 1, blue for player 2 and green for player 3).

Remark 1 In the special case where the following assumption is satisfied

$$
\text{each } P_i \text{ consists of a single path } p_i' \tag{H}
$$

the characteristic function is

$$
v(S) = \sum_{i \in S} r_i - \sum_{j \in E} c_j\Big(\sum_{i : i \in S, p_i' \in Q_j} h_i \Big), \tag{16}
$$

by assuming concave cost functions $c_j,\ j \in E$, the cooperative game is a convex game and there exist core solutions [24].

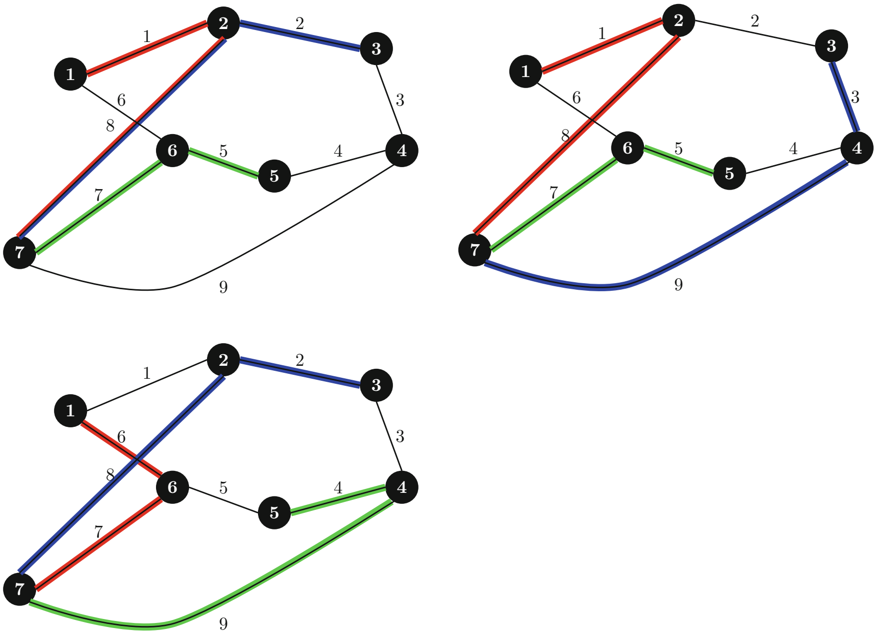


Fig. 2 Optimal paths of Example 1

Remark 2 Let us observe that the network design situation, given the installing cost functions IC and without revenue, is nothing but the congestion situation [14, 15], studied from a noncooperative point of view. For such games, there exists a pure Nash equilibrium, because they are potential games.

3 The Modified Network Design Model

Let us consider the graph G as given in Sect. 2 and an additional data set: a m-dimensional vector $D = (\delta_1, \ldots, \delta_m)$ where $\delta_j > 0$ is a weight on edge j for any $j = 1, \ldots, m$. We consider the network design situation as (N, G, h, OD, r, IC, D) where N, G, h, OD, r, IC are defined in Sect. 2.

Definition 2 Given a network design situation (N, G, h, OD, r, IC, D), we define the network design cooperative game $< N, v >$ where $N = \{1, \ldots, n\}$ is the set of the players and $v : 2^N \to R$ is the characteristic function such that $v(\emptyset) = 0$ and for each coalition $S \subseteq N$ the worth of the coalition is given by

$$v(S) = \sum_{i \in S} r_i - c(S) \tag{17}$$

being $c(S)$ the cost of the coalition S defined as

$$c(S) = \min_{p_i : p_i \in P_i, \forall i \in S} \sum_{j \in E} \Big[c_j \Big(\sum_{i : i \in S, p_i \in Q_j} h_i \Big) + \delta_j \Big] \tag{18}$$

Unfortunately, also in this case the core may be empty as shown in the next example.

Example 2 Let us consider a network design situation (N, G, h, OD, r, IC, D) where $N = \{1, 2, 3\}$, $V = \{1, 2, 3, 4, 5, 6, 7\}$ and $E = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ (with $G = (V, E)$ given as in Example 1), $h = (13, 12, 14)$, $OD = \big\{(1, 7), (3, 7), (5, 7)\big\}$, $r = (13, 12, 14)$, $c_j(x) = \sqrt{x}$, $j \in E$ and we consider the vector of weights on edges $D = (0.5, 0.5, 0.5, 0.6, 0.6, 0.6, 0.5, 0.7, 0.8)$.

In this case it is easy to compute

$$\begin{gathered} c(\{1\}) = 3.1, c(\{2\}) = 3.2, c(\{3\}) = 3.1 \\ c(\{1, 2\}) = 3.7 + \sqrt{2}, c(\{2, 3\}) = 3.9 + \sqrt{2}, c(\{1, 3\}) = 3.7 + \sqrt{2} \\ c(\{1, 2, 3\}) = 6.8 + \sqrt{2}. \end{gathered} \tag{19}$$

The characteristic function is

$$\begin{gathered} v(\{1\}) = 9.9, v(\{2\}) = 8.8, v(\{3\}) = 10.9 \\ v(\{1, 2\}) = 21.3 - \sqrt{2}, v(\{2, 3\}) = 22.1 - \sqrt{2}, v(\{1, 3\}) = 23.3 - \sqrt{2} \\ v(\{1, 2, 3\}) = 32.2 - \sqrt{2} \end{gathered} \tag{20}$$

and the core is empty.

This leads to consider approximate core solutions. As done in [19], we will relax the Budget balance property (8) as follows: we search vectors satisfying for a real number $\gamma \geq 1$ the property

$$\gamma - \textbf{Budget balance} \qquad v(N) \leq \sum_{i \in N} x_i \leq \gamma v(N) \tag{21}$$

The set of vectors in R^n satisfying the γ-Budget balance property (21) and the core property (9) is called the γ-core.

In Example 2 it is possible to compute a lower bound for γ in such a way that the corresponding γ-core is not empty. More precisely it results that $\gamma \geq 1.014$. Consider the smallest value $\bar{\gamma} = 1.014$: the solution of the set of conditions

$$\begin{gathered} x_1 + x_2 + x_3 = 1.014(32.2 - \sqrt{2}) \\ x_1 \geq 9.9, \ \ x_2 \geq 8.8, \ \ x_3 \geq 10.9 \\ x_1 + x_2 \geq 21.3 - \sqrt{2}, \ \ x_1 + x_3 \geq 23.3 - \sqrt{2}, \ \ x_2 + x_3 \geq 22.1 - \sqrt{2} \end{gathered} \tag{22}$$

is (10.54, 9.34, 11.34) that is a vector of the $\bar{\gamma}$-core.

Being $\gamma \geq 1$ and for $\gamma = 1$ the 1-core is nothing but the core, the quantity $\gamma - 1$ represent the cost of cooperation: it is the quantity necessary to find the core solution.

The non emptiness of the γ-core is guaranteed by the Bondareva-Shapley theorem [19].

3.1 Computational Procedure

A numerical procedure to compute core solutions given a network design situation (N, G, h, OD, r, IC) can be divided in two phases:

- 2^n combinatorial optimization problems must be solved to find the shortest paths for players, considering all the feasible coalitions;
- once collected all the coalition values, the core can be computed by using a constrained optimization procedure.

The first phase is based on an ant colony optimization algorithm (ACO) to find the shortest path with variable weight on edges in line with previous papers [5, 7, 8, 16, 18].

ACO is a metaheuristic approach based on ant social behaviour. The organization of an ant colony and its interaction rules have been developed in a computer system to replicate the ability to find shortest paths (or equivalent cost functions) in a well-defined environment.

The ant agent is a heuristic algorithm, capable to move between the nodes of a graph to look for an optimal path. However a heuristics could find a suboptimal trajectory due to the non-linearity of the objective function.

To overcome the problem, an ACO use a colony of ants that can communicate through the so-called *stigmercy*, a form of indirect communication induced by chemical changes applied to the environment. This kind of communication is based on a pheromone trail, that represents a sort of shared memory for the ant colony, spread over the chosen paths.

Let us consider the minimization of a typical N-P hard combinatorial problem. Artificial ants will construct solutions making random steps on a graph $G = (V, E)$ where V represents the set of nodes and E the set of edges.

A pheromone trail τ_j and a heuristics η_j can be associated to each edge j whose endpoints are the nodes a_j and b_j. Usually η_j is strictly connected to the cost function, giving to the ants a valuable support in choosing the best component routes.

Analogously to other evolutionary approaches, the ACO is a succession of several procedures, were epochs represent the time-base of the evolution. For each epoch, ants concurrently build solutions moving themselves on the construction graph, on the basis of the pheromone trails and heuristic information.

In the state $x_r = \{x_{r-1}, j\}$, the ant moves to a node j of its neighbourhood $N^k(x_r)$ via a probabilistic choice biased on a proportional-random rule:

$$p_j^k = \frac{[\tau_j]^\alpha [\eta_j]^\beta}{\sum_{l \in L} [\tau_l]^\alpha [\eta_l]^\beta} \tag{23}$$

where p_j^k is the probability of transition on edge j for the k-th ant via node b_j; τ is the pheromone information associated to the array of edges; η is the heuristic information which definition depends on the specific problem; exponents α and β are parameters used to bias the influence of pheromone trail, and L is the set of edges connected to node a_j.

In our implementation, the heuristic information is based on the length of edges:

$$\eta_j = 1/d_j \tag{24}$$

When one of termination criteria is meet, the ant stops and the algorithm start the upgrade procedure based on AS_{RANK} ([5]) metaphor, where only the $(w-1)$ rank-ordered ants, plus the best-so-far one (the best solution found until the current epoch) are allowed to deposit pheromone.

The best-so-far ant will furnish the greatest feedback, expressed by the weight w, while the rank-based ants will contribute with a rank-weighted amount of pheromone, as expressed by the following expression:

$$\tau_j = \tau_j + \sum_{r=1}^{w-1}(w-r)\Delta\tau_j^r + w\Delta\tau_j^{bsf} \tag{25}$$

with $\Delta\tau_j^k = 1/C^r$ and $\Delta\tau_j^{bsf}$.

To avoid a monotonic growth of the pheromone, allowing the system to forget bad choices and weak solutions, the algorithm preliminarily provides for a reduction of the overall pheromone intensity via an evaporation factor by using the following expression:

$$\tau_j = (1-\rho)\tau_j \qquad \forall(j) \in L \tag{26}$$

where $\rho \in [0, 1]$ denotes the pheromone evaporation rate.

It should be remarked that all the ants move in parallel and independently of one another, except for the pheromone tracking phase, that is performed in a synchronous mode. This circumstance can be seen as a sort of shared learning, where each ant does not adapt itself, but adapt the representation of the problem for the other ones.

Concerning the second phase of the computational procedure, for the core solutions or approximate core solutions, we solve an optimization problem to find the smallest $\gamma \geq 1$ for which the γ-core is nonempty. It consists of minimizing γ constrained with $2^n - 2$ conditions of core property (9) and the γ-budget property (18), considering the optimization variables $(x_1, \dots, x_n, \gamma) \in R^n \times [1, +\infty[$.

4 Airline Network Design

In line with previous papers [3, 23] we present the particular case of airline network design.

Let us consider k given cities and a graph $G = (V, E)$ where $V = \{1, \ldots, k\}$ is the finite set of vertexes or nodes where the cities are located and $E = \{1, \ldots, m\}$ the set of directed edges representing the possible airline routes between pair of cities. Suppose that n airline companies intend to flight on this set of cities, particularly company i wants to connect a given ordered pair of cities (o_i, d_i) with $o_i, d_i \in V$, for any $i \in N$. Here $N = \{1, \ldots, n\}$ represents the set of the airline companies, that are cooperating players in this model.

The initial capacity of each edge of E for transportation is set at zero, and there is an investment cost $c_j(x)$ for installing a new operating flight on edge $j \in E$ for x passengers. The ith company has an income r_i from the transportation of h_i units from o_i to d_i (these numbers can be interpreted as the estimated number of passengers).

Any coalition $S \subseteq N$ of airline companies could construct capacities on the edges of E to create a capacitated network in which the requirements of any company (player) of S are satisfied (admissible network). The scope is to find an optimal network, namely the admissible network of minimum cost for the grand coalition $S = N$.

Let us denote by D the m-dimensional vector $D = (\delta_1, \ldots, \delta_m)$ where δ_j is the distance between the two airports linked by the route j. We consider the distance between any pair of nodes as an additional criterium in order to achieve the minimal cost network (in the following these distances are rescaled with a positive real number k, so $\delta_j = \frac{length\ j}{k}$ for any j). We consider the tuple (N, G, h, OD, r, IC, D) as an airline network design situation.

Definition 3 Given an airline network design situation (N, G, h, OD, r, IC, D), we define the network design cooperative game $< N, v >$ where $N = \{1, \ldots, n\}$ is the set of the airline companies and $v : 2^N \to R$ is the characteristic function such that $v(\emptyset) = 0$ and for each coalition $S \subseteq N$ the worth of the coalition is given by

$$v(S) = \sum_{i \in S} r_i - c(S) \tag{27}$$

being $c(S)$ the cost of the coalition S defined as in (18).

We are interested in core solutions or approximate core solutions for the cooperative game defined above. The problem is solved by means of the procedure presented in Sect. 3.1 and we show the results of a test case concerning the cooperation of three airline companies dealing with operative flights reaching Narita Airport in Tokyo. The real data are the routes lenghts from [20]. The model, as given in this paper, leads to a *cooperation suggestion* situation where the involved companies try to make coalitions in order to share the operating flight between possible cities.

Example 3 Let us consider the airline network design situation (N, G, h, OD, r, IC, D) where $G = (V, E)$ and:

1. $N = \{AirFrance, Lufthansa, Delta\}$ is the set of the airline companies involved;

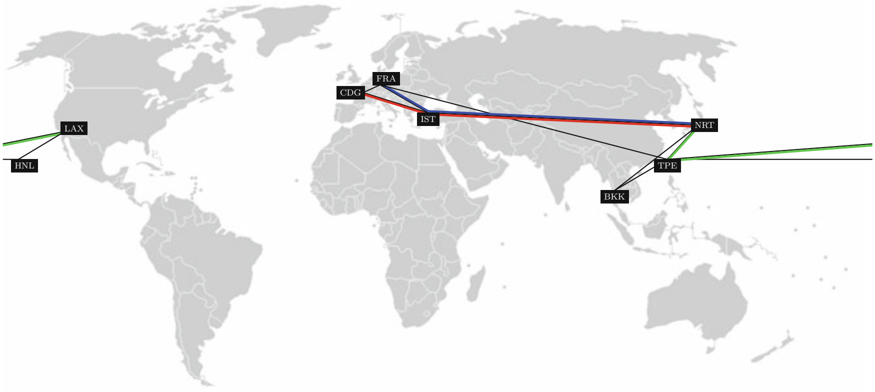


Fig. 3 Minimum cost network for Example 3

2. $V = \{Paris, Istanbul, Frankfurt, Honolulu, LosAngeles, Bangkok, Tokyo, Taiwan\}$ is the set of the cities, denoted in short by the corresponding airport acronym $V = \{CGD, IST, FRA, HNL, LAX, BKK, NRT, TPE\}$
3. $E = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$ is the set of the possible routes denoted also $E = \{CDG - FRA, CDG - IST, FRA - IST, FRA - TPE, IST - NRT, BKK - NRT, BKK - TPE, TPE - NRT, TPE - HNL, TPE - LAX, HNL - LAX\}$
4. $h = (1, 1, 1)$
5. $OD = \big\{(CDG, NRT), (FRA, NRT), (LAX, NRT)\big\}$
6. $r = (13, 12, 14)$,
7. $c_j(x) = \sqrt{x}, \ j \in E$
8. $D = (447, 2239, 1866, 9377, 9006, 4654, 2492, 2183, 8153, 10934, 4113)$ in Kilometers, multiplied by 10^{-3} ([20]).

In this case the characteristic function is

$$\begin{gathered} v(\{1\}) = 9.8755, v(\{2\}) = 8.9128, v(\{3\}) = 10.6883 \\ v(\{1, 2\}) = 19.9709, v(\{2, 3\}) = 20.3364, v(\{1, 3\}) = 20.2917 \\ v(\{1, 2, 3\}) = 30.9630 \end{gathered} \tag{28}$$

and the vector (10.3218, 9.6481, 10.6873) is in the core.

The computational procedure gives the following minimal cost network (Fig. 3, red for player 1, blue for player 2 and green for player 3):

The optimal network suggests Air France and Lufthansa to share the flight from Instanbul to Tokyo, while Delta flights separately from Los Angeles via Taiwan. In this example Player 1 and Player 2 end up to a *sky team*.

5 Concluding Remarks

In the paper we presented the airline network design model in the context of cooperative games. The main idea is to search the optimal path in a given network minimizing the costs due to the distance of the arc and also to the number of users on that arc. So, this implies that cooperation could help in a cost sharing process.

The model has been applied to a very popular problem, namely the airline network design: the formation of alliances between a group of airline companies is very actual subject. Our model, depending on the cost functions, suggests that cooperation can lead to minimal costs and the final network has the minimum cost property.

We studied a simplified model of the airline network design by considering the number of users of a given route and the lenght of that route. It is possible to refine the investigation adding in the model additional optimizing criteria, for example the fuel consumption or the passengers flow. This analysis will be developed in a forthcoming paper.

References

1. Adler, N., Smilowitz, K.: Hub-and-spoke network alliances and mergers: Price-location competition in the airline industry. Transp. Res. Part B **41**, 394–440 (2007)
2. Avrachenkov, K., Elias, J., Martignon, F., Neglia, G., Petrosyan, L.: A Nash bargaining solution for cooperative network formation games. In: Proceedings of Networking 2011, Valencia, Spain, 9–13 May 2011
3. Bittlingmayer, G.: Efficiency and entry in a simple airline network. Int. J. Ind. Organ. **8**, 245–257 (1990)
4. Braun, M., Koch, A., Dahlmann, K., Grewe, V., Gollnick, V.: An airline network design approach considering environmental and economical targets. iN: 27th International Congress of the Aeronautical Sciences, Nice, France (2010)
5. Bullnheimer, B., Hartl, R.F., Strauss, C.: A new rank-based version of the ant system: a computational study. Cent. Eur. J. Oper. Res. Econ. **7**(1), 25–38 (1997)
6. Chen, H., Roughgarden, T., Valiant, G.: Designing networks with good equilibria. In: SODA '08/SICOMP '10 (2008)
7. DAmato, E., Iuspa, L., Del Core, G.: A distributed ant colony algorithm for topological optimization of 2D structural domains. iN: Proceedings of EUROGEN2011 Evolutionary and Deterministic Methods for Design, Optimization and Control, pp. 108–123 (2011)
8. Dorigo, M., Blum, C.: Ant colony optimization theory: a survay. Theor. Comput. Sci. **344**, 243–278 (2005)
9. Eurocontrol: European Route Network Improvement Plan, Part 1: Airspace Design Methdology, https://www.eurocontrol.int/sites/default/files/publication/files/ernip-part-1-airspace-design-methodology$_0$.pdf (2015)
10. Gilles, R.P., Chakrabarti, S., Sarangi, S.: Nash equilibria of network formation games under consent. Math. Soc. Sci. **64**, 159–165 (2012)
11. Jaillet, P., Song, G., Yu, G.: Airline network design and hub location problems. Locat. Sci. **4**, 195–211 (1996)
12. Lederer, P.J.: A competitive network design problem with pricing. Transp. Sci. **27**, 2538 (1993)
13. Lederer, P.J., Nambimadom, R.S.: Airline network design. Oper. Res. **46**, 785804 (1998)
14. Mallozzi, L.: An application of Optimization Theory to the study of equilibria for games: a survey. Cent. Eur. J. Oper. Res. **21**(3), 523–539 (2013)

15. Monderer, D., Shapley, L.S.: Potential games. Games Econ. Behav. **14**, 124–143 (1996)
16. Marinakis, Y., Migdalas, A., Pardalos, P.M.: Expanding neighborhood searchGRASP for the probabilistic traveling salesman problem. Optim. Lett. **2**, 351–361 (2008)
17. Moulin, H.: Game Theory for the Social Sciences, Second and Revised Edition. New York University Press, New York (1986)
18. Neumann, F., Witt, C.: Ant Colony Optimization and the minimum spanning tree problem. Theor. Comput. Sci. **411**, 2406–2413 (2010)
19. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: Algorithmic Game Theory. Cambridge University Press, New York (2007)
20. OAG Worldwide Limited: Official Airline Guide, OAG Bedford (2007)
21. Oum, T.-H., Tretheway, M.W.: Airline hub and spoke systems. J. Transp. Res. Forum **30**, 380–393 (1990)
22. Single European Sky ATM Research: http://www.sesarju.eu/newsroom/brochures-publications/release-5 (2015)
23. Sharkey, W.W.: Network models in economics. In: Bali, M.O., et al. (eds.) Handbooks in OR & MS, vol. 8, Ch. 9 (1995)
24. Topkis, D.: Supermodularity and Complementarity. Princeton University Press, Princeton NJ (1998)

Augmented Lagrangian Approach for Constrained Potential Nash Games

Lina Mallozzi and Domenico Quagliarella

Abstract An approach to the resolution of inequality constrained potential games based on a dual problem is here presented. The dual problem is solved by using a two-level optimization iterative scheme based on a linear program for the dual problem and a classical hybrid evolutionary approach for the primal problem. An application to a facility location problem in presence of obstacles is described.

1 Introduction

In this paper we present an approach to the resolution of inequality constrained potential games based on a dual problem. The dual problem is introduced to take into account the constraints and is based on a two-level optimization iterative scheme. The dual problem is approximated using a linear program in a discrete subdomain which is updated after each iteration and leads to a global suboptimal approximation of Lagrange multipliers. The primal problem, instead, is solved using a classical hybrid evolutionary approach which includes in the set of hybrid operators for local search, both classical gradient based methods like Broyden Fletcher Goldfarb Shanno (BFGS) algorithm [14, pp. 136–143] and advanced stochastic derivative-free algorithms like Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [8]. A formulation based on the augmented Lagrangian, capable of handling inequality constraints, is finally introduced to take into account any non-convexity in the objective function and constraints [14, pp. 523–524], [17].

The developed method is applied to the search of Nash equilibrium points of a classical constrained potential game, namely the facility location problem in presence

L. Mallozzi
Department of Mathematics and Applications, University of Naples Federico II,
Via Claudio 21, 80125 Naples, Italy
e-mail: lina.mallozzi@unina.it

D. Quagliarella (✉)
Fluid Mechanics Department, Multidisciplinary Analysis and Optimization Group,
CIRA—Italian Aerospace Research Centre, Via Maiorise, 81043 Capua, Italy
e-mail: d.quagliarella@cira.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_16

of obstacles, because in the potential case the problem of finding Nash equilibrium solution is an optimization problem. Two illustrative examples of such a game in the context of facility location games are here reported and their solutions illustrated.

2 Iterative Scheme

We consider an optimization problem (primal problem)

$$\min_{x:g(x)\leq 0} f(x) \tag{1}$$

where $f : D \subseteq \mathbb{R}^n \to \mathbb{R}, g : D \subseteq \mathbb{R}^n \to \mathbb{R}^m$ and the domain D is assumed nonempty. We say that $x \in D$ is feasible if $g(x) \leq 0$ (i.e. $g_i(x) \leq 0 \; \forall i = 1, \ldots, m$). Denote the optimal value of (1) by p^*.

Let us consider the Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ associated with the problem (1) defined by

$$L(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) \tag{2}$$

being λ_i is the Lagrangian multiplier associated with the ith constraint $g_i(x) \leq 0$.

For any λ, consider the Langrangian dual function or dual function

$$l(\lambda) = \inf_{x \in D} L(x, \lambda)$$

The dual function is concave. It gives lower bounds on the optimal value p^*:

Proposition 1 *For any* $\lambda \geq 0$ *we have*

$$l(\lambda) \leq p^* \tag{3}$$

Proof Let $\hat{x}$ a feasible point for (1) and $\lambda \geq 0$. Then $\sum_{i=1}^m \lambda_i g_i(\hat{x}) \leq 0$ and $L(\hat{x}, \lambda) = f(\hat{x}) + \sum_{i=1}^m \lambda_i g_i(\hat{x}) \leq f(\hat{x})$. Hence

$$l(\lambda) = \inf_{x \in D} L(x, \lambda) \leq L(\hat{x}, \lambda) \leq f(\hat{x})$$

for any feasible point $\hat{x}$, then the inequality (3) follows.

What is the best lower bound that can be obtained from the dual function l? This question leads to the optimization problem (dual problem)

$$\max_{\lambda:\lambda \geq 0} l(\lambda) \tag{4}$$

that is a convex problem since the objective to be maximized is concave and the constraint is convex. Denote by d^* the optimal value of (4). We have

$$d^* \leq p^* \tag{5}$$

(weak duality property). The difference $p^* - d^*$ is called the optimal duality gap and is always nonnegative. If the equality $p^* = d^*$ holds, then we say that the strong duality holds. Let us call x^* a primal optimal point, i.e. a solution of problem (1), and λ^* a dual optimal point, i.e. a solution of problem (4).

Recall the following result [7]:

Proposition 2 *Let $\lambda \geq 0$ such that $L(x, \lambda) = f(x) + \lambda g(x)$ admits a minimum point x^* in D. Suppose that $g(x^*) = 0$. Then x^* is primal optimal point.*

This leads to solve the problem $\inf_{x \in D} L(x, \lambda)$ for a given λ, that is (under the assumption of the previous proposition) equivalent to the following linear problem

$$\begin{array}{l} \max h \\ h \leq L(x, \lambda) \ \forall x \in D \end{array} \tag{6}$$

the solution of which is $\bar{h} = L(x^*, \lambda)$. Let us observe that if we solve

$$\begin{array}{l} \max h \\ h \leq L(x, \lambda) \ \forall x \in D' \end{array} \tag{7}$$

with $D' \subseteq D$, then the solution $\bar{h}'$ will satisfy $\bar{h}' \geq \bar{h}$.

We give in the following a procedure to compute an optimal primal point or a suboptimal primal point of problem (1). The methods requires that the constraint g is active at the optimum (as stated in proposition 2).

Solve the problem (7) with $D' = \{x_1, \ldots, x_N\} \subseteq D$ a discrete subset of D. Then with $\lambda \geq 0$ we solve the linear programming problem

$$\begin{array}{l} \max h \\ h \leq L(x_1, \lambda) \\ \cdots\cdots\cdots\cdots \\ h \leq L(x_N, \lambda) \end{array} \tag{8}$$

Let us call the solution (h', λ). This solution represents (as stated above) an upper limit to the dual function $l(\lambda)$, and the related multiplier vector λ is an approximation of the Lagrange multiplier related to the true dual optimum. This λ vector can be used to solve the unconstrained problem, and three different results are possible:

1. all constraints are active;
2. one or more constraint is violated (o equivalently, some constraints are inactive and other violated).
3. one or more constraint is inactive;

In the first case, proposition 2 ensures that the obtained solution is optimal. In the second case, the new unfeasible points found allow to add to the auxiliary problem a new set of inequalities that permits to find an updated solution vector (h'', λ') that will exclude from the optimal solution of the unconstrained problem the previously found unfeasible solution. In the third case, a feasible solution is found but some constraints are still not active. In this case, maximizing h may lower the values of λ^k if the g_k constraint is not active. If the k-th constraint has not influence on the problem, then the corresponding λ^k is equal to zero.

Summing up, a two step optimization process is iterated until a satisfactory equilibrium point is found. The optimization process is described by the following scheme:

1. The initial $(\lambda^k, k = 1, \dots, n)$ is arbitrarily chosen.
2. An evolutionary optimizer works to the minimization of the Lagrangian (2).
3. A linear programming solver maximizes the h in problem (8). The λ^k values are updated using the results of the optimization.
4. The process is repeated starting from step 2 until a satisfactory equilibrium is reached.

When the approximated convexity requirements for the objective-constraint space stated in [7] are satisfied, the procedure admits an equilibrium point that corresponds to the constrained optimum sought. Non-convex objective-constraint spaces can be treated using an augmented Lagrangian approach [14] or transforming the constraints using monotonic functions, as it is illustrated, again, in reference [7].

3 Augmented Lagrangian

Here the method illustrated in [14, pp. 523–524] is slightly reformulated to adapt it to the convention here followed on the constraints that are satisfied if their value is less than or equal to zero. As in the above cited reference, we suppose for simplicity that the problem has no equality constraints. Thus, using the proximal point approach, the constrained primal optimization problem (1) can be expressed as an unconstrained one:

$$\min_{x \in \mathbb{R}^n} F(x) \tag{9}$$

where

$$F(x) = \max_{\lambda \geq 0} \left\{ f(x) + \sum_{i=1}^{m} \lambda_i c_i(x) \right\} = \begin{cases} f(x) & \text{if } g(x) \leq 0 \\ \infty & \text{otherwise} \end{cases} \tag{10}$$

If x is infeasible then it is $c_i(x) > 0$ for some i. Therefore it is possible to have $F(x)$ infinite by choosing the λ_i arbitrarily large and positive and letting $\lambda_j = 0$ for all $j \neq i$. If x is feasible then it is $c_i(x) \leq 0$ for all $i = 1, \dots, m$, and the maximum is obtained at $\lambda = 0$ with $F(x) = f(x)$. Summing up, we have

$$\min_{x \in \mathbb{R}^n} F(x) = \min_{x : g(x) \leq 0} f(x) \tag{11}$$

which is the original inequality-constrained problem. Function (10) is not directly usable to obtain the constrained minimum of f because it is non-smooth. For practical uses F should be replaced by a smooth approximation $\hat{F}(x, \lambda^k, \mu_k)$ which depends on a penalty parameter μ_k and on a Lagrange multiplier estimate λ^k. This approximation is defined as follows:

$$\hat{F}(x, \lambda^k, \mu_k) = \max_{\lambda \geq 0} \left\{ f(x) + \sum_{i=1}^{m} \lambda_i c_i(x) - \frac{1}{2\mu_k} \sum_{i=1}^{m} \left(\lambda_i - \lambda_i^k\right)^2 \right\} \tag{12}$$

The final term in (12) applies a penalty for any move of λ away from the previous estimate λ^k. Therefore the new maximizer λ will remain proximal to the previous estimate λ^k. Since (12) is a bound-constrained quadratic problem in λ, separable in the λ_i, it can be easily solved explicitly, obtaining

$$\lambda_i = \begin{cases} 0 & \text{if } c_i(x) + \lambda_i^k / \mu_k \leq 0 \\ \lambda_i^k + \mu_k c_i(x) & \text{otherwise} \end{cases} \tag{13}$$

By substituting these values in (12), we find that

$$L_A(x, \lambda^k, \mu_k) = \hat{F}(x, \lambda^k, \mu_k) = f(x) + \sum_{i=1}^{m} \Psi\left(c_i(x), \lambda_i^k, \mu_k\right) \tag{14}$$

where Ψ is a function of three scalar arguments defined as follows:

$$\Psi(c, \lambda, \mu) \overset{\text{def}}{=} \begin{cases} -\frac{1}{2\mu}\lambda^2 & \text{if } c + \lambda/\mu \leq 0 \\ \lambda c + \frac{\mu}{2} c^2 & \text{otherwise} \end{cases} \tag{15}$$

An iterative optimization process can be set up by minimizing $L_A(x; \lambda^k, \mu_k)$ with respect to x. The classical approach consists in increasing gradually the value of μ_k at each iteration and in using the formula (13) to obtain a new estimate λ^{k+1} of the Lagrange multipliers.

The approach we take here is different because the estimation of Lagrange multipliers is again obtained through the solution of the linear programming problem (8), while two possible approaches are possible to obtain the new estimates of μ_k. The first one is the classical gradual increment of μ_k, while the second one uses the computed and stored values of the discrete subset D' to solve a new auxiliary optimization problem:

$$\mu_k = \min \mu \tag{16}$$

subject to:

$$L_A(x^*, \lambda^k, \mu) \leq L_A(x_i, \lambda^k, \mu) \qquad \forall i = 1, \ldots, N$$

where $x^* \in D'$ is the current feasible optimum. This approach has the advantage of giving the lower limit of μ_k needed to solve exactly the optimization problem in the discrete subset D'. For practical uses, however, a slightly increased value of the parameter $\mu_k^* > \mu_k$ should be given in the subsequent iteration because the algorithm has to search the optimum value in the whole feasible set D.

4 Optimization Procedure Implementation

The above described two-level optimization procedure was implemented coupling a linear programming code that is aimed to solve the approximated dual problem (8) in the discrete subset D' with an evolutionary optimizer that solves the smoothed primal problem (14). Figure 1 shows the coupling between the dual and primal optimizer and the data flux between each software module.

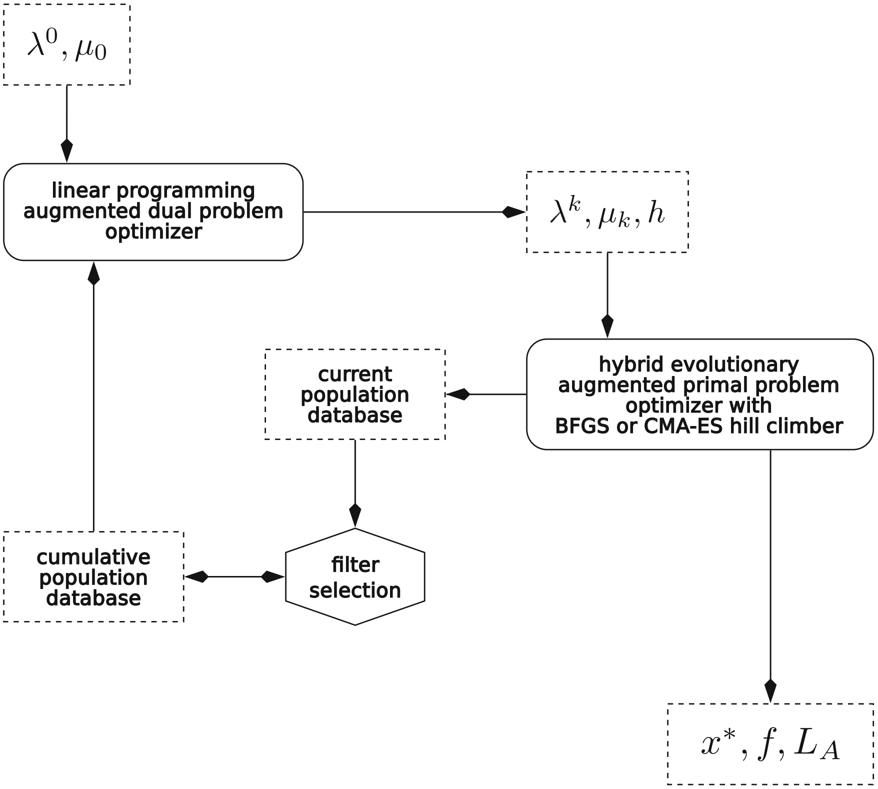

λ^0, μ_0
linear programming augmented dual problem optimizer
λ^k, μ_k, h
current population database
hybrid evolutionary augmented primal problem optimizer with BFGS or CMA-ES hill climber
cumulative population database
filter selection
x^*, f, L_A

Fig. 1 Augmented Lagrangian dual level optimizer scheme

The dual optimizer is based on the “GNU Linear Programming Kit” (GLPK). The GLPK [10] software package is intended for solving large-scale linear programming mixed integer programming, and other related problems. It is a set of routines written in ANSI C and organized in the form of a callable library.

The optimizer used for the primal problem is, instead, based on an in-house developed ANSI C library capable of solving both single and multi-objective problems using evolutionary computing methods [2, 15, 16]. A distinctive feature of this library is the ability to easily implement and use hybrid algorithms. Hybrid evolutionary algorithms are able to mix, at variable integration levels, two or more search techniques with, possibly, complementary features and have been one of the first techniques adopted for improving genetic algorithm performance, while keeping the desirable flexibility features of genetic algorithms [3]. The hybridization is here obtained coupling a plain genetic algorithm with two hill-climbing operators. The first one is based on the well known BFGS method for unconstrained minimization [9, 19], and the gradient of the objective function can be evaluated either using finite differences or through ad hoc defined software procedures. In case of finite differences both forward and central difference schemes can be used for gradient computation. The second operator, instead, is based on the CMA-ES [8] algorithm, that is an advanced stochastic derivative-free algorithms that belongs to the class of evolutionary algorithms and evolutionary computation. As in classical Evolution Strategies, new candidate solutions are sampled according to a multivariate normal distribution in $\mathbb{R}^n$. The covariance matrix of this distribution accounts for pairwise dependencies between the problem variables, and adaptation of the covariance matrix amounts to learning a second order model of the underlying objective function similar to the approximation of the inverse Hessian matrix in the Quasi-Newton method in classical optimization. The CMA-ES source code used to implement the operator was the one in C language provided by N. Hansen and available at https://www.lri.fr/~hansen/cmaes_inmatlab.html.

The interface between the two levels of the optimizer is realized in python language, as well as the filter that is used to select the individuals generated by the module primal and that are entered in the database (discrete set D') used by the dual module to update the estimates of λ^k, μ_k and h.

An elitism mechanism is used to preserve the best individual obtained during the evolution process.

5 A Test Problem with Duality Gap

The problem presented below is taken from [6] and its numerical primal solution is reported in [5]. The problem has $n = 8$ primal and $m = 22$ dual variables.

$$\begin{aligned}
&\min \quad x_1 + x_2 + x_3 \\
&\text{s. to:} \\
&\qquad 1000(-1 + 0.0025(x_4 + x_6) \leq 0 \\
&\qquad 1000(-1 + 0.0025(x_5 + x_7 - x_4)) \leq 0 \\
&\qquad 1000(-1 + 0.01(x_8 - x_5)) \leq 0 \\
&\qquad 0.001(-x_1 x_6 + 833.33252 x_4 + 100 x_1 - 83333.333) \leq 0 \\
&\qquad 0.001(-x_2 x_7 + 1250 x_5 + x_2 x_4 - 1250 x_4) \leq 0 \\
&\qquad 0.001(-x_3 x_8 + x_3 x_5 - 2500 x_5 + 1250000) \leq 0 \\
&\qquad 100 \leq x_1 \leq 10000,\ 1000 \leq x_2 \leq 10000,\ 1000 \leq x_3 \leq 10000 \\
&\qquad 10 \leq x_4 \leq 1000,\ 10 \leq x_5 \leq 1000,\ 10 \leq x_6 \leq 1000 \\
&\qquad 10 \leq x_7 \leq 1000,\ 10 \leq x_8 \leq 1000
\end{aligned} \tag{17}$$

The first six constraints are active at the minimum and the multiplier values are computed in [18] using the Karush-Kuhn-Tucker (KKT) optimality conditions.

In the present work, the CMA-ES hybrid operator was used in the primal optimizer. The problem is non-convex and thus required the augmented Lagrangian formulation to be solved. The best solution obtained is $\mathbf{x}^P = (579.325017, 1360.02555, 5109.89716, 182.019236, 295.604109, 217.980772, 286.415136, 395.604111)$ with $f^P = 7049.24773$, and it is in good agreement with the numerical solution given in [5].

The dual solution, instead, is given by the Lagrange multiplier vector computed solving the problem (8), before the correction provided by Eq. (13). The dual solution obtained is $\lambda_j^D = 0, \forall j \epsilon \{1, \ldots, 22\} \setminus \{7, 9, 11\}$ and is equal to 1, $\forall j \epsilon \{7, 9, 11\}$ with $h = 2100$. This result is in perfect agreement with the dual solution reported in [6].

6 Facility Location Problem

The multifacility location is concerned with the problem of locating n new facilities ($n \in \mathbb{N}, n > 1$) with respect to m existing facilities (demand points, $m \in \mathbb{N}$), in order to minimize a total cost function. We assume that there are n firms competing in order to locate the new facilities and the location problem can be stated as a Nash equilibrium problem [1]. Each firm decides the location of one facility.

The *facility location game* is the n-player strategic form game

$$\Gamma^{FL} =< N; S; f_1, \ldots, f_n >$$

where $N = \{1, ..., n\}$ and $S, f_1, \ldots, f_n$ are defined by the following assumptions:

(i) Each firm i has to set up a new facility in a point $x^i \in S \subset \mathbb{R}^2$, where S is the compact set of the feasible locations. Firms $1, \ldots, n$ are the players and S the strategy set of each player. We denote $x = (x^1, \ldots, x^n) \in S^n$ and $x^{-i} = (x^1, \ldots, x^{i-1}, x^{i+1}, \ldots, x^n)$.

(ii) m demand points (existing facilities) denoted by $t^k \in \mathbb{R}^2$, $k = 1, ..., m$ have to be connected to any new facility.

(iii) The function $d(y, z)$ is a measure of the distance between any two points y and z in $\mathbb{R}^2$. The distance $d(y, z)$ is expressed by a l_p norm in $\mathbb{R}^2$. A l_p norm is defined by $||v||_p = [|v_1|^p + |v_2|^p]^{1/p}$, $v = (v_1, v_2) \in \mathbb{R}^2$ and $1 \leq p \leq +\infty$; for $p = 2$ we have the Euclidean norm.

(iv) The transportation cost between two new facilities x^i and x^j is given by a function of the distance, i.e. $C\big(d(x^i, x^j)\big)$; the transportation cost between the new facility x^i and the demand point t^k is given by a function of the distance, i.e. $B\big(d(x^i, t^k)\big)$.

(v) The new facilities will be located in $(\bar{x}^1, ..., \bar{x}^n) \in S^n$ such that each firm i wants to minimize the total transportation cost

$$f_i(x^1, \ldots, x^n) = \sum_{1 \leq j \leq n, j \neq i} C\big(d(x^i, x^j)\big) +$$

$$\sum_{1 \leq k \leq m} B\big(d(x^i, t^k)\big)$$

The function f_i defined on S^n is the cost function of player ith.

A solution to the facility location game is a Nash equilibrium [1] of the game $\Gamma^{FL} =< N; S; f_1, \ldots, f_n >$, i.e. a vector $\bar{x} = (\bar{x}^1, \ldots, \bar{x}^n) \in S^n \subset \mathbb{R}^{2n}$ such that

$$f_i(\bar{x}) \leq f_i(\bar{x}^1, \ldots, \bar{x}^{i-1}, x^i, \bar{x}^{i+1}, \ldots, \bar{x}^n) \qquad \forall x^i \in S_i$$

for any $i \in N$.

It is possible to study the existence of Nash equilibrium strategies thanks the potential structure of the facility location game.

The facility location game Γ^{FL} is a potential game [11, 13], with potential function defined on S^n by

$$P(x^1, ..., x^n) = \sum_{1 \leq h < j \leq n} C\big(d(x^h, x^j)\big) +$$

$$\sum_{1 \leq j \leq n} \sum_{1 \leq k \leq m} B\big(d(x^j, t^k)\big).$$

Recall that the game $< N; S; \{f_i, i \in N\} >$ is called a *potential game* [11, 13] if there is a (*potential*) *function* $P : S^n \mapsto \mathbb{R}$ such that for all $i \in N$ and for each $x^{-i} \in S^{n-1}$

$$f_i(y, x^{-i}) - f_i(z, x^{-i}) = P(y, x^{-i}) - P(z, x^{-i}) \;\; \forall y, z \in S_i,$$

where $x^{-i} = (x^1, \ldots, x^{i-1}, x^{i+1}, \ldots, x^n)$.

Clearly, elements of argmin (P) are Nash equilibria of the game. This implies that the solutions to the facility location problem are the solution of the optimization problem

$$\min_{(x^1,\ldots,x^n)\in S^n} P(x^1,\ldots,x^n).$$

In situations where the functions C, B are decreasing in the distance, the admissible set can be reduced: for example [12] if $C(t) = B(t) = \frac{1}{t^2}, t > 0$ being t the distance between two facilities, we must define the facility location problem in the set

$$A = \{(x^1,\ldots,x^n) \in S^n : x^i \in S,\ x^i \neq x^j,\ x^i \neq t^k,$$

$$\forall i, j, k = 1,\ldots,n, j \neq i\}.$$

As it happens in several concrete situations, there could be a forbidden place (obstacle) in the region, say a subset $O \subset S$. The facility location problem has to be set in the admissible region $T = S \setminus O$ (or $T = A \setminus O$), and the facility location problem is solved by the constrained optimization problem

$$\min_{(x^1,\ldots,x^n)\in T^n} P(x^1,\ldots,x^n). \tag{18}$$

Usually the obstacle (convex set or non convex set) can be described by inequalities and the problem (18) can be solved with the iterative scheme of Sect. 2.

6.1 Test Case 1

Let us consider $S = [0,1]^2$, $n = m = 4$ and $C(t) = B(t) = \frac{1}{t^2}, t > 0$. We want to locate four facilities x^1, x^2, x^3, x^4 in the square given the demand points $t^1 = (0,0), t^2 = (1,0), t^3 = (1,1), t^4 = (0,1)$. Suppose that in this region there is a forbidden zone located outside the circle $\{x = (x_1, x_2) \in S : (x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.0625\}$.

The problem is to find a solution to (18) with $T = A \setminus O$ where

$$A = \{x = (x^1, x^2, x^3, x^4) \in S^n : x^i \in S,\ x^i \neq x^j,\ x^i \neq t^k,$$

$$\forall i, j, k = 1,\ldots,4, j \neq i\}$$

and

$$O = \{x \in A : (x_1^i - 0.5)^2 + (x_2^i - 0.5)^2 > 0.0625, i = 1,2,3,4\}.$$

In terms of coordinates, if $x^i = (x_1^i, x_2^i)$ and $t^k = (t_1^k, t_2^k)$ $(i, k = 1, 2, 3, 4)$, the potential function P is given by

$$P(x^1, x^2, x^3, x^4) = \sum_{1 \le i < j \le 4} \frac{1}{\sqrt{(x_1^i - x_1^j)^2 + (x_2^i - x_2^j)^2}} +$$

$$\sum_{1 \le i \le 4} \sum_{1 \le k \le 4} \frac{1}{\sqrt{(x_1^i - t_1^k)^2 + (x_2^i - t_2^k)^2}}.$$

The constrained problem is to find the minimum point of P in the set A with the four constraints

$$(x_1^i - 0.5)^2 + (x_2^i - 0.5)^2 \le 0.0625$$

for $i = 1, 2, 3, 4$.

Figure 2 reports the results obtained using the presented approach to the facility location test problem Sect. 6.1.

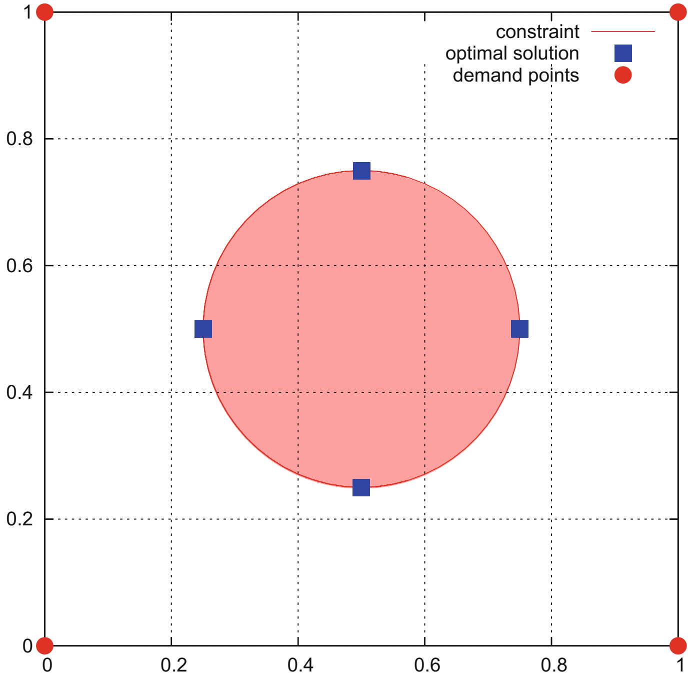


Fig. 2 Nash equilibrium to the constrained potential game Sect. 6.1

6.2 Test Case 2

The problem here reported is obtained by adding a further non-convex constraint to the test example described in Sect. 6.1. This new constraint reduces the feasible region to the set defined below:

$$\begin{aligned} T = \{x = (x_1, x_2) \in A : \\ &(x_1 - 0.5)^2 + (x_2 - 0.5)^2 \leq 0.0625, \\ &x_1 \leq x_2, x_2 \leq 1 - x_1, \text{ for } x_1 \geq 0.5\}. \end{aligned}$$

In terms of coordinates, if $x^i = (x_1^i, x_2^i)$ and $t^k = (t_1^k, t_2^k)$ $(i, k = 1, 2, 3, 4)$, the potential function $P(x^1, x^2, x^3, x^4)$ is the same as in Test case 1 and the constrained problem is to find the minimum point of P in the constrained set $A \setminus O$.

Figure 3 reports the results obtained using the presented approach to the above defined test problem.

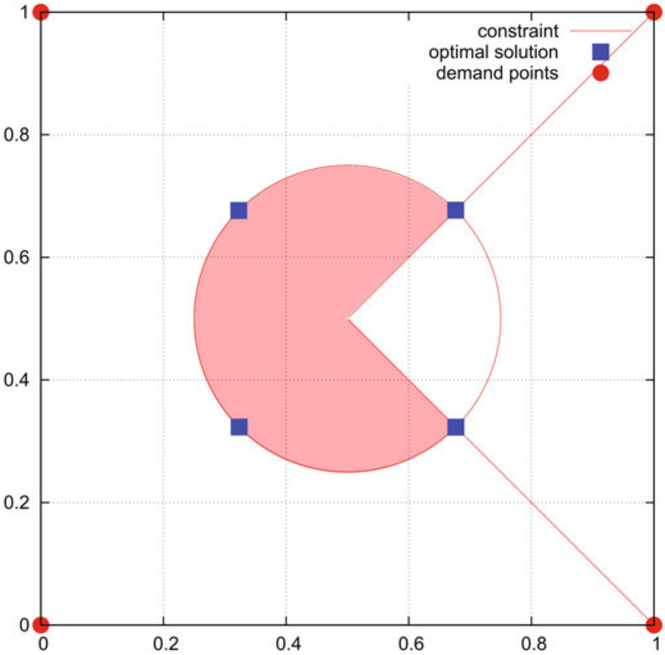


Fig. 3 Optimal Nash equilibrium solution to the constrained potential game Sect. 6.2

7 Conclusions

A technique for constrained optimization based on Lagrange multipliers and a non-cooperative two-player hierarchical game has been introduced and applied to the solution of constrained potential games. The two players in competition are two optimization procedures that solve two connected problems, namely an unconstrained minimization of the Lagrangian (2) related to the original problem (1), and a companion *linear* maximization problem (8) built using the values of the objective function and constraints computed by the first player in the previous step. When non-convex problems in the objective-constraint space have to be dealt with, the Lagrangian (2) is substituted by a smooth augmented Lagrangian formulation (14) which is able to address inequality constraints. The value of the penalty parameter μ_k is estimated at each iteration solving the auxiliary problem (16). The advantage of the proposed approach for the estimation of λ^k and μ_k, compared to better known and used techniques of local adjustment of these parameters, is that in their calculation and update all the past history of the optimization process is taken into account, and, consequently, the method is much better suited to global optimum search problems.

The developed evolutionary optimization framework was used to solve a constrained multifacility location problem that can be modeled within the framework of potential games. Both convex and non-convex constraint set were addressed successfully.

References

1. Başar, T., Olsder, G.: Dynamic Noncooperative Game Theory. In Classics in Applied Mathematics, 2nd edn. no. 23. Society for Industrial and Applied Mathematics, Philadelphia, PA (1999). https://doi.org/10.1137/1.9781611971132.
2. Catalano, L.A., Quagliarella, D., Vitagliano, P.L.: Aerodynamic shape design using hybrid evolutionary computing and multigrid-aided finite-difference evaluation of flow sensitivities. Eng. Comput. **32**(2), 178–210 (2015). https://doi.org/10.1108/EC-02-2013-0058.
3. Davis, L.: Handbook of genetic Algorithms. Van Nostrand Reinhold, New York (1991)
4. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, New York, NY, USA (2001)
5. Deb, K., Datta, R.: A fast and accurate solution of constrained optimization problems using a hybrid bi-objective and penalty function approach. In: 2010 IEEE Congress on Evolutionary Computation (CEC), pp. 1–8 (2010). https://doi.org/10.1109/CEC.2010.5586543
6. Deb, K., Gupta, S., Dutta, J., Ranjan, B.: Solving dual problems using a coevolutionary optimization algorithm. J. Glob. Optim. **57**(3), 891–933 (2012). https://doi.org/10.1007/s10898-012-9981-5.
7. Everett III, H.: Generalized lagrange multiplier method for solving problems of optimum allocation of resources. Oper. Res. **11**, 399–417 (1963)
8. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evolut. Comput. **9**(2), 159–195 (2001)
9. Luenberger, D.G.: Linear and Nonlinear Programming, 2nd edn. Addison-Wesley Inc., Reading, Massachusetts (1984)

10. Makhorin, A.: GNU Linear Programming Kit, Version 4.55. Free Software Foundation, 51 Franklin St, Fifth Floor, Boston, MA, 02110–1301, USA (2014). http://www.gnu.org/software/glpk/glpk.html
11. Mallozzi, L.: An application of optimization theory to the study of equilibria for games: a survey. Cent. Eur. J. Oper. Res. **21**(3), 523–539 (2013). https://doi.org/10.1007/s10100-012-0245-8.
12. Mallozzi, L., D'Amato, E., Daniele, E.: A game theoretical model for experiment design optimization. In: Rassias, T.M., Floudas, C.A., Christodoulos, A., Butenko S. (eds.) Optimization in Science and Engineering, chap. 18, pp. 357–368. Springer (2014). In Honor of the 60th Birthday of Panos M. Pardalos
13. Monderer, D., Shapley, L.S.: Potential games. Games Econ. Behav. **14**, 124–143 (1996)
14. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, New York (2006)
15. Quagliarella, D., Vicini, A.: A genetic algorithm with adaptable parameters. In: 1999 IEEE International Conference On Systems, Man, And Cybernetics. Institute of Electrical and Electronic Engineers (IEEE), Tokyo, Japan (1999). ISBN 0-7803-5734-5
16. Quagliarella, D., Vicini, A.: GAs for aerodynamic shape design I: general issues, shape parametrization problems and hybridization techniques. In: Périaux, J., Degrez, G., Deconinck, H. (eds.) Lecture Series 2000–07. Genetic Algorithms for Optimisation in Aeronautics and Turbomachinery. Von Karman Institute, Belgium (2000)
17. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis, Grundlehren der mathematischen Wissenschaften, vol. 317. Springer, Berlin, Heidelberg (1998)
18. Tulshyan, R., Arora, R., Deb, K., Dutta, J.: Investigating ea solutions for approximate kkt conditions in smooth problems. In: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, GECCO '10, pp. 689–696. ACM, New York, NY, USA (2010). https://doi.org/10.1145/1830483.1830609.
19. Vanderplaats, G.N.: Numerical Optimization Techniques for Engineering Design: with Applications. Mc Graw–Hill (1984)

A Diversity Dynamic Territory Nash Strategy in Evolutionary Algorithms: Enhancing Performances in Reconstruction Problems in Structural Engineering

David Greiner, Jacques Périaux, J. M. Emperador, B. Galván and G. Winter

Abstract Game-theory based Nash–evolutionary algorithms are efficient to speed-up and parallelize the optimum design procedure. They have been applied in several fields of engineering and sciences, mainly, in aeronautical and structural engineering. The influence of the search space player territory has been shown as having an important role in the algorithm performance. Here we present a study where a diversity enhanced dynamic player territory is introduced and its behavior is tested in a reconstruction problem in structural engineering. The proposed diversity dynamic territory seems to increase the optimization procedure robustness, and improves the results from a classical dynamic territory, in a structural frame test case.

1 Introduction

Enhancing the efficiency of global population-based stochastic optimizers is an important research focus, especially when the problem to solve is a real world problem implying a high CPU cost. That is the frequent case in solving optimum design problems in different fields of computational engineering (such as aeronautical engineering [1] or structural engineering [2]), where numerical methods such as finite element methods or boundary element methods are involved. One of the tools which has been shown promising in improving the efficiency of these methods, is through the use of game-theory based evolutionary algorithms.

D. Greiner (✉) · J. M. Emperador · B. Galván · G. Winter
Institute of Intelligent Systems and Numerical Applications in Engineering - SIANI, Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas, Spain
e-mail: david.greiner@ulpgc.es

J. Périaux
Mathematical Information Technology Department (MIT), University of Jyvaskyla, Finland & International Center for Numerical Methods in Engineering (CIMNE) – Universidad Politécnica de Cataluña, Barcelona, Spain
e-mail: jperiaux@gmail.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_17

Among the game-theory based evolutionary algorithms (EAs), Nash EAs [3–6] have been shown efficient to speed-up and parallelize the optimum design procedure. The Nash player domain decomposition territory has been proven a possible factor of great influence in the success of the approach and in the algorithm performance. Different domain decomposition territory are tested in this work in the context of a reconstruction problem in structural engineering, showing their impact in the algorithm performance. Moreover, a dynamic territory where the diversity of the population is enhanced is proposed, and compared with a standard 'a priori' dynamic territory. In Sect. 2, Nash EAs are described, including the dynamic territory procedure and introducing the diversity enhanced dynamic territory proposed here. The structural problem handled is explained in Sect. 3, and the frame structural test case in Sect. 4. This paper continues with the results and discussion in Sect. 5, and finalizes with the conclusions in Sect. 6.

2 Nash–Evolutionary Algorithms

Nash–EAs were introduced in Sefrioui et al. [3] for solving computational fluid dynamics problems in aeronautical engineering. They hybridize mathematical concepts from Nash equilibrium [7, 8] (competitive game theory where players maximize their payoffs while considering strategies of their competitors) in the evolutionary optimization process. A set of subpopulations co-evolve simultaneously dealing each of them, only with a territory of the search variables. These subpopulations interact towards the equilibrium. A virtual Nash game approach has been applied in inverse shape optimization computational fluid dynamics problems [5] and in structural engineering problems [9–12] as an improvement technique versus the standard panmictic EA. In virtual Nash games approach [13] only a single global objective is considered, linked to one discipline, in contrast of real Nash games where more than one objective and disciplines (multidisciplinary design) are taken into account. It has been applied with success in inverse problems where the fitness function is a sum of separable terms (such as the case of many shape optimization problems). The allocation of the search variables (genes of the chromosome) to the Nash player subpopulations defines the Domain Decomposition (DD) territory of the game-theory based EA. For every player (subpopulation) the chromosome genes are partitioned in two subsets: the first corresponds to the variables that will be considered as optimization variables in this subpopulation, and the second corresponds to the variables that will be assigned values from the best individuals of the other player/s (subpopulation/s). This distribution could have great influence in the results of the optimum design procedure [12, 13]. Among the factors to take into account in the domain decomposition allocation are the number of variables assigned to each territory and their distribution according to the physics of the handled problem. Three different territory Domain Decomposition (DD) strategies will be tested in this paper: Static Nash, Dynamic Nash and Diversity Dynamic Nash. They are explained as follows.

2.1 Static Nash Territory DD

This is the standard Nash EA, where the territory splitting among the players is allocated 'a priori' and it is fixed without changes throughout all the algorithm evolution, until the stopping criterion of the EA search is satisfied.

2.2 Dynamic Nash Territory DD

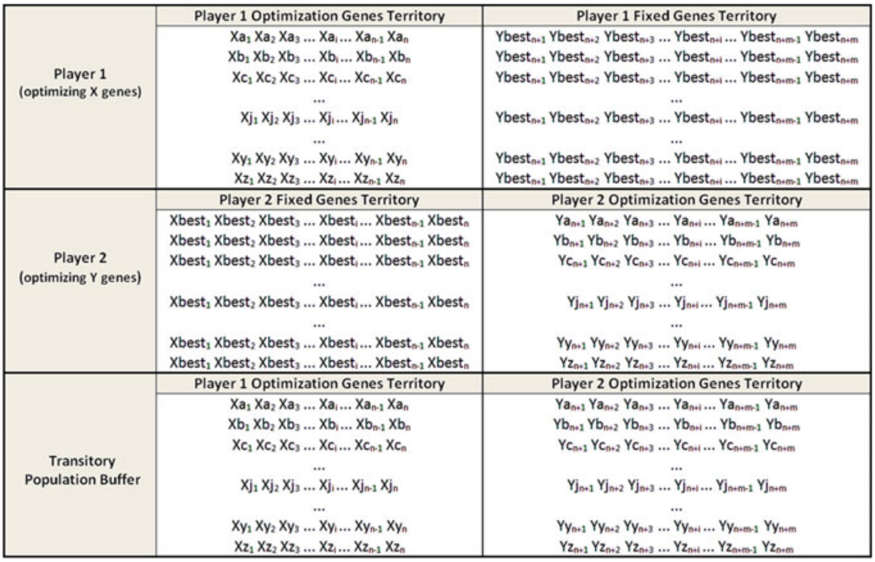
When the territory splitting is allowed to change during the algorithm evolution, then a dynamic territory DD is defined. The number of generations where the selected territory is fixed defines the 'epoch' parameter. Therefore, every epoch generations, the territory DD changes during the optimization procedure. In this paper, this territory DD will be chosen randomly every time the epoch is activated, among a predefined set of possibilities.

2.3 Diversity Dynamic Nash Territory DD

A diversity enhanced dynamic Nash territory DD is an approach introduced in [12] (called there Nash-EA dynamic DD and here diversity dynamic territory (DDT) domain decomposition). When the epoch is activated, this DDT constructs a transitory population buffer where each chromosome is composed of the optimized (not fixed) territory of each player of the virtual Nash game; see schematic process in Fig. 1. This transitory population buffer is taken as reference for constructing the next Nash player subpopulations with the new chosen territory DD (where genes will be fixed by the best individual values of the new allocation distribution according to each player allocation). Next, the influence of these territory domain decompositions is tested in the context of the reconstruction problem in structural engineering, comparing the Nash strategies defined in Sects. 2.1–2.3.

3 The Structural Problem

This work handles with the structural reconstruction problem. Its aim is to achieve the structure which most fits the maximum reference stresses. The optimum structural bar design is defined as a design in which some allocation of every bar in the structure has a maximum stress value as accurately equal as the maximum reference stress for that bar. Expression (1) shows the fitness function (FF) to be minimized (reconstruction problem).



Player 1 (optimizing X genes)	Player 1 Optimization Genes Territory	Player 1 Fixed Genes Territory
	$Xa_1\ Xa_2\ Xa_3 \dots Xa_i \dots Xa_{n-1}\ Xa_n$	$Ybest_{n+1}\ Ybest_{n+2}\ Ybest_{n+3} \dots Ybest_{n+i} \dots Ybest_{n+m-1}\ Ybest_{n+m}$
	$Xb_1\ Xb_2\ Xb_3 \dots Xb_i \dots Xb_{n-1}\ Xb_n$	$Ybest_{n+1}\ Ybest_{n+2}\ Ybest_{n+3} \dots Ybest_{n+i} \dots Ybest_{n+m-1}\ Ybest_{n+m}$
	$Xc_1\ Xc_2\ Xc_3 \dots Xc_i \dots Xc_{n-1}\ Xc_n$	$Ybest_{n+1}\ Ybest_{n+2}\ Ybest_{n+3} \dots Ybest_{n+i} \dots Ybest_{n+m-1}\ Ybest_{n+m}$
	...	...
	$Xj_1\ Xj_2\ Xj_3 \dots Xj_i \dots Xj_{n-1}\ Xj_n$	$Ybest_{n+1}\ Ybest_{n+2}\ Ybest_{n+3} \dots Ybest_{n+i} \dots Ybest_{n+m-1}\ Ybest_{n+m}$
	...	...
	$Xy_1\ Xy_2\ Xy_3 \dots Xy_i \dots Xy_{n-1}\ Xy_n$	$Ybest_{n+1}\ Ybest_{n+2}\ Ybest_{n+3} \dots Ybest_{n+i} \dots Ybest_{n+m-1}\ Ybest_{n+m}$
	$Xz_1\ Xz_2\ Xz_3 \dots Xz_i \dots Xz_{n-1}\ Xz_n$	$Ybest_{n+1}\ Ybest_{n+2}\ Ybest_{n+3} \dots Ybest_{n+i} \dots Ybest_{n+m-1}\ Ybest_{n+m}$
Player 2 (optimizing Y genes)	Player 2 Fixed Genes Territory	Player 2 Optimization Genes Territory
	$Xbest_1\ Xbest_2\ Xbest_3 \dots Xbest_i \dots Xbest_{n-1}\ Xbest_n$	$Ya_{n+1}\ Ya_{n+2}\ Ya_{n+3} \dots Ya_{n+i} \dots Ya_{n+m-1}\ Ya_{n+m}$
	$Xbest_1\ Xbest_2\ Xbest_3 \dots Xbest_i \dots Xbest_{n-1}\ Xbest_n$	$Yb_{n+1}\ Yb_{n+2}\ Yb_{n+3} \dots Yb_{n+i} \dots Yb_{n+m-1}\ Yb_{n+m}$
	$Xbest_1\ Xbest_2\ Xbest_3 \dots Xbest_i \dots Xbest_{n-1}\ Xbest_n$	$Yc_{n+1}\ Yc_{n+2}\ Yc_{n+3} \dots Yc_{n+i} \dots Yc_{n+m-1}\ Yc_{n+m}$
	...	...
	$Xbest_1\ Xbest_2\ Xbest_3 \dots Xbest_i \dots Xbest_{n-1}\ Xbest_n$	$Yj_{n+1}\ Yj_{n+2}\ Yj_{n+3} \dots Yj_{n+i} \dots Yj_{n+m-1}\ Yj_{n+m}$
	...	...
	$Xbest_1\ Xbest_2\ Xbest_3 \dots Xbest_i \dots Xbest_{n-1}\ Xbest_n$	$Yy_{n+1}\ Yy_{n+2}\ Yy_{n+3} \dots Yy_{n+i} \dots Yy_{n+m-1}\ Yy_{n+m}$
	$Xbest_1\ Xbest_2\ Xbest_3 \dots Xbest_i \dots Xbest_{n-1}\ Xbest_n$	$Yz_{n+1}\ Yz_{n+2}\ Yz_{n+3} \dots Yz_{n+i} \dots Yz_{n+m-1}\ Yz_{n+m}$
Transitory Population Buffer	Player 1 Optimization Genes Territory	Player 2 Optimization Genes Territory
	$Xa_1\ Xa_2\ Xa_3 \dots Xa_i \dots Xa_{n-1}\ Xa_n$	$Ya_{n+1}\ Ya_{n+2}\ Ya_{n+3} \dots Ya_{n+i} \dots Ya_{n+m-1}\ Ya_{n+m}$
	$Xb_1\ Xb_2\ Xb_3 \dots Xb_i \dots Xb_{n-1}\ Xb_n$	$Yb_{n+1}\ Yb_{n+2}\ Yb_{n+3} \dots Yb_{n+i} \dots Yb_{n+m-1}\ Yb_{n+m}$
	$Xc_1\ Xc_2\ Xc_3 \dots Xc_i \dots Xc_{n-1}\ Xc_n$	$Yc_{n+1}\ Yc_{n+2}\ Yc_{n+3} \dots Yc_{n+i} \dots Yc_{n+m-1}\ Yc_{n+m}$
	...	...
	$Xj_1\ Xj_2\ Xj_3 \dots Xj_i \dots Xj_{n-1}\ Xj_n$	$Yj_{n+1}\ Yj_{n+2}\ Yj_{n+3} \dots Yj_{n+i} \dots Yj_{n+m-1}\ Yj_{n+m}$
	...	...
	$Xy_1\ Xy_2\ Xy_3 \dots Xy_i \dots Xy_{n-1}\ Xy_n$	$Yy_{n+1}\ Yy_{n+2}\ Yy_{n+3} \dots Yy_{n+i} \dots Yy_{n+m-1}\ Yy_{n+m}$
	$Xz_1\ Xz_2\ Xz_3 \dots Xz_i \dots Xz_{n-1}\ Xz_n$	$Yz_{n+1}\ Yz_{n+2}\ Yz_{n+3} \dots Yz_{n+i} \dots Yz_{n+m-1}\ Yz_{n+m}$

Fig. 1 Diversity dynamic Nash territory DD: transitory population buffer (every epoch). Example in 2 player case

$$Fitness\ Function = \sqrt{\sum_{i=1}^{Nbars} (\sigma_{MAX-i} - \sigma_{MAX-Ri})^2} \tag{1}$$

where $\sigma_{MAX\text{-}i}$ is the maximum calculated stress and $\sigma_{MAX\text{-}Ri}$ the maximum reference stress, in bar i. A null value of FF (or reconstruction problem) means a perfect match of stresses and a location of the aimed structural design.

4 Test Case

A fifty-five bar sized frame structural test case [14] is handled here shown in Fig. 2, and considering the case of discrete cross-section type variables. Particularly, the frame structure of reference considered is the one corresponding to IPE330 cross section type in all beams (being the search space an interval between the IPE080 and the IPE500) and HEB450 cross section type in all columns (being the search space an interval between the HEB100 and the HEB450). Details about the corresponding maximum stress in each bar can be consulted in [11].

Here, a two player (two subpopulations) splitting territory approach will be used. Six Domain Decomposition (DD) player territories are tested in Sect. 5. The distribution of bars in every case are shown as follows in Fig. 3 (black and light gray colors indicate membership to each territory, respectively): an alternating every one

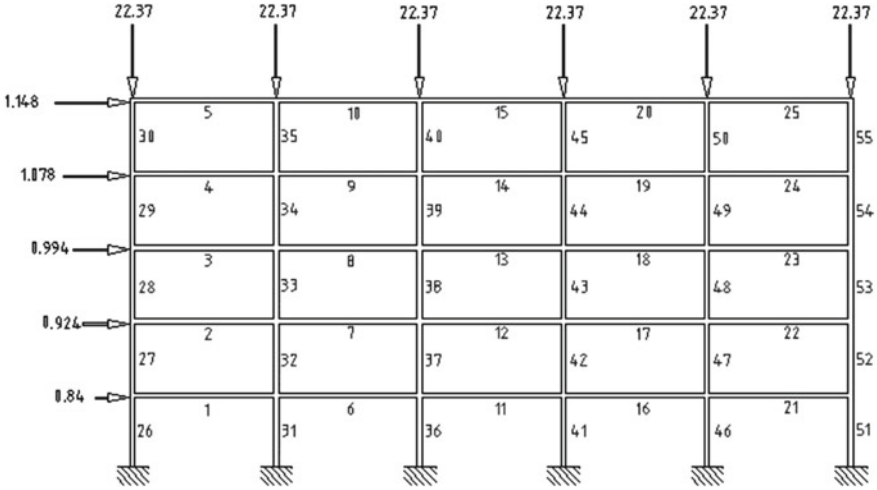
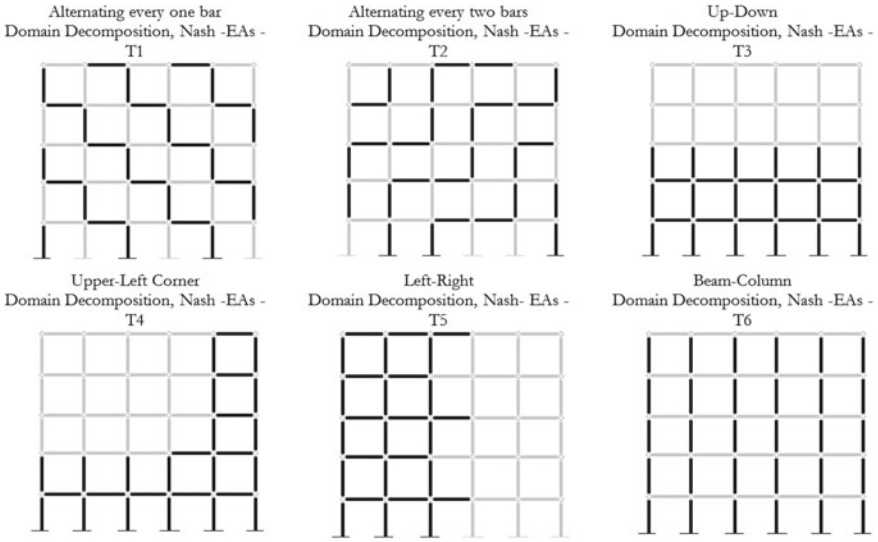


Fig. 2 Computational domain, boundary conditions and loadings available in [14]


Alternating every one bar Domain Decomposition, Nash -EAs - T1

Alternating every two bars Domain Decomposition, Nash -EAs - T2

Up-Down Domain Decomposition, Nash -EAs - T3

Upper-Left Corner Domain Decomposition, Nash -EAs - T4

Left-Right Domain Decomposition, Nash- EAs - T5

Beam-Column Domain Decomposition, Nash -EAs - T6

Fig. 3 Nash static T1-T6 domain decompositions bar frame

bar DD-T1, an alternating every two bars DD-T2, an up-down DD-T3, an upper-left corner DD-T4, a left-right DD-T5, and a beam-column DD-T6. In all cases territories divide the variable search space in two subsets of 27 and 28 bars; except in the beam-column DD, where subsets are of 25 and 30 bars, respectively. These set of DD territories will be also the predefined set of random selection DDs in the dynamic and diversity dynamic Nash EAs.

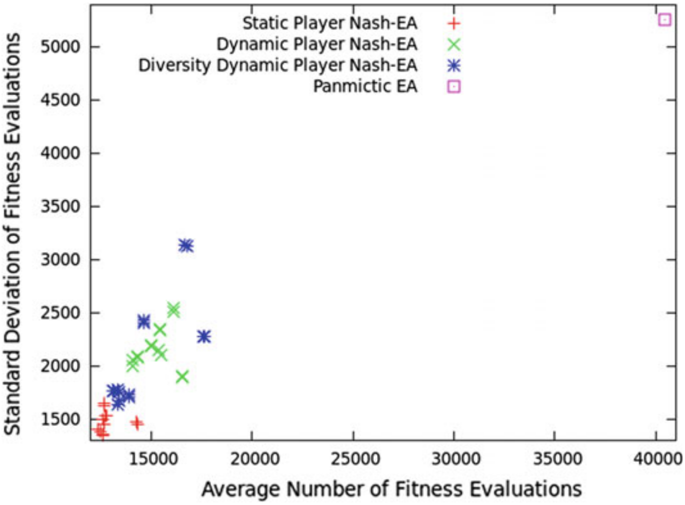


Fig. 4 Panmictic EA and Nash EAs results—fitness evaluations required to obtain the optimum design out of 30 independent runs

5 Results and Discussion

A population size of 80 individuals, a mutation rate of 0.4% and an elitist generational replacement strategy keeping the two best individuals in every generation has been used. Thirty independent runs have been executed in every DD case, with standard reflecting Gray EAs codification, uniform crossover and a stopping criterion of 200,000 maximum fitness evaluations per subpopulation (on average one fitness function takes 1.08×10^{-4} s. on an Intel Core i7-3770-3.40 GHz). The standard dynamic territory and the diversity enhanced dynamic territory have been tested with epoch values of V1, V2, V4, V8, V16, V24 and V32 (Vk means exchange of Nash territory DD every k generations). Figures 4, 5, 6, 7 and 8 show the average number of fitness evaluations (x-axis) and the standard deviation (square root of variance) of fitness evaluations (y-axis) required to achieve the optimum design solution after all the 30 independent executions (results are shown in terms of per subpopulation values). All Nash EAs achieve the optimum design after 30 runs.

In addition, a statistical analysis for evolutionary computation results will be held (see e.g. [15]). Table 1 shows results of a first test for normality (Lilliefors test), where the null hypothesis corresponds to a normal distribution assumption—if $h = 1$, then the null hypothesis is rejected with a statistical significance level α of 0.05-. The probability value, or p value is the probability of obtaining a statistic as different or more different from the parameter specified in the null hypothesis as the statistic computed from the data. p values of the significance test are in Table 1 (if $p \leq 0.05$, then H_0 is rejected and $h = 1$; if $p > 0.05$, then H_0 is not rejected and $h = 0$).

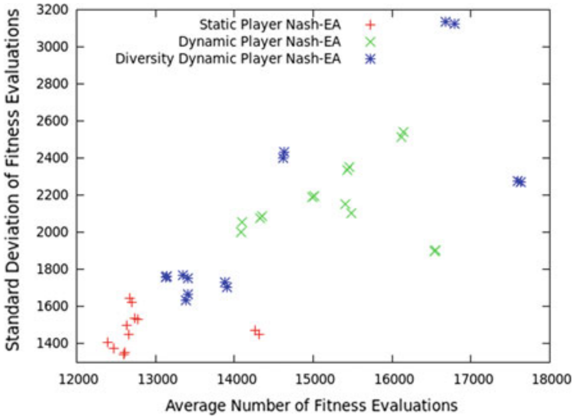


Fig. 5 Nash EAs results—fitness evaluations required to obtain the optimum design out of 30 independent runs

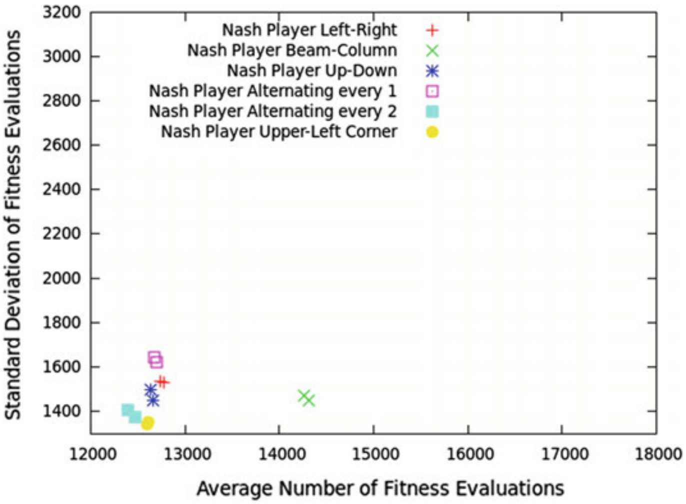


Fig. 6 Nash EAs Results—static player territory—fitness evaluations required to obtain the optimum design out of 30 independent runs

As there are samples where the null hypothesis is rejected, a Gaussian distribution of data cannot be assumed, and therefore a non-parametric statistics (also known as order or ordinal or rank statistics) will be considered, using ranks instead of actual values to perform statistics. Non-parametric multi-level analysis (when comparing multiple samples instead of just two), particularly, Kruskal-Wallis tests—based on mean-rank analysis- are taken in the following Sects. 5.1–5.5, to compare results. As 2 players are used, 2 populations are mentioned in every case. When performing statistical analysis, if the samples do not come from the same distribution (null

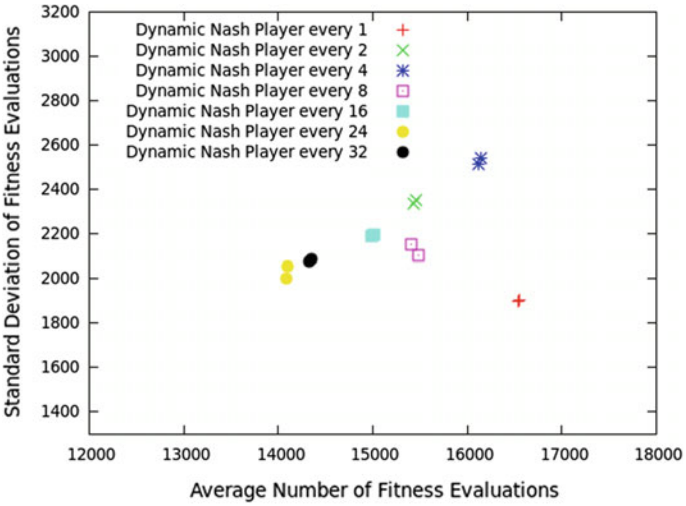


Fig. 7 Nash EAs results—'a priori' dynamic player territory—fitness evaluations required to obtain the optimum design out of 30 independent runs

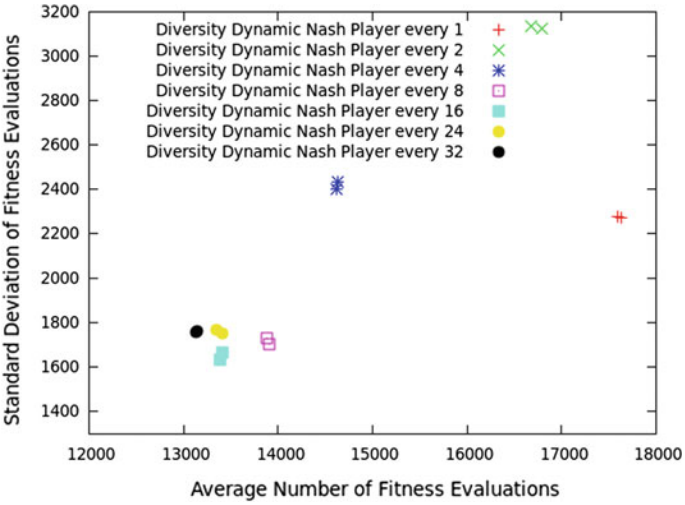


Fig. 8 Nash EAs results—proposed diversity dynamic player territory—fitness evaluations required to obtain the optimum design out of 30 independent runs

hypothesis is rejected) it is desirable because that means that there are samples (methods, here Nash EAs partitioning types) that perform better than others. Result of the Kruskal-Wallis test is shown by the Q value (if lower than 0.05 when statistical significance level $\alpha = 0.05$, then the null hypothesis is rejected) and box-plot figures showing quartile based representation of the statistics data.

Table 1 Results of 30 independent runs. Lilliefors test; null hypothesis H_0: data come from a normal distribution; statistical significance level $\alpha = 0.05$

Territory type	Population 1		Population 2	
	h	p	h	p
Nash altern. every 1, T1	0	0.219840	0	0.342045
Nash altern. every 2, T2	0	0.127510	0	0.500000
Nash up-down, T3	1	0.040685	0	0.161271
Nash upper-left corner, T4	0	0.134268	1	0.019577
Nash left-right, T5	0	0.156495	1	0.006724
Nash beam-column, T6	0	0.082490	1	0.047410
Nash dynamic every 1	0	0.500000	0	0.500000
Nash dynamic every 2	0	0.117470	0	0.097099
Nash dynamic every 4	1	0.005590	1	0.007075
Nash dynamic every 8	0	0.429216	0	0.174908
Nash dynamic every 16	1	0.032835	1	0.024703
Nash dynamic every 24	1	0.021349	1	0.028928
Nash dynamic every 32	0	0.147423	0	0.179531
Nash divers. dynamic ev.1	1	0.018510	0	0.092044
Nash divers. dynamic ev.2	0	0.500000	0	0.500000
Nash divers. dynamic ev.4	0	0.085307	0	0.070318
Nash divers. dynamic ev.8	1	0.045142	0	0.062507
Nash divers. dynamic ev.16	0	0.180167	0	0.172296
Nash divers. dynamic ev.24	0	0.089251	0	0.169355
Nash divers. dynamic ev.32	0	0.332962	0	0.393541
Panmictic population	0	0.334726	–	–

5.1 *Comparing Static Nash Domain Decomposition*

The static Nash DDs are compared in this section. This results are shown in Figs. 9 and 10. The null hypothesis (all samples correspond to the same distribution) is rejected ($\alpha = 0.05$) with a Q value $= 1.06 \times 10^{-7}$ and after a multicomparison test, it is statistically significant that samples corresponding to executions of the Nash Beam-Column territory are worse than the other static DD samples. Therefore, it is shown here that the DD can have a significative impact in the algorithm results. To guess which DD could be worse or better 'a priori' is not an easy exercise, and it could depend on the physics and type of the handled problem, even the particular test case. Here, the obvious difference of this beam-column DD with respect of the other DDs is its different non-homogeneous territory size.

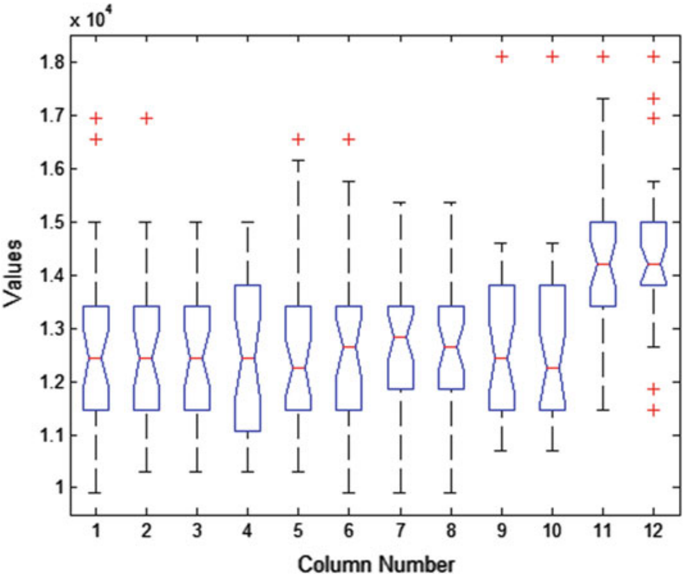


Fig. 9 Static Nash. Kruskal Wallis test results. (Order is 1–2: Nash Alt. every 1; 3–4: Nash Alt. every 2; 5–6: Nash up-down; 7–8: Nash upper-left corner; 9–10: Nash left-right; 11–12: Nash beam-column). Q value = 1.06×10^{-7}

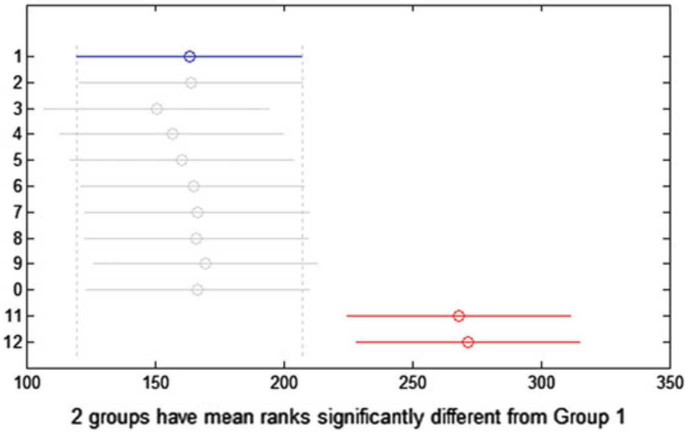


Fig. 10 Static Nash. Multicomparison test. Samples from Nash beam-column territory are worse

5.2 Comparing Dynamic Nash Domain Decomposition

The standard dynamic Nash DD is compared in this section. Each time the DD is changed, a random DD is chosen among the values tested in the static Nash DD section (from T1 to T6, as in Figs. 2, 3, 4, 5, 6 and 7). The epoch parameter has been

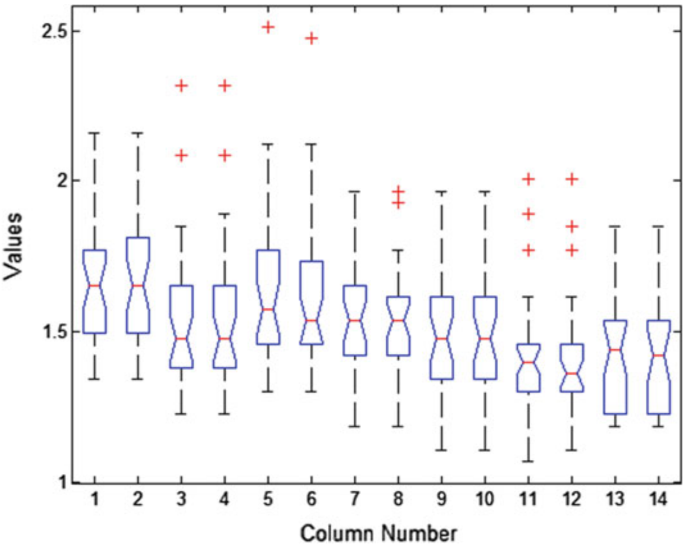


Fig. 11 Dynamic Nash. Kruskal Wallis test results. Epoch values, order is 1–2: 1; 3–4: 2; 5–6: 4; 7–8: 8; 9–10: 16; 11–12: 24; 13–14: 32). Q value = 2.61 × 10^{-8}

tested with values 1, 2, 4, 8, 16, 24 and 32 generations. These results are shown in Figs. 11 and 12. The null hypothesis (all samples correspond to the same distribution) is rejected ($\alpha = 0.05$) with a Q value = 2.61 × 10^{-8} and after a multicomparison test, it is statistically significant that the samples with the lower mean ranks: executions of the epoch values: 24 and 32 have better mean ranks than those from value 1. It is shown here that this parameter can have a significative impact in the algorithm results; it seems that a reasonable number of generations are required to let the algorithm exploit the genetic information towards the optimum. It is also remarkable that this dynamic Nash DD maintains the high robustness of the procedure, because all the independent executions were able to achieve the optimum.

5.3 Comparing Diversity Dynamic Nash Domain Decomposition

The diversity enhanced dynamic Nash DD is compared in this section. As in Sect. 5.2, epoch has been tested with values 1, 2, 4, 8, 16, 24 and 32 generations. These results are shown in Figs. 13 and 14. The null hypothesis (all samples correspond to the same distribution) is rejected ($\alpha = 0.05$) with a Q value = 0 and after a multicomparison test, it is statistically significant that samples corresponding to executions of epoch: 8, 16, 24 and 32 are better than those of values 1 and 2. Therefore, this parameter has a significative impact in the algorithm results; it seems also that a reasonable number of

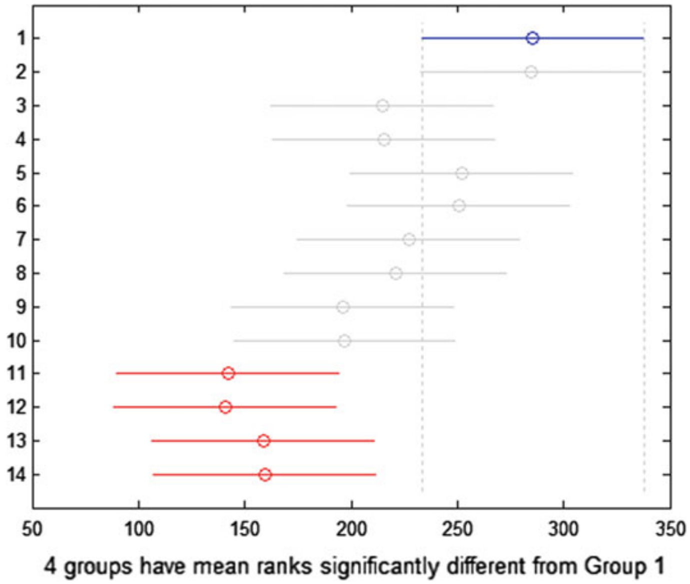


Fig. 12 Dynamic Nash. Multicomparison test. Samples from epoch values 24 and 32 have better mean ranks than those from value 1

generations are required to let the algorithm exploit the genetic information towards the optimum. As in Sect. 5.2, this diversity enhanced dynamic Nash DD maintains the high robustness of the procedure, as all the independent executions were able to achieve the optimum.

5.4 *Comparing Dynamic Nash Domain Decomposition with Diversity Dynamic Nash Domain Decomposition*

The best values of epoch of the dynamic Nash DD and the diversity dynamic Nash DD are compared in this section. Particularly, the epoch parameter has been compared with values 16, 24 and 32 generations. These results are shown in Figs. 15 and 16. Null hypothesis (all samples correspond to the same distribution) is rejected ($\alpha = 0.05$) with Q value $= 3.0 \times 10^{-4}$ and after a multicomparison test, it is statistically significant that samples corresponding to executions of the epoch value 32 of the diversity dynamic Nash DD are better than those of values 16 of the standard dynamic Nash DD. In addition, in every case, mean ranks of diversity dynamic Nash EAs are lower than those of the dynamic Nash EA.

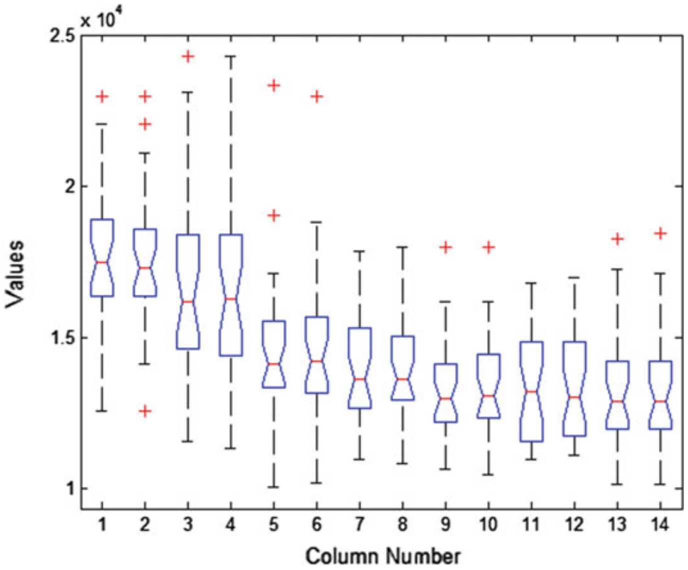


Fig. 13 Diversity dynamic Nash. Kruskal Wallis test results. Epoch values, order is 1–2: 1; 3–4: 2; 5–6: 4; 7–8: 8; 9–10: 16; 11–12: 24; 13–14: 32). Q value = 0

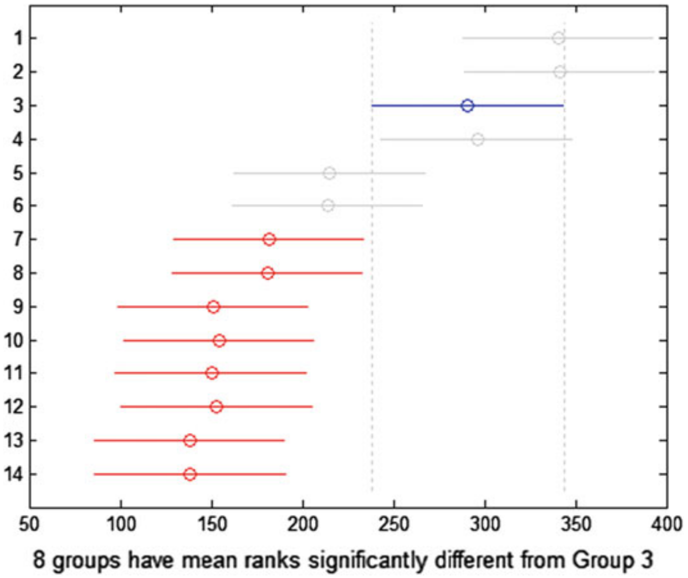


Fig. 14 Diversity dynamic Nash. Multicomparison test. Samples from epoch values 8, 16, 24 and 32 have better mean ranks than those from values 1 and 2

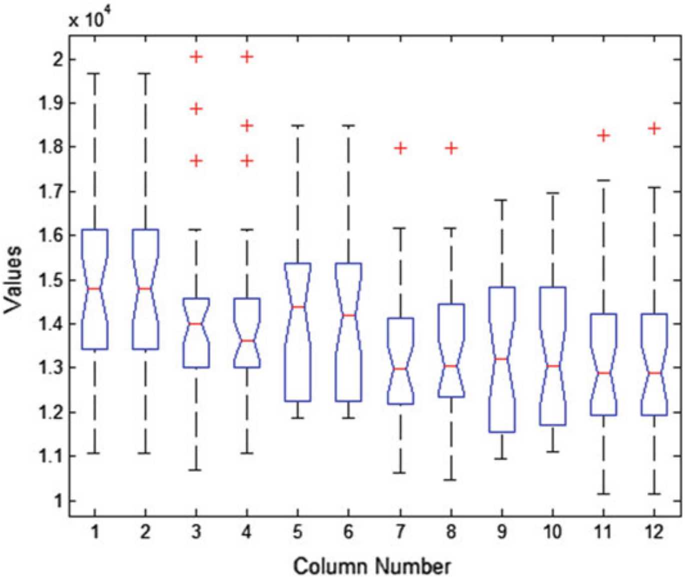


Fig. 15 Dynamic Nash versus diversity dynamic Nash. Kruskal Wallis test results. Dynamic Nash, epoch values, order is 1–2: 16; 3–4: 24; 5–6: 32; diversity dynamic Nash, order is 7–8: 16; 9–10: 24; 11–12: 32. Q value = 3.0×10^{-4}

5.5 Final Comparisons

The convergence curves of the average and best values of the fitness function throughout the optimization process are compared in Figs. 17 and 18, respectively (static Nash Beam-Column, dynamic Nash with epoch 24, and diversity dynamic Nash with epoch 32, and panmictic population).

A Kruskall Wallis test where the best values of the Static Nash DD (discarding the beam-column DD) and the diversity dynamic Nash (epoch = 32) is performed (see Fig. 19). With a Q value of 0.8748, the null hypothesis (all samples correspond to the same distribution) is not rejected. Therefore, although the diversity dynamic Nash has the highest mean rank among all this set, from a statistical significance point of view we cannot reject the hypothesis that all these samples came from the same distribution. From the engineering optimization point of view, nevertheless, a crucial difference lays in the fact that when using the diversity enhanced dynamic Nash approach, 'a priori' no decision has to be made about assuming a particular DD.

Finally, these sample set is compared with the standard panmictic EA (Fig. 20). With a Q value of 1.077×10^{-13}, the null hypothesis (all samples correspond to the same distribution) is rejected. The multi comparison test, confirms that is this panmictic EA the one which shows a significantly worst mean rank (Fig. 21).

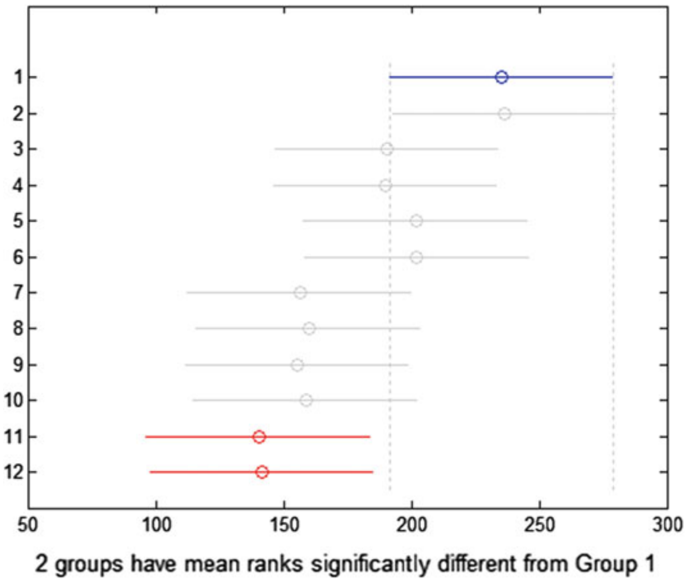


Fig. 16 Dynamic Nash versus diversity dynamic Nash. Multicomparison test. Samples from epoch values 32 of diversity dynamic Nash have better mean ranks than those from values 16 of dynamic Nash

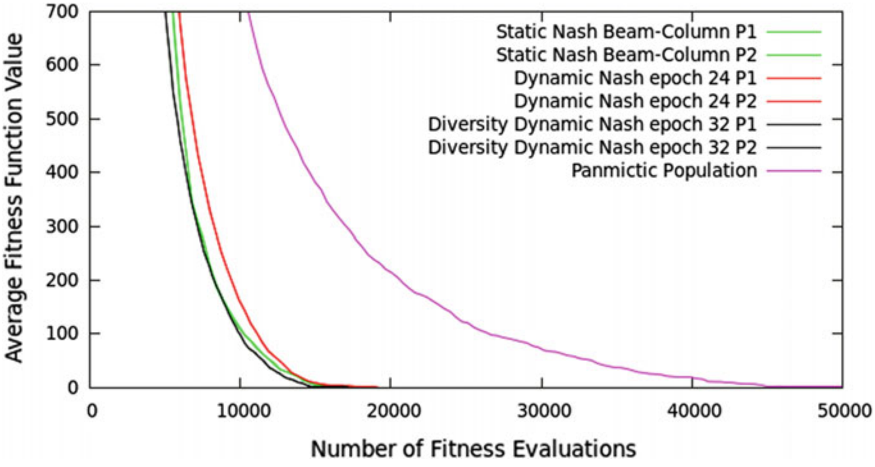


Fig. 17 Average convergence of several territory DDs

5.6 Overall Discussion

The numerical results of a reconstruction problem in structural engineering obtained in this paper, suggest the following general remarks:

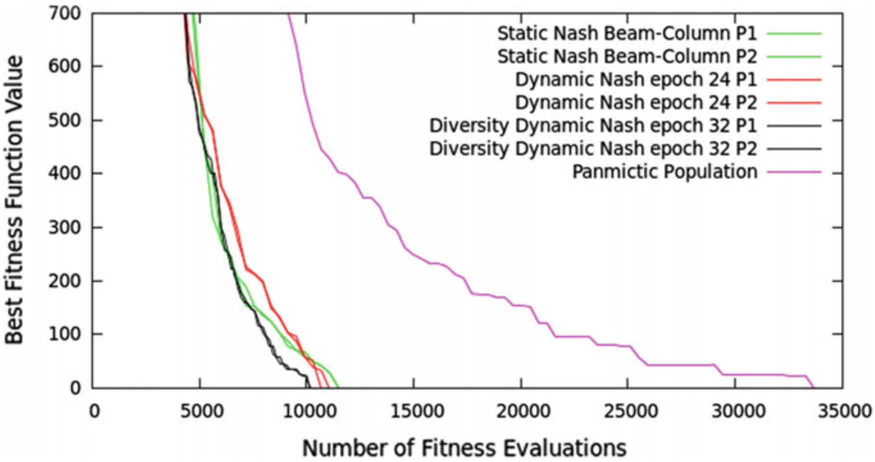


Fig. 18 Best convergence of several territory DDs

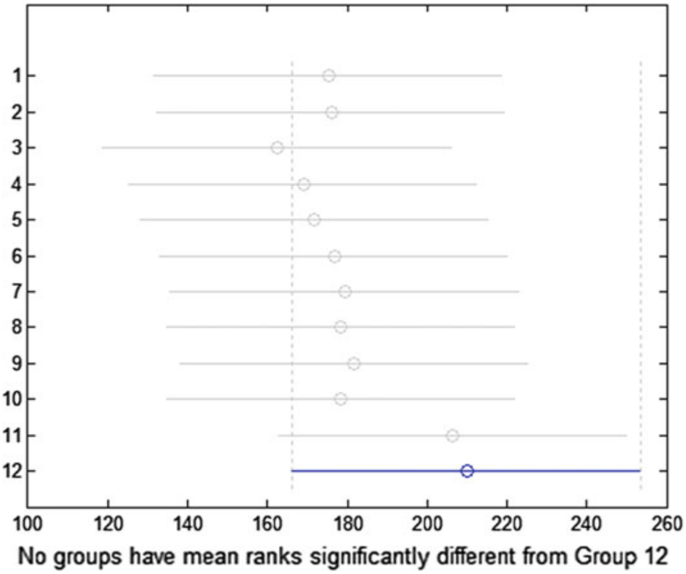


Fig. 19 Best static Nash DD and diversity dynamic Nash (epoch = 32). Order is 1–2: Nash Alt. every 1; 3–4: Nash Alt. every 2; 5–6: Nash up-down; 7–8: Nash upper-left corner; 9–10: Nash left-right; 11–12: diversity dynamic Nash; Kruskal Wallis test results. Q value = 0.8748; Multicomparison test

(a) It has been evidenced that the selection of the DD in the Nash EAs can affect the convergence speed of the algorithm;

(b) Using a diversity dynamic Nash EA diminishes the possibility of choosing 'a priori' a not optimal domain decomposition. With an increase of the epoch value

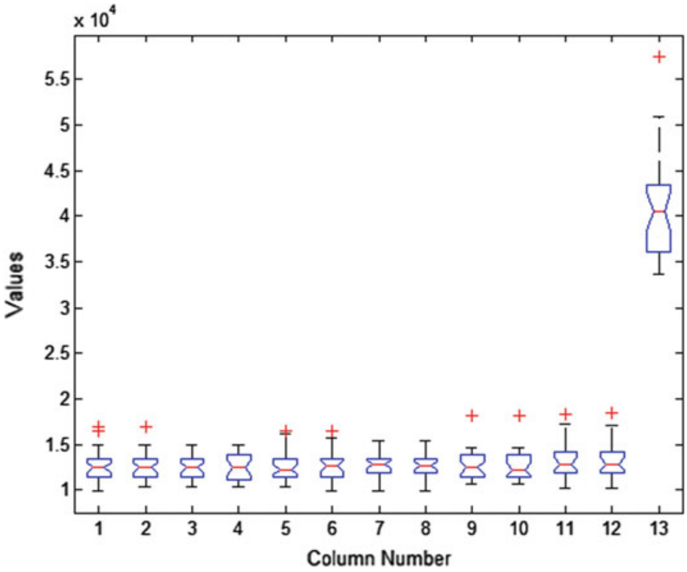


Fig. 20 Best static Nash DD versus diversity dynamic Nash (epoch = 32) and Panmictic EAs. Kruskal Wallis test results. Q value = 1.077×10^{-13}

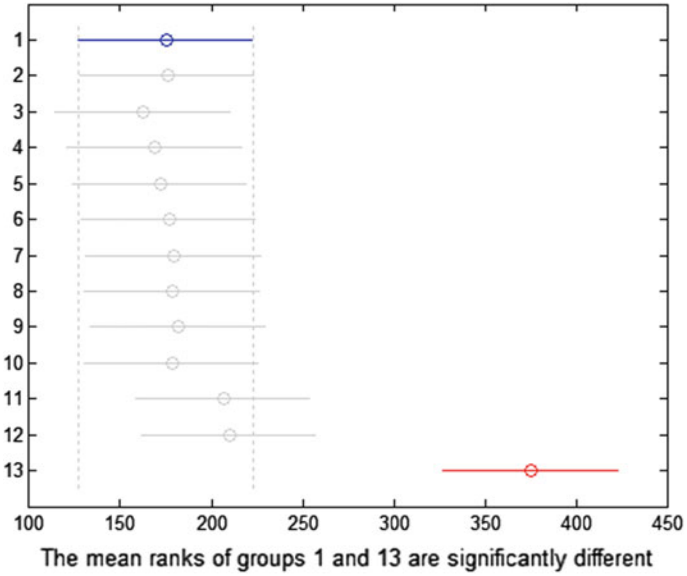


Fig. 21 Best static Nash DD, diversity dynamic Nash (epoch = 32) and Panmictic EAs. Multicomparison test. Samples from panmictic EA have worse mean ranks

Vk, a diversity dynamic Nash EA is able to evolve at least as quickly as the ad hoc static DDs from a statistical significant point of view;

(c) The diversity enhanced dynamic Nash EA proposed here, obtain the overall lowest mean ranks when compared to the standard dynamic Nash EA;

(d) At least considering the bar frame reconstruction problem, Nash EA has shown a high robustness, being able to achieve the best solution in all executions (static and dynamic cases);

(e) Again, the advantage of using a Nash EA is significant, measured in terms of the number of fitness evaluations required to find the best design, when compared with the panmictic EA.

6 Conclusions

A new dynamic territory approach of a Nash EA has been proposed in this research, and tested with a bar frame reconstruction problem in structural engineering. The comparison with the standard static and dynamic Nash DD has shown that this diversity enhanced dynamic Nash EA is a promising game-theory based tool in engineering optimization, avoiding the 'a priori' choice of a territory split.

As future work, the parallelization of this new approach, as well as its application in other structural engineering inverse and optimization problems—such as the fully stressed design problem- will be investigated to test its potentiality in performances and quality design.

References

1. Periaux, J., Gonzalez, F., Lee, D.C.: Evolutionary optimization and game strategies for advanced multi-disciplinary design: applications to aeronautics and UAV design. In: Intelligent Systems, Control and Automation: Science and Engineering Series, vol. 75. Springer, (2015)
2. Greiner, D., Emperador, J.M., Galvan, B., Winter, G., Periaux, J.: Optimum structural design using bio-inspired search methods: a survey and applications. In: Becerra, V., Vasile, M. (eds.) Computational Intelligence in Aerospace Sciences. American Institute of Aeronautics and Astronautics AIAA, pp. 373–414 (2014)
3. Sefrioui, M., Periaux, J.: Nash genetic algorithms: examples and applications. In: Proceedings of the 2000 Congress on Evolutionary Computation CEC-2000, pp. 509–516. IEEE Press, La Jolla Marriott Hotel La Jolla, California, USA, ISBN 0-7803-6375-2 (2000)
4. Lee, D.S., Périaux, J., González, L.F., Srinivas, K., Oñate, E.: Active flow control bump design using hybrid Nash-game coupled to evolutionary algorithms. In Pereira, J.C.F., Sequeira, A., Pereira, J.M.C. (eds.) Proceedings of V European Conference on Computational Fluid Dynamics ECCOMAS CFD 2010, ECCOMAS CFD, Lisbon, Portugal, pp. 1–14 (2010)
5. Leskinen, J., Périaux, J.: Distributed evolutionary optimization using Nash games and GPUs—applications to CFD design problems. Comput. Fluids **80**, 190–201 (2012)
6. Leskinen, J., Wang, H., Périaux, J.: Increasing parallelism of evolutionary algorithms by Nash games in design inverse flow problems. Eng. Comput. **30**(4), 581–600 (2013)

7. Nash, J.F.: Equilibrium points in N-person games. Proc. Natl. Acad. Sci. **36**, 46–49 (1950)
8. Nash, J.F.: Non-cooperative games. Ann. Math. **54**(2), 286–295 (1951)
9. Greiner, D., Périaux, J., Emperador, J.M., Galván, B., Winter, G.: A hybrid Nash genetic algorithm for reconstruction inverse problems in structural engineering. Report of the Department of Mathematical Information Technology, Series B Scientific Computing, Nº B 5/2013, University of Jyväskylä, Finland (2013)
10. Greiner, D., Periaux, J., Emperador, J.M., Galván, B., Winter, G.: A study of Nash-evolutionary algorithms for reconstruction inverse problems in structural engineering. In: Greiner D et al. (eds.) Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Scinces, Computational Methods in Applied Sciences, vol. 36, pp. 321–333. Springer (2015)
11. Periaux, J., Greiner, D.: Efficient parallel Nash—genetic algorithm for solving inverse problems in structural engineering. In: Neittaanmäki, P., et al. (eds.) Mathematical Modeling and Optimization of Complex Structures, Computational Methods in Applied Sciences, vol. 40, pp. 205–228. Springer (2016)
12. Greiner, D., Périaux, J., Emperador, J.M., Galván, B., Winter, G.: Game theory based evolutionary algorithms: a review with Nash applications in structural engineering optimization problems. Arch. Comput. Methods Eng. **24**(4), 703–750 (2017). https://doi.org/10.1007/s11831-016-9187-y
13. Tang, Z.L., Desideri, J.A., Periaux, J.: Distributed optimization using virtual and real game strategies for aerodynamic design. INRIA Research Report RR-4543, inria-00072045 (2002)
14. Greiner, D., Winter, G., Emperador, J.M.: Single and multi-objective frame optimization by evolutionary algorithms and the auto-adaptive rebirth operator. Comput. Methods Appl. Mech. Eng. **37**(35), 3711–3743 (2004)
15. Wineberg, M.: Statistical analysis for evolutionary computation: an introduction. Tutorial in GECCO 2012, Philadelphia, USA (2012)

Interactive Inverse Modeling Based Multiobjective Evolutionary Algorithm

Karthik Sindhya and Jussi Hakanen

Abstract An interactive version of the inverse modeling based multiobjective evolutionary algorithm is presented. Instead of generating a representation of the whole Pareto optimal front, the algorithm aims at producing solutions in the regions where the decision maker is interested in. This is facilitated through an interactive solution process where the decision maker iteratively evaluates a set of solutions shown to her/him and the preference information obtained is used to adapt the search process of the algorithm.

1 Introduction

Recently, preference-based multiobjective evolutionary approaches have become an important research topic among evolutionary multiobjective optimization (EMO, see e.g. [4]). Instead of approximating the whole Pareto front, that is, the set of all Pareto optimal solutions in the objective space, it is typically of interest to approximate only some region of the Pareto front that the decision maker (DM) is interested in (see e.g. [1]). In order to enable the DM to learn about the behaviour of the problem considered and about his/her own preferences, (s)he should be able to change her/his preferences based on the solutions obtained during the solution process. This can be achieved by using interactive methods on top of preference-based multiobjective evolutionary approaches [8]. More information on interactive multiobjective optimization methods in general can be found in [10]. Interactive EMO algorithms are especially useful for problems having more than three objectives when the traditional EMO algorithms are no longer efficient [6].

Several interactive multiobjective evolutionary algorithms have been proposed in the literature. For an extensive review of these interactive algorithms refer [11].

K. Sindhya (✉) · J. Hakanen (✉)
Faculty of Information Technology, University of Jyvaskyla,
P.O. Box 35(agora), FI-40014 University of Jyvaskyla, Jyväskylä, Finland
e-mail: karthik.sindhya@jyu.fi

J. Hakanen
e-mail: jussi.hakanen@jyu.fi

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_18

Here we present some prominent ones. In [5], a novel interactive optimization tool called I-EMO was proposed. Here first a set of Pareto optimal solutions were generated using an EMO algorithm and then a DM was helped through a graphical user interface to find her/his preferred solution(s) among them. Thiele et al. [12] proposed a preference based evolutionary algorithm for multiobjective optimization, wherein the DM progressively provided preference information in every generation as desirable objective function values and directed the search towards only preferable regions of the Pareto optimal front. In [6], Deb et al. proposed a progressively interactive EMO algorithm called PI-EMO. In PI-EMO algorithm, the DM is asked to provide preference information after every fixed interval of generations. The preference information is provided as a ranking of selected solutions shown in the current generation, which is subsequently used to construct an utility function. This utility function is further used to direct the search towards preferable solutions of the DM.

An emerging approach in EMO is estimation of distribution algorithms which instead of traditional reproduction operators build probabilistic models estimating the distribution of promising candidate solutions, New solutions are then produced by sampling the obtained distributions. A recent example of such an algorithm is the inverse modeling based multiobjective evolutionary algorithm (IM-MOEA) [3] which builds inverse Gaussian process based models from the objective space to the decision space in order to approximate the distribution of promising solutions. This enables generating new candidate solutions via sampling directly in the objective space.

This paper introduces an interactive IM-MOEA algorithm where the DM is able to guide the search towards her/his preferred regions of the Pareto front. To facilitate this, a decision making module is added to the IM-MOEA algorithm which takes into account the preferences of the DM for a prefixed number of interactions during the search. Through the decision making module, the DM is shown a number of solution candidates among which (s)he needs to select one, select two (that is, define a region in the Pareto front) or disregard all the solutions. Based on her/his actions, the search mechanism of the algorithm is adjusted accordingly.

The rest of the paper is divided into two parts. First, the proposed interactive IM-MOEA algorithm is described along with the necessary description of the original IM-MOEA algorithm. Then, the paper ends with a brief description of the numerical experiments and some concluding remarks.

2 Interactive IM-MOEA Algorithm

2.1 IM-MOEA

The main idea of the IM-MOEA algorithm presented in [3] is to enable sampling in the objective space instead of the decision space. This is implemented by introducing Gaussian process based inverse mapping from the objective functions to the decision variables. When sampling in the objective space, the corresponding decision variable values are obtained by using the approximated inverse mapping. To ease the computation, the multivariate inverse model is decomposed into several univariate inverse models. The number of inverse models is reduced by using a random grouping strategy where some of the decision variables are assigned to a specific objective function. In practice, inverse modelling is performed inside sub populations created from the whole population with the help of fixed number of uniformly distributed reference vectors in the objective space. More information can be found in [3].

2.2 Incorporation of Decision Maker Preferences Through Adapting Reference Vectors

The original IM-MOEA method [3] uses K uniformly distributed reference vectors to divide the population into sub populations. Each individual of the population is assigned to the closest reference vector and the solutions that are assigned to the same reference vector form a sub population resulting in K sub populations. In order to improve the performance for problems with discontinuous Pareto fronts, an adaptive reference vector scheme was proposed in [2]. In each generation, one of the reference vectors will be replaced by another randomly generated vector. In the beginning of the solution process, that is the exploration phase, the vector which has the most solutions assigned will be replaced. As opposed to that in the exploitation phase, the vector with the lowest number of solutions assigned will be replaced. More details of the adaptive version of the IM-MOEA algorithm can be found in [2].

The main idea of the interactive IM-MOEA algorithm presented here is to take advantage of the adaptive reference vector generation suggested in [2] and generate solutions that are desirable to the DM. Here, instead of replacing the reference vectors with randomly generated ones, such reference vectors will be added that can yield solutions reflecting the preferences expressed by the DM. The proposed interactive IM-MOEA algorithm consists of a decision making module attached to the original IM-MOEA algorithm. The interactive algorithm follows the general algorithm of IM-MOEA (see [3]) except the interaction step (step 4 shown in the Algorithm 1). The interaction step is executed not in each generation but only when interaction with the DM is needed. This can be determined, for example, based on how many times the DM wants to interact during the solution process (value of this parameter can come from the decision). In that case, generations when the interaction is needed can be determined based on the number of interactions and the maximum number of function evaluations. Additionally, the crowding distance measure used for selection in [3] is replaced by a distance based selection in order to favour more solutions preferred by the DM. The details of the decision making module will be described in the next subsection.

Algorithm 1 The pseudo code of interactive IM-MOEA

```
initialization
while termination condition is not satisfied do
    partition of the combined population
    if interaction required then decision making module
    non-dominated sorting and selection
    for $k = 1$ to $K$ do
        inverse modeling
        reproduction
    end for
    update the combined population
end while
```

2.3 Decision Making Module

In this subsection we extend the IM-MOEA algorithm with the addition of a decision making module, which can be conveniently used to handle DM's preferences. As mentioned earlier, the decision making module presented in Algorithm 2 is based on the idea of adaptive reference vectors presented by Cheng et al. [2]. In addition to the parameters inherited from the original IM-MOEA algorithm, interactive IM-MOEA has two additional parameters, i.e. the number of interactions the DM wishes to have with the interactive IM-MOEA algorithm and the number of solutions (s)he wishes to see in each interaction. It must be noted that the DM has the freedom to change these two parameters during any interaction.

The decision making module is classified into four steps, i.e. classification of reference vectors, generate solutions for the DM, preference information from the DM and adjustment of reference vectors. In the first step, the current set of reference vectors are classified into active and inactive reference sets. The reference vectors with associated individuals form an active reference set and reference vectors with no associated individuals form an inactive reference set. In addition, the reference vector having the most solutions associated with is identified as the lead reference vector $\mathbf{v}^{\mathbf{lead}}$. In the second step, the active reference set is clustered into N_{DM} clusters and one representative reference vector from each cluster is chosen. Next for each N_{DM} reference vectors, an individual that makes minimum angle with it is chosen. Thus N_{DM} individuals are shown as candidate solutions to the DM, based on which the DM provides her/his new preference information. The value for the parameter N_{DM} can be asked from the DM (that is, how many different solutions the DM wants to see at a time) or, otherwise, a default value can be used.

In step 3, the decision maker can either choose to explore further without choosing any solution, choose one solution or choose a preferred region identified by two different solutions. Based on what the decision maker chooses, different actions are taken in step 4. If the DM disregards all solutions shown to her/him in step 3, the lead reference vector $\mathbf{v}^{\mathbf{lead}}$ is replaced by a randomly generated reference vector

Algorithm 2 Decision making module

Step 1: *Classification of reference vectors*:

- For each reference vector $\mathbf{v_i}$, $i = 1, \ldots, K$, calculate the number of individuals ($nsol_i$) of the population associated with it.
- The set of all reference vectors with no associated individuals form an inactive reference set IR and the rest of the reference vectors form an active reference set AR.
- From the set AR, the reference vector with maximum number of associated individuals is termed as lead reference vector $\mathbf{v^{lead}}$.

Step 2: *Generate candidate solutions for the DM*

- Cluster the elements of AR into N_{DM} clusters (If $|AR| < N_{DM}$, then $N_{DM} = |AR|$). From every cluster choose one reference vector as a representative index of the corresponding cluster. Subsequently, for every reference vector among all associated solutions, a solution $\mathbf{z_i}$ is chosen whose position in the objective space has minimum angle with the reference vector. Thus $\mathbf{z_i}$, $i = 1, \ldots, N_{DM}$ candidate solutions are shown to the DM.

Step 3: *Preference information from the DM*. DM follows one of the following three paths:

- disregards all N_{DM} solutions shown to her/him and wishes to explore more. Go to *Step 4a*.
- chooses one solution among N_{DM} solutions as his preferred solution $\mathbf{z_{pref}}$. Go to *Step 4b*.
- identifies a preferred region defined by two solutions ($\mathbf{z_{pref}}$ and $\mathbf{z_{pref+1}}$) to explore further. Go to *Step 4c*.

Step 4: *Adjustment of reference vectors*

- *Step 4a:* Replace $\mathbf{v^{lead}}$ with a new randomly generated reference vector.
- *Step 4b:* The whole set of reference vectors is updated as follows: If this is not the last interaction, the extreme reference vectors $e^i = (0, \ldots, 1, \ldots, 0)^T$, that is, unit vectors, are included. Then, the reference vector $\mathbf{v^{near}}$ having the minimum angle with the reference vector associated with $\mathbf{z_{pref}}$ ($\mathbf{v^{pref}}$) is added. Accordingly, the vector $\mathbf{v^{ref}}$ that is obtained by reflecting $\mathbf{v^{near}}$ with respect to $\mathbf{v^{pref}}$ is also added. The remaining reference vectors are uniformly distributed between $\mathbf{v^{near}}$ and $\mathbf{v^{ref}}$. The procedure is illustrated in Figure 1.
- *Step 4c:* The whole set of reference vectors is updated as follows: If this is not the last interaction, the extreme reference vectors e^i are included as in *Step 4b*. Then, the reference vectors associated with $\mathbf{z_{pref}}$ and $\mathbf{z_{pref+1}}$ are added and the rest of the reference vectors are equally distributed between these two vectors.

in *step 4a*. Alternatively, if the decision maker chooses one solution, all reference vectors other than the $\mathbf{v^{pref}}$ (i.e. the one that the preferred solution is associated with) are deleted. The entire set of reference vectors is regenerated between the reference vector $\mathbf{v^{near}}$ that is the closest with the $\mathbf{v^{pref}}$ and its reflected counterpart $\mathbf{v^{ref}}$ with respect to $\mathbf{v^{pref}}$ as shown in Fig. 1. This is done in order to bias the search towards region of interest to the decision maker. Finally, if the decision maker prefers two solutions (i.e. is interested in the region between them), the set of reference vectors is generated between the two reference vectors that the preferred solutions indicated by the DM are associated with. Additionally, at each interaction (except the final), the extreme reference vectors are included to be able to allow changes in preference information and maintain diversity in the population. They are not included in the final interaction in order to guarantee convergence according to preferences specified

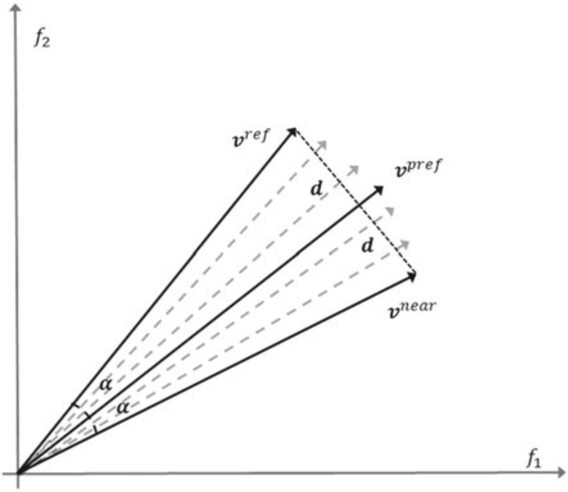


Fig. 1 Illustration of reference vector generation in step 4b of the Algorithm 2

by the DM. Note that the number of the reference vectors remains fixed throughout the search.

3 Numerical Experiments

The performance of the interactive IM-MOEA algorithm consisting of the proposed decision making module in conjunction with the IM-MOEA algorithm is demonstrated here by using two and three objective problems. The problems selected are bound constrained problems F2 (two objective problem, modified from the ZDT2 problem) and F4 (three objective problem, modified from the DTLZ2 problem) from [3], both having 30 decision variables. The population size used was 50 for both problems and the maximum number of function evaluations was 50000 and 100000 for F2 and F4, respectively. The number of reference vectors was set to $K = 10$ for both problems as suggested in [3]. The random group size was set little higher ($L = 10$) than proposed in [3] in order to get more accurate results. Clustering in step 2 of the decision making module is performed by using K-means clustering [7]. The interaction between the DM and the interactive IM-MOEA algorithm for both problems F2 and F4 are summarized in Table 1. In every interaction the solutions chosen by the DM are marked in bold font.

Table 1 Candidate solutions shown to DM during four interactions for problems F2 and F4

Interaction	Solution number	Candidate solutions (problem F2) (f_1, f_2)	Candidate solutions (problem F4) (f_1, f_2, f_3)
1	1	(0.000, 1.000)	(0.146, 41.3, 0.012)
	2	(0.122, 0.988)	(0.000, 6.81, 27.8)
	3	(0.267, 0.932)	(73.9, 0.022, 0.002)
	4	**(0.415, 0.830)**	(4.95, 6.27, 9.39)
	5		(26.8, 0.000 33.2)
2	1	**(0.507, 0.747)**	**(0.553, 0.602, 1.01)**
	2	**(0.475, 0.788)**	(0.000, 0.000, 1.37)
	3	(0.354, 0.878)	(0.001, 12.7, 0.000)
	4	(0.312, 0.907)	(19.0, 1.56, 0.000)
	5	(0.000, 1.000)	(5.33, 1.04, 5.90)
3	1	**(0.000, 1.000)**	(0.000, 0.000, 1.03)
	2	(0.482, 0.770)	(0.297, 0.747, 0.718)
	3	**(0.471, 0.780)**	**(0.015, 1.54, 0.212)**
	4	(0.502, 0.751)	**(0.654, 0.353, 0.750)**
	5	(0.492, 0.761)	(4.79, 0.000, 1.05)
4	1	(0.000, 1.000)	(0.413, 0.889, 0.282)
	2	**(0.361, 0.873)**	(1.67, 0.000, 0.101)
	3	**(0.224, 0.953)**	(0.002, 1.03, 0.000)
	4	(0.149, 0.980)	(0.000, 0.000, 1.02)
	5	(0.471, 0.780)	**(0.649, 0.603, 0.569)**
5	1		(0.000, 0.000, 1.06)
	2		**(0.587, 0.693, 0.495)**
	3		**(0.656, 0.558, 0.553)**
	4		(1.02, 0.000, 0.002)
	5		(0.049, 1.00, 0.008)

3.1 Example with a Biobjective Problem

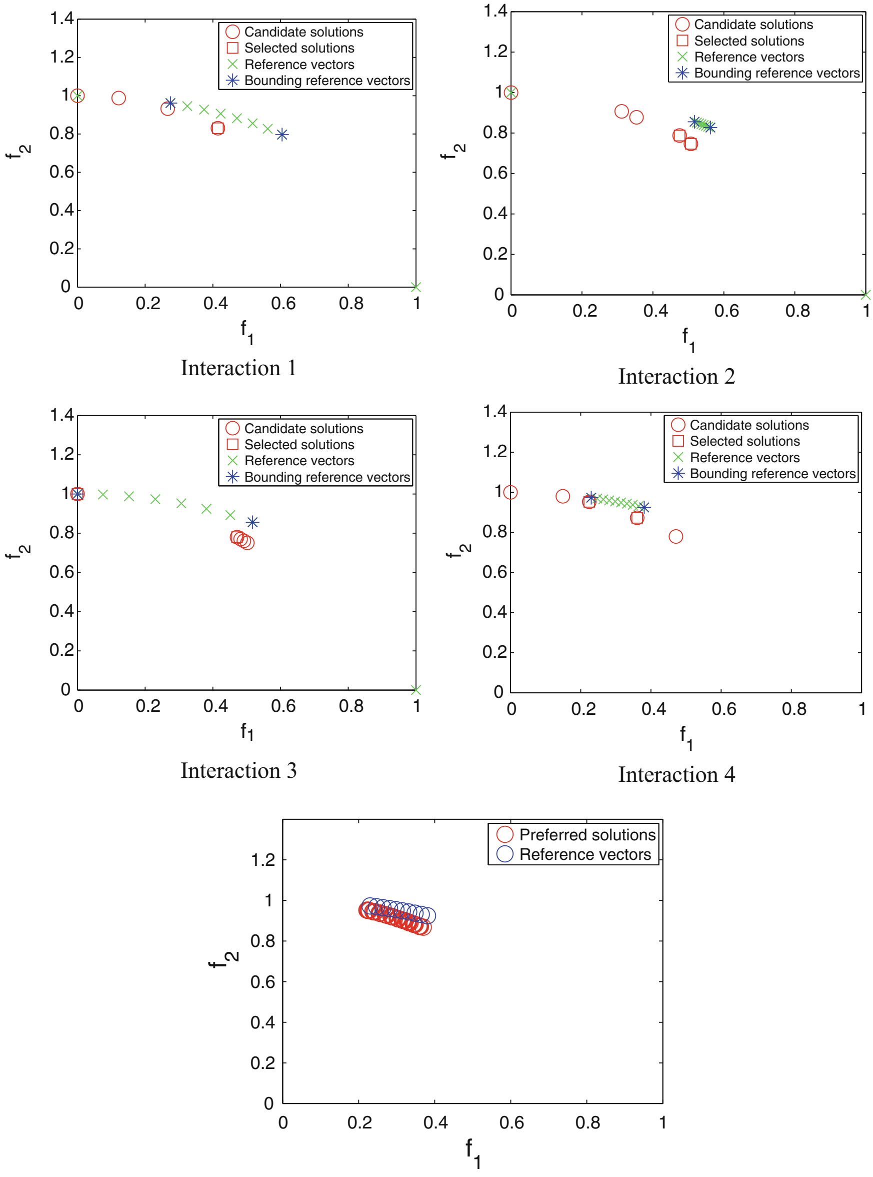
In the first example, problem F2 with a nonconvex Pareto front will be used. The DM wants to interact four times ($N_{ia} = 4$) and he wants to see five solutions at each interaction ($N_{DM} = 5$). Initially, his preferences are to get as close to $f_1 = 0.5$ as possible. In the first interaction, the candidate solutions shown to him are presented in Table 1. Note that there are only four candidate solutions since only four of the reference vectors are active (i.e., have associated solutions) at this phase.

The range of the solutions shown for f_1 is [0.000, 0.415] and, according to his preferences, he selects candidate solution number 4 as the preferred one. Figure 2 shows the candidate solutions, a solution selected by the DM and the updated set of the reference vectors where the bounding reference vectors are highlighted. Note that it is only possible to show this kind of visualization to the DM when the number of objective functions is two or three. For problems with more objectives, other types of visualization techniques need to be used (see e.g. [9]). In addition, reference vectors are not meant to be shown to the DM but are in this paper included for illustrative purposes. Based on the preferences, the candidate solutions shown to the DM for the second interaction are presented in Table 1. This time, there are five candidate solutions since at least five reference vectors have associated solutions.

Candidate solution 1 has a value very close to the initial preferences of the DM ($f_1 = 0.507$). Next, he wants to see more solutions between candidate solutions 1 and 2 and selects them as preferred solutions. Figure 2 shows the candidate solutions, solutions selected by the DM and the updated set of the reference vectors where the bounding reference vectors are highlighted. Note that now the spread of the reference vectors is smaller when the DM wants to zoom in to a specific region in the Pareto front. Based on those preferences, the next five candidate solutions are shown in Table 1.

Now all the candidate solutions except number 1 are very close to the initial preferences of the DM. Having examined the solutions in more detail, he is not interested in them anymore but wants to improve f_1 further. To zoom out from the region of his initial preferences, he selects the candidate solutions 1 and 3 as he wants to examine the region for lower f_1 values. Accordingly, Fig. 2 shows the candidate solutions, solutions selected by the DM and the updated set of the reference vectors where the bounding reference vectors are highlighted. Note that now the spread of the reference vectors is bigger since the DM wants to zoom out. The resulting candidate solutions are now shown in Table 1.

Based on the candidate solutions shown, the DM was able to move towards a region where f_1 values are smaller than 0.5. This is now the final interaction and he wants to focus in the area between candidate solutions 2 and 3. Figure 2 shows the candidate solutions, solutions selected by the DM and the updated set of the reference vectors where the bounding reference vectors are highlighted. Note that after the last interaction, the extreme reference vectors are not anymore included since the focus is now on converging according to the last preferences. The set of final solutions obtained is shown in Fig. 2 along the final set of reference vectors. As can be seen, the solutions obey nicely to the final set of reference vectors that are based on the last preferences of the DM. In addition, the interactive IM-MOEA algorithm showed in this example that it can follow the changes in the DM's preferences.



Interaction 1

Interaction 2

Interaction 3

Interaction 4

Final solutions

Fig. 2 Interactions with the DM for problem F2

3.2 Example with a Three Objective Problem

To illustrate the behavior of the interactive IM-MOEA algorithm in three objective problems, we use the problem F4 from [3]. In this case, the DM wants to interact five times ($N_{ia} = 5$) and, again, he wants to see five solutions at each interaction ($N_{DM} = 5$).

Initially, his preferences are to get as well balanced solution between the three objectives as possible. In the first interaction, the candidate solutions shown to him are presented in Table 1. None of the candidate solutions are acceptable for the DM (they are still very far from the Pareto front) and, thus, the DM does not select any of the solutions. This means that only one of the reference vectors is updated according to step 1 in Algorithm 2.

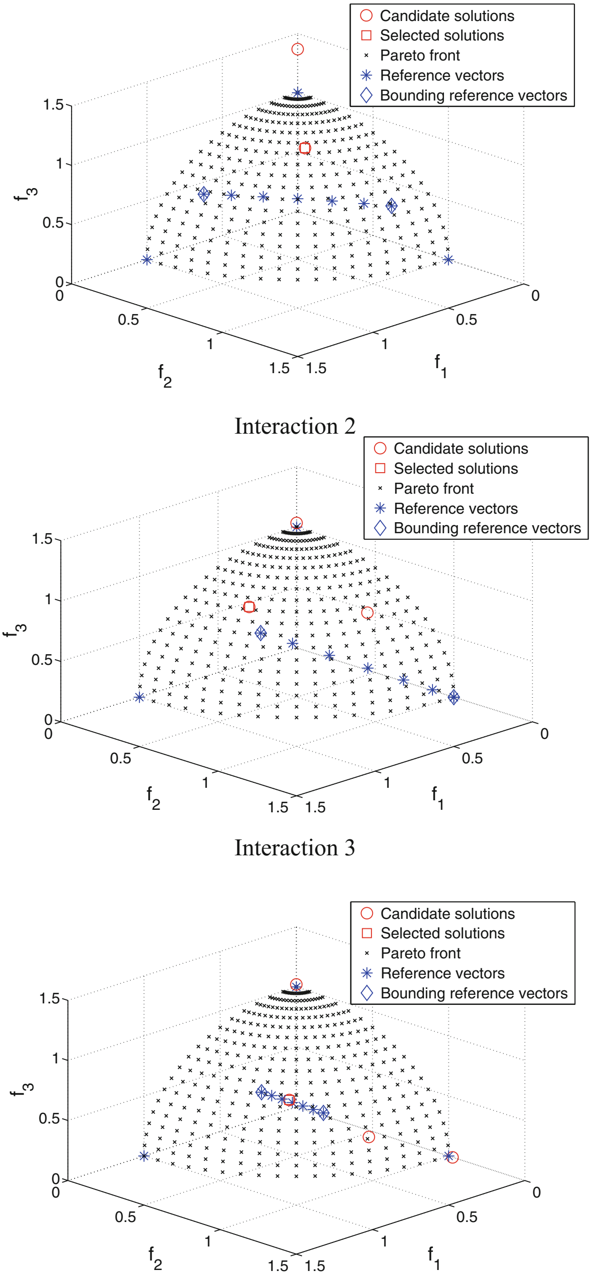
The candidate solutions for the second interaction are shown in Table 1. Although many solutions are still not acceptable, the candidate solution number 1 seems to be best of them for the DM and, therefore, he selects that one. Figures 3 and 4 show the candidate solutions, solutions selected by the DM and the updated set of the reference vectors where the bounding reference vectors are highlighted. In addition, the real Pareto front is also illustrated here to ease the visualization of the performance. For practical problems, the Pareto front is not known and, therefore, can not be visualized to the DM. The resulting candidate solutions for the third interaction are shown in Table 1.

Since most of the candidate solutions start to be in the ranges acceptable for the DM, next he wants to aim at finding a good balance between the objectives. Therefore, he selected candidate solutions 3 and 4 and the Fig. 3 shows the candidate solutions, solutions selected by the DM and the updated set of the reference vectors where the bounding reference vectors are highlighted. As can be seen in Fig. 3, he tries to aim towards the middle region of the Pareto front. The resulting candidate solutions are shown in Table 1.

Based on the candidate solutions obtained, number 5 seems to represent a good compromise between the objectives so that is selected by the DM next. Again, the resulting data is shown in Fig. 4 and the resulting candidate solutions for the last interaction can be seen in Table 1.

Among the resulting candidate solutions, three (1, 4 and 5) represent solutions where one of the objectives has not so good value and the other two (2 and 3) represent good balance between all the objectives. Therefore, the DM chose 2 and 3 (see Fig. 4 for illustration). The final solutions obtained are shown in Fig. 4 and it can be seen that they approximate balanced compromises between the objectives.

Fig. 3 Interactions 2–4 with the DM for problem F4

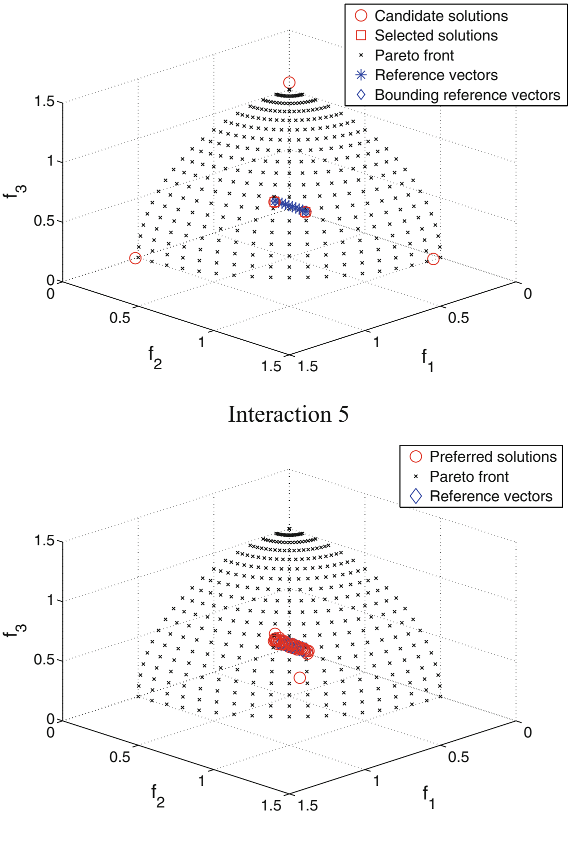


Interaction 2

Interaction 3

Interaction 4

Fig. 4 Interaction 5 with the DM and final solutions for problem F4



Interaction 5

Final solutions

4 Conclusions

In this paper we have proposed a novel interactive IM-MOEA algorithm that progressively considers the preference information of the DM and finally generates a set of solutions desirable to the DM. This algorithm strives to minimize the cognitive burden on the DM by providing the DM flexibility to decide on the number of solutions (s)he wishes to investigate and how many times (s)he wishes to interact. The algorithm presented considers a simple approach of generating reference vectors in the regions preferable to the DM, thereby generating preferred solutions. The results indicate the ease with which the DM was able to direct his search using the interactive IM-MOEA algorithm and find her/his preferred solution.

Our future research topics include enhancing reference vector adaptation for problems having more than two objectives and adding constraint handling to the proposed algorithm. In addition, testing the algorithm with real-world problems having larger number of objectives is required, for example, the General Aviation Aircraft problem [13].

Acknowledgements This research was funded by Tekes, the Finnish Funding Agency for Innovation (DeCoMo project). The authors greatly appreciate discussions with Prof. Yaochu Jin and Prof. Kaisa Miettinen related to the ideas presented in this paper as well as MTech Tinkle Chugh for his insight in the IM-MOEA algorithm.

References

1. Branke, J.: Consideration of partial user preferences in evolutionary multiobjective optimization. In: Branke, J., Deb, K., Miettinen, K., Slowinski, R. (eds.) Multiobjective Optimization: Interactive and Evolutionary Approaches, pp. 157–178. Springer (2008)
2. Cheng, R., Jin, Y., Narukawa, K.: Adaptive reference vector generation for inverse model based evolutionary multiobjective optimization with degenerate and disconnected pareto fronts. In: Gaspar-Cunha, A., Henggeler Antunes, C., Coello Coello, C. (eds.) Evolutionary Multi-Criterion Optimization, pp. 127–140. Springer (2015)
3. Cheng, R., Jin, Y., Narukawa, K., Sendhoff, B.: A multiobjective evolutionary algorithm using gaussian process based inverse modeling. IEEE Trans. Evol. Comput. **19**(6), 838–856 (2015)
4. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, Chichester (2001)
5. Deb, K., Chaudhuri, S.: I-EMO: An interactive evolutionary multi-objective optimization tool. In: Pal, S.K., Bandyopadhyay, S., Biswas, S. (eds.) Pattern Recognition and Machine Intelligence, pp. 690–695. Springer, Berlin, Heidelberg (2005)
6. Deb, K., Sinha, A., Korhonen, P., Wallenius, J.: An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. IEEE Trans. Evol. Comput. **14**(5), 723–739 (2010)
7. Jain, A., Murty, M., Flynn, P.: Data clustering: a review. ACM Comput. Surv. **31**, 264–323 (1999)
8. Jaszkiewicz, A., Branke, J.: Interactive multiobjective evolutionary algorithms. In: Branke, J., Deb, K., Miettinen, K., Slowinski, R. (eds.) Multiobjective Optimization: Interactive and Evolutionary Approaches, pp. 179–194. Springer, Berlin (2008)
9. Miettinen, K.: Survey of methods to visualize alternatives in multiple criteria decision making problems. OR Spectr. **36**(1), 3–37 (2014)
10. Miettinen, K., Ruiz, F., Wierzbicki, A.P.: Introduction to multiobjective optimization: Interactive approaches. In: Branke, J., Deb, K., Miettinen, K., Slowinski, R. (eds.) Multiobjective Optimization: Interactive and Evolutionary Approaches, pp. 27–57. Springer, Berlin (2008)
11. Purshouse, R., Deb, K., Mansor, M., Mostaghim, S., Wang, R.: A review of hybrid evolutionary multiple criteria decision making methods. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 1147–1154 (2014)
12. Thiele, L., Miettinen, K., Korhonen, P., Molina, J.: A preference-based evolutionary algorithm for multi-objective optimization. Evol. Comput. **17**(3), 411–436 (2009)
13. Woodruff, M.J., Reed, P.M., Simpson, T.W.: Many objective visual analytics: rethinking the design of complex engineered systems. Struct. Multidiscip. Optim. **48**(1), 201–219 (2013)

Multi-disciplinary Design Optimization of Air-Breathing Hypersonic Vehicle Using Pareto Games and Evolutionary Algorithms

Peng Wu, Zhili Tang and Jacques Periaux

Abstract The design integration of a supersonic combustion ramjet engine (SCRAMJET) with an airframe remains a critical task for guarantying a successful mission of trans atmospheric or hypersonic cruise vehicles. For this purpose, the operational efficiency has to be established by the effective specific impulse and the thrust to weight ratio of the accelerating vehicle. In order to analyze the foregoing problems, a design methodology based on Evolutionary Algorithms (EAs) and Game Strategies (GS) is developed. In this study, Evolutionary Algorithms (EAs) are used to solve MDO problems. The proposed methodology is tested and its performances and quality design evaluated for optimizing a 2-D air-breathing hypersonic vehicle shape at cruise flight conditions: Euler flow, Mach number = 8; angle of attack = 0°; flight altitude = 30 km, involving aerodynamics, thermodynamics and propulsion disciplines. The set up of an operational flight corridor requires a compromise among air-breathing engine performance, vehicle aerodynamic performance, and structural thermal load limit resulting from aero-heating. For this purpose, the operational efficiency is established by the effective specific impulse and thrust to weight ratio of the accelerating vehicle. In order to analyze the foregoing problems, a methodology is developed, which permits a quick performance evaluation of an idealized, integrated SCRAMJET vehicle for preliminary design analysis. A Pareto-EAs methodology is used to find design and off design solutions of an integrated vehicle consisting of the fore body inlet, the supersonic flow combustor and the after body expansion nozzle. From preliminary numerical experiments on a generic test case 2-D air breathing vehicle and analysis of results, the Pareto-EAs numerical approach is a promising methodology with game coalition for its use in industrial aeronautical design and well suited for its implementation on HPCs for increasing its efficiency.

P. Wu · Z. Tang
College of Aerospace Engineering, Nanjing University of Aeronautics
and Astronautics, Nanjing 210016, China

J. Periaux (✉)
Mathematical Information Technology Department (MIT), University of Jyvaskyla,
Finland and International Center for Numerical Methods in Engineering (CIMNE)-Universidad
Politécnica de Cataluña, Barcelona, Spain
e-mail: jperiaux@gmail.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_19

1 Introduction

Game Strategies and Pareto Games (PG) in particular are gaining importance for solving Multi- Disciplinary Optimization (MDO) in Aerospace Engineering problems over the past decade. Quite often the solution of a MDO problem can be viewed as a cooperative game between several Pareto Players linked to the physics.

In this study, Evolutionary Algorithms (EAs) are combined with (PG) to solve MDO problems. The proposed methodology is tested and its performances and quality design evaluated for optimizing the 2-D air-breathing hypersonic vehicle shape considering simultaneously aerodynamics, thermodynamics and propulsion disciplines [1–4]. The set up of an operational flight corridor requires a compromise among air-breathing engine performance, vehicle aerodynamic performance, and structural thermal load limit resulting from aero-heating.

The ad hoc design integration of a supersonic combustion ramjet engine (SCRAMJET) with an airframe drives the success of the mission of trans atmospheric or hypersonic cruise vehicles. A special attention must be given to the hypersonic atmospheric boost phase of the mission when most of the propulsive energy is expended. For this purpose, the operational efficiency is established by the effective specific impulse and the thrust to weight ratio of the accelerating vehicle. In order to analyze the airbreathing problem, a methodology is developed, which allows a fast performance evaluation of an idealized, integrated SCRAMJET vehicle [5] in the context of a preliminary design analysis. The capabilities of methodology require to master the following steps: (i) designing an integrated vehicle configuration made of the fore body inlet, the supersonic flow combustor and the after body expansion nozzle; (ii) generating the design and off-design performances data of the configuration and (iii) performing several design iterations for tradeoff studies.

The hypersonic flight conditions of the associated multidisciplinary optimization problem are: Mach number = 8; angle of attack = 0°; flight altitude = 30 km. An inviscid Euler flow model is used with no chemistry model. In the sequel, a methodology is developed, which permits a quick performance evaluation of an idealized, integrated SCRAMJET vehicle for preliminary design analysis. The four objective functions to be optimized include: acceleration performance (maximization), lift coefficient (maximization), drag coefficient (minimization) and ratio of lift to drag (L/D) (maximization).

Samples of design and off-design optimizations with analysis of generic hypersonic vehicle configurations are presented and discussed using Pareto games coupled to real coded Genetic Algorithms (GAs).

2 Integrated Forebody Surface/SCRAMJET Inlet Analysis

At hypersonic speed, the displacement angle of the bow shock wave emanating from the nose of the vehicle is very small (within a few degrees). This fact combined with the mass flow requirement of the SCRAMJET engine can cause the longitudinal length of the external wedge or conical compression device to become very large. In order to optimize the SCRAMJET performance and to keep the overall vehicle length to a minimum, it is suggested that the lower surface of the vehicle fore body be integrated to form part of the external compression device. For a preliminary design analysis, the fore body compression surfaces are modeled as a two-dimensional multi ramp wedge. The behavior of the flow field through shock waves is described using inviscid oblique shock theory. The effect of displacement thickness and friction as a result of viscous flow interaction are briefly discussed later.

The geometry of the fore body compression surface is designed in order to capture 100% of the free stream air mass flow at an arbitrarily selected design flight condition. Parameters that define the design flight condition are: Mach number (M = 8), angle of attack (AoA = 0°), and altitude (30 km). There are several ways to specify the geometry: (1) the slope of each ramp on the wedge, the longitudinal distance between the vehicle nose and the inlet lip of the cowl are given; (2) a mass flow requirement and the distance between the nose and cowl inlet position are given; (3) a mass flow requirement and an initial ramp angle are given. For designs having a constrained fore body length, the vertical position of the cowl is adjusted so the bow shock wave intercepts the cowl lip. The subsequently downstream ramp positions are longitudinally adjusted so as to make all the shock waves generated by the multi ramp wedge coalesce and intercept the cowl lip (see Figs. 1 and 2).

The strength of the bow shock is dependent upon the total compression angle measured relative to the free stream. This angle includes the initial ramp deflection and angle of attack. The greatest loss in total pressure occurs across the bow shock. Because of this, the overall total pressure recovery of the local flow field may not

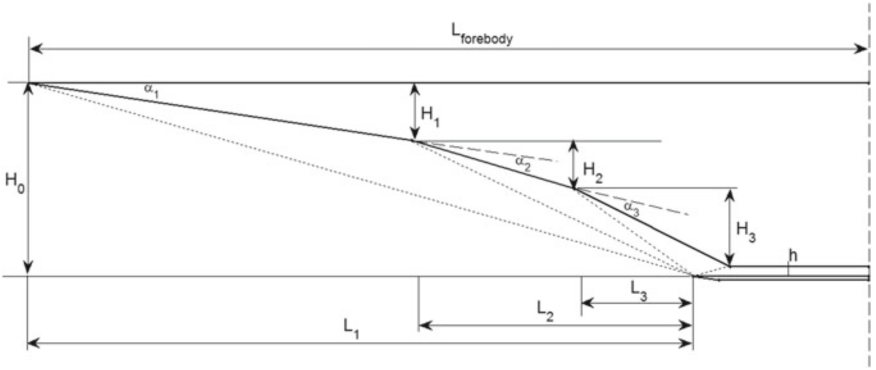


Fig. 1 Integrated fore body/inlet schematics: three shock waves converged at the lip

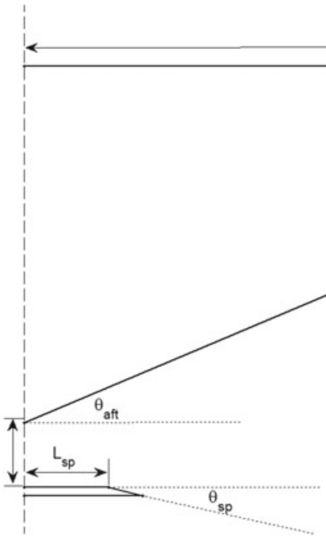


Fig. 2 Integrated after body/nozzle schematics

be improved much by having a smooth isentropic compression downstream of the nose. For this reason and for analytic simplicity, a compression surface with three ramps appears to suffice for the fore body design analysis.

The subsequent internal compression processes downstream of the cowl inlet to the combustor inlet are tailored to align the final flow direction to be parallel with respect to the vehicle body coordinate system with flux, temperature, and pressure values defined at the combustor entrance point and used as initial conditions for the subsequent combustor inlet analysis.

3 Divergent Area Supersonic Combustor Analysis

Many complexities and uncertainties exist in the supersonic combustor design, for example and among others, the turbulent mixing process of fuel in a supersonic air stream; the residence and reaction times of the fuel/air mixture that determine the supersonic combustor size; the burning characteristics of the fuel/air mixture in an environment of high-speed (order of 3 km/s) and low-pressure (order of 0.1 atmosphere) flow; the high-enthalpy flow contributing to the real-gas effects such as species dissociation of gas; the chemical reaction of combusting media,….

Many CFD analyses are being done by researchers to enhance the understanding of these complex phenomena. However, these CFD approaches are costly and unsuitable for conducting preliminary design studies. To provide a more rapid, less costly alternative methodology for use in preliminary design analysis, an idealized

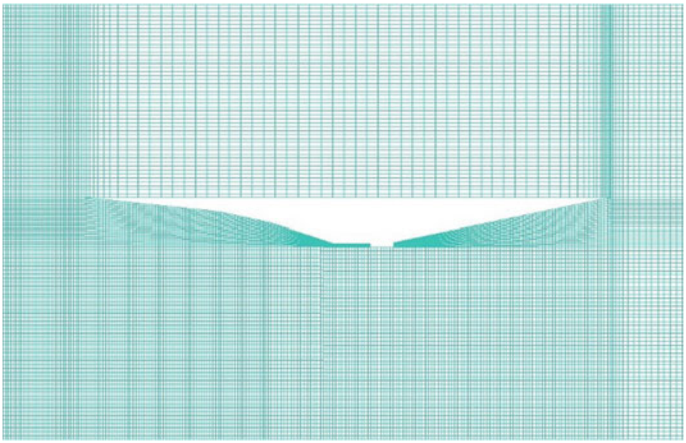


Fig. 3 Coarse grid about 30000 cells

supersonic combustor analytical model is derived. Essentially, all of these uncertainties about the combustion process are considered as if they exist in a “black box” [5]. Then engineering judgment is applied to establish a reasonable efficiency factor of energy conversion based upon the energy balance, i.e., the work exerted by the vehicle to the surrounding atmosphere is directly proportional to the available heat content of the fuel.

For a generic hypersonic vehicle with a flat top surface, the aerodynamic forces induced by the leeward surface are considered negligible. However, in reality, the aerodynamic contributions induced by the non-flat upper vehicle components should be included. For the present analysis in this lecture, aerodynamic contributions are derived from the forces on the lower fore body surface and the external cowl, which are not included in the definition of the propulsive forces.

It is often quoted in the literature that SCRAMJET performance analysis should be conducted along a constant dynamics pressure flight path. However, the imposed aero heating rate is proportional to the cube of velocity so that the heating rate increases proportionally with the velocity along a constant dynamic pressure flight path. In this case, the heating rate becomes unacceptable at higher Mach numbers. For flight in the constant heating rate corridor, the SCRAMJET performance drops off proportionally with increasing altitude. Therefore, a considerable amount of tradeoff study is required to design a viable system. The present computational methodology permits us to conduct these tradeoff studies quickly and economically.

Figures 3 and 4 describe coarse and fine mesh for simulating an Euler flow at Mach number = 8, an angle of attack = 0° and an altitude = 30 km while the simulated zoomed flow region around the scramjet configuration is presented on Fig. 5.

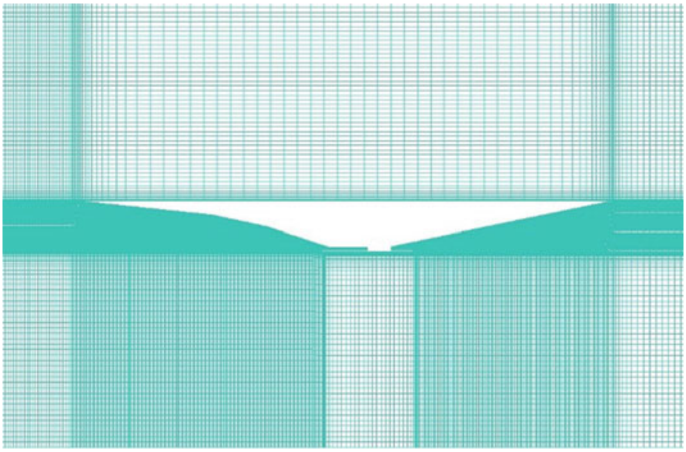


Fig. 4 Fine grid about 80000 cells

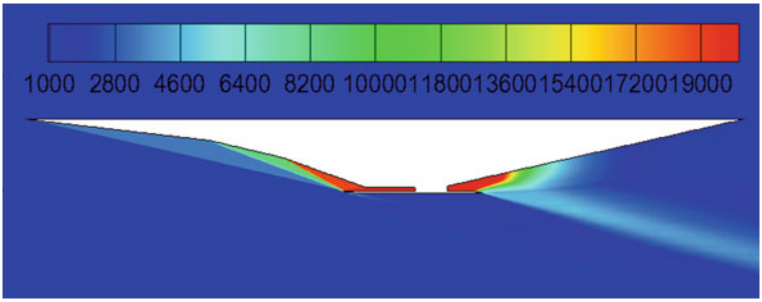


Fig. 5 Mach number contours of the flow field

4 Multi Disciplinary Optimization of Airbreathing Vehicle

In this section the optimization of a generic air breathing vehicle configuration test case described above on Figs. 1 and 2 and using evolutionary tools coupled with a Pareto game is implemented, simulated and discussed.

4.1 Introduction

This test case provides the requested computational information data to solve the multi-objective design optimization of an air breathing vehicle configuration consisting of two design elements: the fore body and the after body of a scramjet operating at hypersonic flow conditions.

4.2 Definition of the Test Case

Design the geometry of a scramjet with respect to the geometrical variables of its two forebody (Fig. 1) and after body (Fig. 2) elements described above.

Fore body shape design: three wedge shapes, three shock waves converged at the lip.

The height of the after body is equal to the forebody, once the turning angle of the upper wall $\theta_{afterbody}$ and the length L_{sp} and the turning angle of the lower θ_{sp} wall is determined, the geometric size can be determined.

by

maximizing Q: recovery of total pressure at inlet

maximizing $\mathrm{C_L/C_D}$: lift to drag ratio

maximizing $\mathrm{I_{sp}}$:

specific impulse (is defined by dividing the thrust component oriented in the flight-path direction by the fuel mass flow).

maximizing $\mathrm{I_{sp-eff}}$:

minimizing $\mathrm{T_{max}}$: maximum temperature in flow field.

4.3 Software Requirement Requirements

Three main software are requested to perform the optimization of this test case:

- an in house compressible Euler flow analyser with no chemistry
- a mesh generator used repetitively during the optimization procedure
- an EAs based evolutionary optimiser,

The optimizer is coupled with a Pareto game software.

4.4 Computational Domain

A zoomed illustration of the discretized computational domain can be found on Figs. 3 and 4.

4.5 Baseline Configuration with Prescribed Boundary Conditions on the Fore Body and After Body Geometries

Definition of the flow solver:

The mathematical modeling of the flow environment around the scramjet is de fined with the following physical hypotheses:

- Implicit Euler solver
- Perfect gas
- No chemistry reaction

The combustion process provides both the inlet computational condition as pressure-outlet, and the outlet as pressure-inlet.

The far-field is taken as pressure far-field, the wall is considered as no through wall.

The fore body and simulator equations are described below (cf. Fig. 1.).

Forebody:

$$L_1 = L_{forebody} - l$$

$$L_2 = \frac{H_0 - L_1 \ \tan \ \alpha_1}{\tan(\beta_2 + \alpha_1) - \tan \ \alpha_1}$$

$$H_1 = H_0 - L_2 \ \tan(\beta_2 + \alpha_1)$$

$$L_3 = \frac{L_2 \ \tan(\beta_2 + \alpha_1) - L_2 \ \tan(\alpha_1 + \alpha_2)}{\tan(\beta_3 + \alpha_1 + \alpha_2) - \tan(\alpha_1 + \alpha_2)}$$

$$H_2 = L_2 \ \tan(\beta_2 + \alpha_1) - L_3 \ \tan(\beta_3 + \alpha_1 + \alpha_2)$$

$$H_3 = (L_3 + l) \ \tan(\alpha_1 + \alpha_2 + \alpha_3)$$

$$H_0 = H_1 + H_2 + H_3 + h$$

$$\tan \ \beta = \frac{H_0}{L_1}$$

$$\tan \ \alpha = \frac{Ma^2 \ \sin^2 \ \beta - 1}{[1 + \frac{\gamma+1}{2} - \sin^2 \beta) Ma^2] \ \tan \ \beta}$$

$$\frac{\tan(\beta-\alpha)}{\tan\ \beta}=\frac{(\gamma-1)Ma^2\ \sin^2\ \beta+2}{(\gamma+1)Ma^2\ \sin^2\ \beta}$$

Simulator (according to P. J. Waltrup et al. results [8]):

$$L_{isolator}=0.01(0.971875(H_0/h)^2-24.0875(H_0/h)+162.3)$$

- Multi-stage simulation: Standard mesh for a 2-D hypersonic flow field analysis

4.6 Optimization and Objective Functions

The objective of the design optimization of this air breathing vehicle test case is to find the shape of the scramjet configuration. It consists of the following optimization with four (4) objective functions:

$$\begin{cases} \underset{Y}{Max}\quad \{Q; c_l / c_d; I_{sp}\} \\ \underset{Y}{Min}\{T_{\max}\} \\ Design\quad Variables\quad Y=\{\alpha_1,\alpha_2,\alpha_3; f; \theta_{afterbody}, \theta_{sp}, L_{sp}\} \end{cases}$$

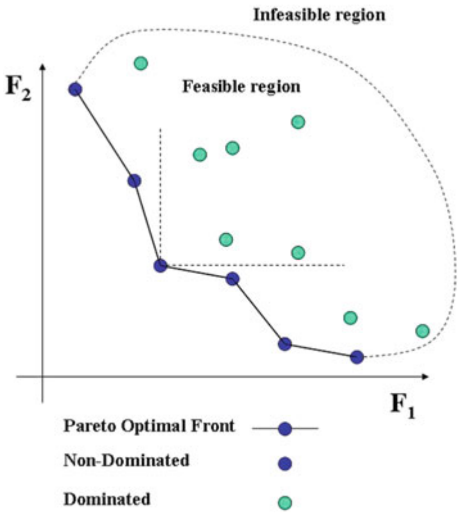
The Pareto shape optimization procedure used in this air breathing vehicle test case is implemented with the real coding NSGA II software. We refer to Pareto [6] and Deb [7] for the description of this optimizer.

4.7 Cooperative Games: Pareto Optimality

The main interest of a Pareto based evolutionary algorithm is that the optimization process has not to aggregate objectives. It provides the fitness values of a candidate solution directly from the comparison of their respective objective vectors.

An usual way to represent the solutions of a multi-objective problem is based on the concept of Pareto optimality or non-dominated individuals [6]. Figure 6 shows the Pareto optimality concept for a two conflicting objectives problem. The solution of a multi-objective problem is the Pareto optimal set or Pareto Front which is captured using a cooperative game identifying the set of non-dominated solutions. This process spans the complete range of compromised designs between the two objectives. Most real world problems involve a number of inseparable objectives where there is no

Fig. 6 Pareto optimality



unique optimum, but a set of compromised individuals known as Pareto optimal (or non-dominated) solutions. We use in this air breathing vehicle multi objective optimization problem the Pareto optimality principle: a solution to a multi-objective problem is considered Pareto optimal if there is no other solutions that better satisfy all the objectives simultaneously. The Pareto optimal solutions then represent a trade-off information among the objectives.

Definition of a non dominated solution:

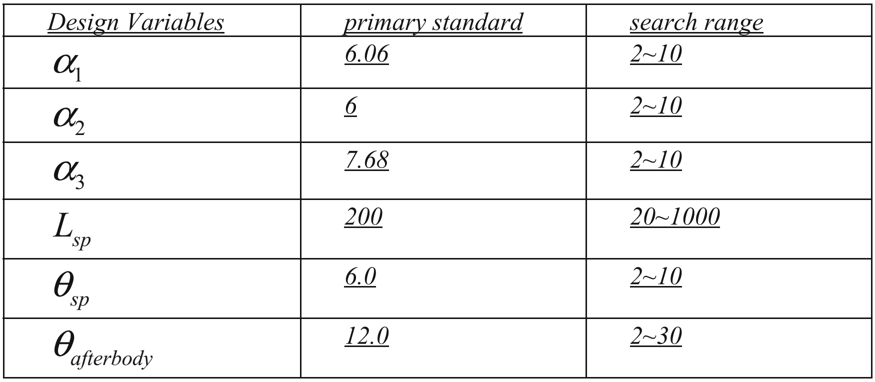
For a two objectives minimization problem, a vector x_1 is said partially less than vector x_2 if and only if:

$$\forall i : \ f_i(x_1) \leq f_i(x_2) \ \ and \ \ \exists_i : \ \ f_i(x_1) < f_i(x_2)$$

In this case the solution x_1 *dominates* the solution x_2.

As EAs consider multiple points simultaneously, they are capable of finding a number of solutions in a Pareto set. Pareto selection ranks the population and selects the non-dominated individuals for the Pareto front. A comprehensive theory, literature review and implementation of Multi-objective EAs (MOEAs) including the NSGAII and VEGA algorithms can be found in Deb [7].

More details and applications of Pareto games can be found in Periaux et al. [8].



Design Variables	*primary standard*	*search range*
α_1	*6.06*	*2~10*
α_2	*6*	*2~10*
α_3	*7.68*	*2~10*
L_{sp}	*200*	*20~1000*
θ_{sp}	*6.0*	*2~10*
$\theta_{afterbody}$	*12.0*	*2~30*

Fig. 7 Design upper/lower bounds for the air breathing vehicle geometry

4.8 Design Point

The optimizer requires information of candidate solutions provided by the flow analyzer operating at the following cruise flight conditions:

- Altitude = 30 km M = 8, $\alpha = 0$

Atmospheric parameters:

- Pressure 1197 Pa
- Density 0.0180 kg/m3
- Temperature 226.65 K
- Sonic speed 301.7 m/s

4.9 Design Parameters: Search Space for Shape Optimization

Six (6) design variables are used to optimize the shape of the scramjet geometry. For each variable, upper and lower bounds give limits of the search space of non dominated solutions.

Design bounds for the optimization problem are shown on Fig. 7.

4.10 Outputs Results

The major outputs of the geometry optimization include:

- The capture of the non-dominated solutions of the Pareto front;

- Optimized design variables of a non dominated solution named Pareto Member (PM)
- Values of the objective functions of a non-dominated solution named Pareto Member (PM)
- Zoomed Mach numbers of one Pareto member (PM) flow field.

4.11 Some Preliminary Results Obtained with NSGA II Software for the Shape Optimisation Scramjet Problem

Two (2) non-dominated solutions members (Pareto members PM1and PM2) on the Pareto Front shows optimized shape geometries of airbreathing vehicles.

The total computational time in terms of CPU spent for functions evaluations a 40 generations run on a PC cluster of 72 CPU with 80 individuals using NSGA II software and a mesh of 100000 cells takes 77 h 20 min to capture the Pareto Front. The convergence history of the Pareto Front of the five objective functions are represented on Figs. 8, 9 and 10.

Figure 11 shows the optimal shape geometry of the air breathing vehicle obtained with optimal values of the design variables given above.

Figure 12 shows the optimal shape geometry of the air breathing vehicle obtained with optimal values of the design variables given above.

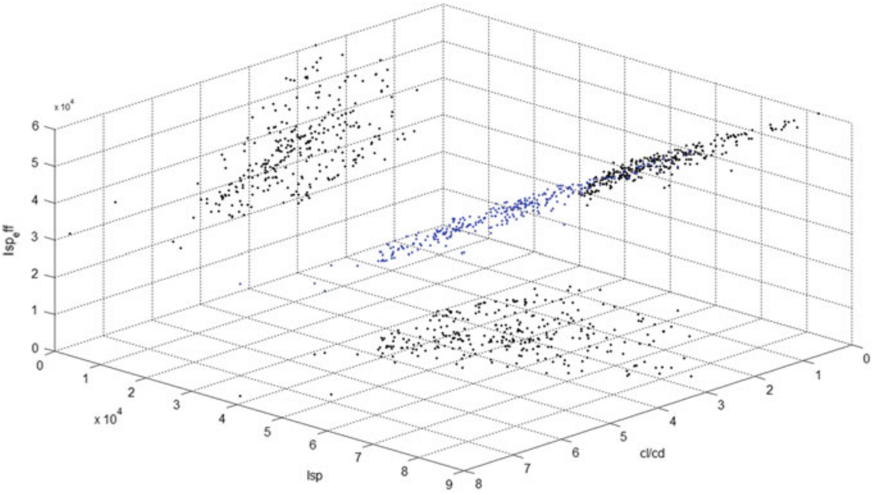


Fig. 8 Non dominated solutions capture of Cl/Cd, Isp and Isp-eff

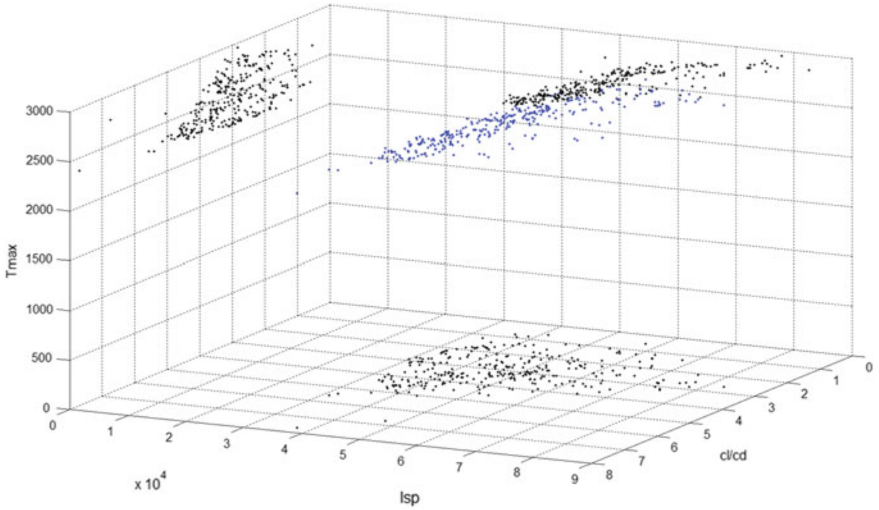


Fig. 9 Non dominated solutions capture of Cl/Cd, Isp and Tmax

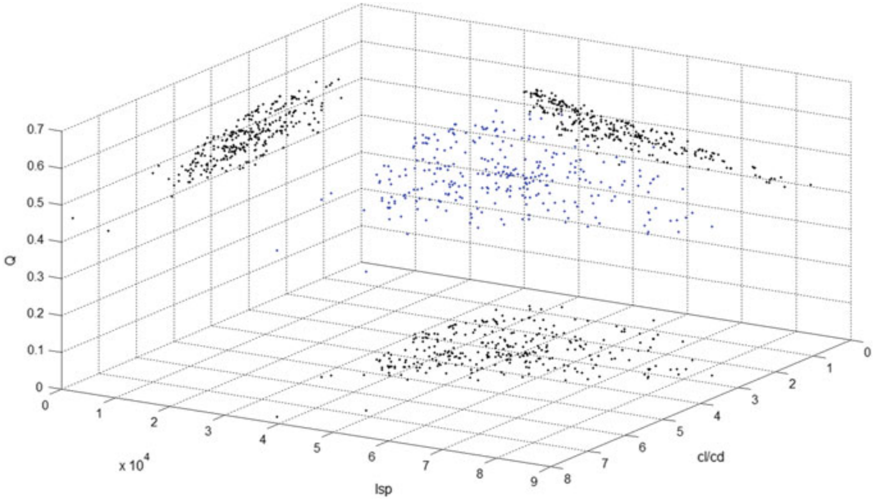
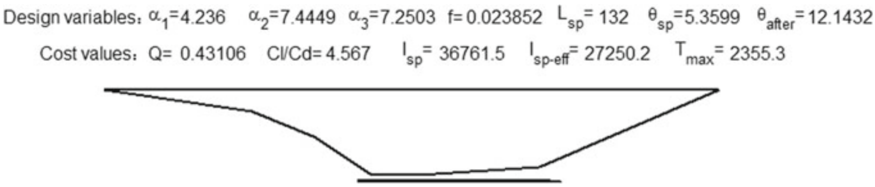


Fig. 10 Non dominated solutions capture of Cl/Cd, Isp and Q

5 Conclusion

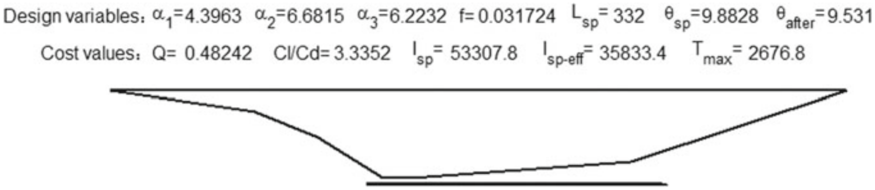
From the preliminary numerical experiments and presentation of results, it is concluded that the Pareto-EAs methodology is satisfactory for conceptual designs of hypersonic air breathing vehicles considering multi disciplines and constraints. In



Design variables: α_1=4.236 α_2=7.4449 α_3=7.2503 f=0.023852 L_{sp}= 132 θ_{sp}=5.3599 θ_{after}=12.1432

Cost values: Q= 0.43106 Cl/Cd= 4.567 I_{sp}= 36761.5 $I_{sp\text{-}eff}$= 27250.2 T_{max}= 2355.3

Fig. 11 Non dominated design solution with design variables and values of objective functions (PM1)



Design variables: α_1=4.3963 α_2=6.6815 α_3=6.2232 f=0.031724 L_{sp}= 332 θ_{sp}=9.8828 θ_{after}=9.531

Cost values: Q= 0.48242 Cl/Cd= 3.3352 I_{sp}= 53307.8 $I_{sp\text{-}eff}$= 35833.4 T_{max}= 2676.8

Fig. 12 Non dominated design solution with design variables and values of objective functions (PM2)

order to be used in industrial design environments, computational efficiency taking into account more complex physics modeling has to be improved with the coupling of EAs with others games (Nash, Stackelberg) or games coalition (Nash-Pareto) with EAs, Periaux et al. [8] Due to their easy implementation on HPCs a significantly increase of efficiency will be available in the next phase of this study.

References

1. Study of an Air-Breathing Engine for Hypersonic Flight (Figs. 9–10) (PDF). Universitat Politecnica de Catalunya. Retrieved 3 June 2015
2. Harsha, P.T., Keel, L.C., Castrogiovanni, A., Sherrill, R.T.: X-43A vehicle design and manufacture. AIAA 2005–3334
3. Heiser, W.H., Pratt, D.T., Daley, D.H., Mehta, U.B.: Hypersonic Airbreathing Propulsion. American Institute of Aeronautics and Astronautics, Washington, D.C (1994)
4. Roudakov, A., Semenov, V., Hicks, J.: Recent flight test results of the joint CIAMNASA Mach 6.5 Scramjet Flight Program. NASA/TP-1998-206548
5. Waltrup, P.J., White, M.E., Zarlingo, F., Gravlin, E.S.: History of Ramjet and Scramjet Propulsion Development for U.S. Navy Missiles, 50 years Milton S. Eisenhower R&T Center (1997)
6. Pareto, V.: Cours d'Economie Politique. Rouge, Lausanne, Switzerland (1896)

7. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. Wiley (2003)
8. Periaux, J., Gonzalez, F., Lee, D.C.: Evolutionary Optimization and Game Strategies for Advanced Design: Applications to Aeronautics, Control and Automation. Science and Engineering Series, vol. 75. Springer (2015)

Part VI
Optimisation Under Uncertainty

Innovative Methodologies for Robust Design Optimization with Large Number of Uncertainties Using ModeFRONTIER

Alberto Clarich and Rosario Russo

Abstract This paper describes the methodologies that have been developed by ESTECO during the first phase of UMRIDA European Project, in the field of Uncertainty Management and Robust Design Optimization, and that have been implemented in the software platform modeFRONTIER. In particular, in the first part there are proposed two methodologies, one based on SS-ANOVA regression applied directly to the uncertainties variables and one based on a stepwise regression methodology applied to the Polynomial Chaos terms used for the uncertainty quantification. Aeronautical test cases proposed by UMRIDA consortium are used to verify the validity of the methodologies. In the second part, the state of art methodologies for Robust Design Optimization are compared with a new proposed approach, based on a min-max definition of the objectives, and the application of Polynomial Chaos coefficients for an accurate definition of percentiles (reliability-based robust design optimization). Also in this case an Aeronautical CFD test case is proposed to validate the methodologies.

1 Introduction

As proved by recent studies [1], one of the most efficient methodology to manage accurately the uncertainties is the application of Polynomial Chaos expansion [2]. This methodology however requires a minimum number of samples which increases heavily with the number of uncertainties, and a typical industrial optimization case (for instance at least 10 simultaneous uncertainties) can be hardly treated as a feasible task.

For this reason, we propose in this paper some approaches to handle efficiently industrial problems of this kind, both on the side of Uncertainty Management (UQ) and on the side of Robust Design Optimization (RDO).

A. Clarich (✉) · R. Russo
ESTECO SpA, Area Science Park, Padriciano 99, 34151 Trieste, Italy
e-mail: engineering@esteco.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_20

For the UQ, the proposed solution is to use methodologies that allow to identify which are the uncertainties having higher statistical effects on the performances of the system, this way allowing to apply the Polynomial Chaos expansion with a smaller number of uncertainties, and therefore a significant smaller number of samples, on a system which is statistically equivalent to the original. As alternative, the uncertain parameters effects analysis can be applied directly to the Polynomial Chaos terms, this way reducing the number of unknown coefficients and therefore the number of needed samples to complete the UQ, without discarding necessarily at all an uncertain variable from the problem (keeping this way an higher accuracy).

For the RDO methodologies, we propose a methodology based on the min-max formulation of objectives, which guarantees the reduction of objectives numbers with respect to a classical RDO approach [3], and therefore the possibility of reducing drastically the number of configurations to be evaluated and of simulations to be performed. In order to guarantee an accurate application of this methodology, we developed an approach which is based on the exploitation of Polynomial Chaos coefficients to evaluate accurately the percentiles of the quantities to be optimized/constrained. This methodology is also called reliability-based design optimization [4], and the solution we propose, based on Polynomial Chaos exploitation, is innovative and very promising in terms of efficiency.

2 UQ of Large Number of Variables: SS-ANOVA and Stepwise Regression for Sparse Collocation

Smoothing Spline ANOVA (SS-ANOVA) [5] models are a family of smoothing methods suitable for both uni-variate and multi-variate modeling/regression problems characterized by noisy data, given the assumption of Gaussian-type responses. In particular, SS-ANOVA is a statistical modeling algorithm based on a function decomposition similar to the classical analysis of variance (ANOVA) decomposition and the associated notions of main effect and interaction. Each term - main effects and interactions—can be used to reveal the percentage contribution of each single uncertain parameter, and of any uncertain variables pair, on the output global variance, since in a statistical model the global variance can be explained (decomposed) into single model terms.

A generic multi-variate regression problem could be stated as a constrained minimization problem, that can be expressed in Lagrangian terms as:

$$\min L\left(f\right)+\frac{\lambda}{2}J\left(f\right) \tag{1}$$

where *L(f)* is defined as *minus log likelihood* of the model *f(x)* given the data, to be minimized to maximize the data fit, and *J(f)* is defined as a *quadratic roughness functional*, to be subjected by a constraint—*J(f)* $\leq \rho$—that can be used to preserve the overfitting (a large roughness guarantees a smoother model, while smaller values imply rougher functions but better agreement to the data).

We can generally assume that the regression model $f(\boldsymbol{x})$ can be expressed as a sum of N independent components $f_j(x_j)$, each one function of a single variable x_j. By this approximation, the regression model would take into account only main effects (the effect of each single variable).

A more complete regression model, which has to consider also interaction effects, will include in $f(\boldsymbol{x})$ also the interaction terms $f_{ij}(x_i, x_j)$.The smoothing parameters, needed to solve the regression problem, can be determined by a proper data-driven procedure, such as the generalized cross validation (GCV), as described in [6]. The number of decomposition terms, which is equal also to the minimum number of needed sampling points, is equal to $N(N - 1)/2$ with N number of variables.

We can therefore apply the definition of internal product projecting the $f(\boldsymbol{x})$ to any component f_k obtaining the value of its contribution (or probability), by the (normalized) expression:

$$\pi_k = \frac{\langle f_k, f\rangle}{\|f\|^2} \tag{2}$$

Expression 2 is called *contribution index* $_k$ and expresses the relative significance of the different terms composing the model, therefore the contribution of each variable main effect or interaction effect.

As alternative to the first methodology for UQ here proposed, we have adopted another approach [7], which consists in applying a regression analysis directly on the Polynomial Chaos expansion (PCE) expression, in other words the PCE will keep only those terms which actually affect the output, discarding the others.

The methodology consists first in ranking the terms using a Least Angle Regression (LAR) technique [8] and then in assessing how many PCE terms should be kept.

The LAR ranking is accomplished by the following procedure ($\mathbf{P}_{\boldsymbol{a}_i}$ represents a generic PCE term).

- Set residual $\mathbf{Res} = \mathbf{output} - \text{mean}\,(\mathbf{output})$
- The first selected polynomial term $\mathbf{P}_{\boldsymbol{a}_1}$ is the one with the highest correlation with $\mathbf{Res}$ namely: $\mathbf{P}_{\boldsymbol{a}_1}$ such that $\text{corr}\left(\mathbf{P}_{\boldsymbol{a}_1}, \mathbf{Res}\right) = \max\left(\text{corr}\left(\mathbf{P}_{\boldsymbol{a}_i}, \mathbf{Res}\right)\right)$
- Set $\mathbf{P}_{\boldsymbol{a}_i} = \mathbf{P}_{\boldsymbol{a}_1}$
- For k from 1 to the number of PCE terms to be ranked do:
    - Set $\mathbf{Res} = \mathbf{Res} - \lambda\mathbf{P}_{\boldsymbol{a}_i}$ where λ is such that: $\text{corr}\left(\mathbf{P}_{\boldsymbol{a}_i}, \mathbf{Res}\right) = \text{corr}\left(\mathbf{P}_{\boldsymbol{a}_j}, \mathbf{Res}\right)$, the polynomial $\mathbf{P}_{\boldsymbol{a}_j}$ is selected.
    - Solve a least square problem: find c_i and c_j that minimize $(c_\mathrm{i}\mathbf{P}_{\boldsymbol{a}_i} + c_\mathrm{j}\mathbf{P}_{\boldsymbol{a}_j} - \mathbf{Res})^2$
    - Set $\boldsymbol{P}_{\boldsymbol{a}_i} = c_\mathrm{i}\mathbf{P}_{\boldsymbol{a}_i} + c_\mathrm{j}\mathbf{P}_{\boldsymbol{a}_j}$ (new direction)
- Next k

The order of selection of the PCE terms will reflect a ranking based on how much each term affects the output. Once the ranking is done it is necessary to establish a way to choose how many PCE terms should be kept. The criterion for this is based on the Mean Leave one Out Error (Err_{LOO}).

$$Err_{LOO} = \frac{1}{N}\sum\nolimits_{i=1}^{N}\Delta_i^2 \tag{3}$$

where N is the number of samples and

$$\Delta_i = \text{output}(\mathbf{x}_i) - \widehat{M}\left(\mathbf{x}_i\right) \tag{4}$$

i.e. the difference between the output corresponding to the i-th sample and the output computed from the PCE obtained excluding from the training samples the i-th.

It is possible to show [6] that Δ_i can be estimated by the following expression:

$$\Delta_i = \frac{\text{output}(\mathbf{x}_\text{i}) - M\left(\mathbf{x}_i\right)}{1 - h_i} \tag{5}$$

where $M\left(\mathbf{x}_i\right)$ is the output evaluated by the PCE computed this time using all the samples and

$$h_i = diag\left(\mathbf{P}\left(\mathbf{P}^T\mathbf{P}\right)^{-1}\mathbf{P}^T\right) \tag{6}$$

where $P_{ij} \equiv \mathbf{P}_{\boldsymbol{a}_j}(\mathbf{x}_i)_{\boldsymbol{i}=1\ldots \boldsymbol{N};\boldsymbol{j}=1\ldots \boldsymbol{Nterms}}$

Using the previous expressions, given a certain number of PCE terms, it is possible to compute the corresponding Err_{LOO}.

The criterion to select the number of terms consists in monitoring the two quantities:

$$R_{LOO} \equiv 1 - \frac{Err_{LOO}}{var\left(\mathbf{output}\right)} \tag{7}$$

$$R_{squared} \equiv 1 - \frac{Err_{squared}}{var\left(\mathbf{output}\right)} \tag{8}$$

where $Err_{squared}$ is the squared error sum, i.e. the sum of the squared differences between each sample output and the corresponding output value estimated using the PCE (this time using all the samples); var (**output**) is the output variance considering all the samples.

$R_{squared}$ and R_{LOO} are functions of the number of PCE terms: the first will generally increase as the number of terms increases, while the second tends initially to increase as the number of terms increases, but from a certain number of terms on, it starts showing a decreasing trend.

$R_{squared}$ is sensitive on how much the PCE expansion is able to approximate the output, while R_{LOO} is sensitive to overfitting problems: the ideal number of terms should guarantee a good compromise, namely $R_{squared}$ around 0.9, or higher, and R_{LOO} close to its maximum before the decreasing trend due to overfitting.

Table 1 List of uncertainties

Uncertainty	Most likely value (m)	Minimum value (a)	Maximum value (b)	PDF-type
Free stream mach	0.729 (case 6) 0.734 (case 9)	95% m	105% m	Symmetric beta-PDF
Angle of attack	2.31° (case 6) 2.54° (case 9)	98% m	102% m	Symmetric beta-PDF
Thickness-to-chord	Nominal value from geometry	97% Nominal	103% Nominal	Symmetric beta-PDF
Camber line	Nominal value from geometry	Nominal-0.01% chord	Nominal + 0.01% chord	Symmetric beta-PDF

Of course any different strategy for ranking the PCE terms and choosing the proper number of terms could be chosen, it is important however to employ a strategy somehow sensitive to both accuracy and overfitting control. In any case the proposed strategy has got the advantage of keeping separated the ranking algorithm from the number of terms choice.

The described approach gives the important benefit of reducing the global number of unknown coefficients for the PCE expansion, and therefore giving the possibility as well of reducing the number of sampling points, needed for the PCE training.

Conversely from the SS-ANOVA methodology, however, the great advantage is that any uncertainty is not necessarily discarded, but its effect might be included in a smaller set of polynomial terms.

3 UQ Test Case Application

To validate the methodologies proposed in previous section, we have applied them to the test case BC-02 of UMRIDA European Project [9].

The test case consists in the UQ quantification of a RAE 2822 airfoil [10], for a specified conditions, and for a total of 13 uncertainties (operational and geometrical). Nominal parameters and uncertainties type and parameters are defined in Table 1.

In particular, the geometrical uncertainties refer to the camber line and the thickness-to-chord ratio of the nominal profile, which have been fitted by a Bezier parametric curve [11], of respectively 7 and 8 control points, uniformly spaced in the abscissas. Since the extreme points are fixed, we consider a total of 5 uncertainties for the (ordinates of) control points of the thickness-to-chord curve (named from *Y1_thickness* to *Y5_thickness*), and 6 for the camber curve (named from *Y1_chord* to *Y6_chord*).

Figure 1 reports the process workflow created for the set-up of this test case in modeFRONTIER software from ESTECO. Each component of the process is defined by dedicated modules (input variables, CAD/CAE interfaces, output variables) interconnected between them, in order to allow the automatic execution of the simulations

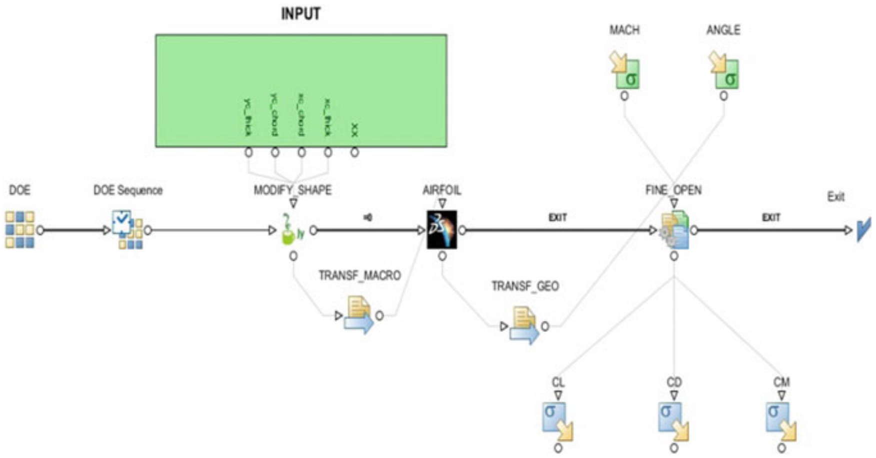


Fig. 1 Workflow for process automation in modeFRONTIER

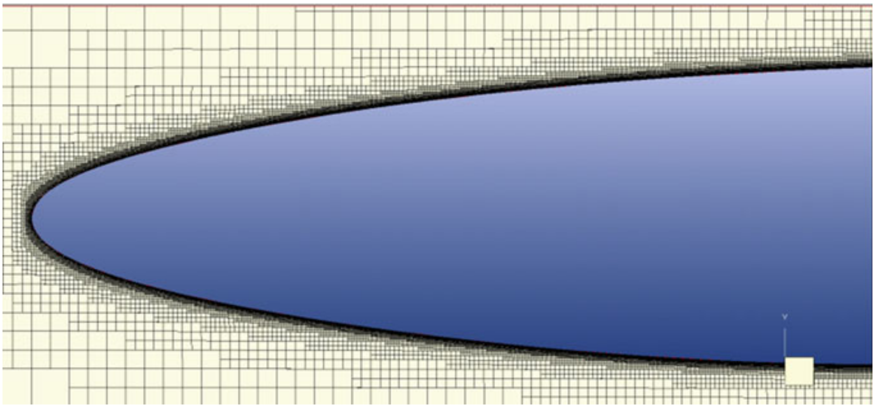


Fig. 2 Mesh overview and detail in FINE/open model

for each design sample which is proposed by the selected algorithm for the UQ. modeFRONTIER software contains as well all the tools needed to complete automatically the UQ of the required parameters.

The mesh provided for this test case has been elaborated by ESTECO in FINE/Open software from NUMECA. The mesh is characterized by an overall number of cells equal to about 1/2 million, which require an average time to complete the simulation of one design sample in about 1 h, using a 2-cpu machine. Around the airfoil the mesh is refined, because it is important to reduce the effect of numerical uncertainties, which are not considered in the problem.

Figure 2 reports a detail of the mesh used.

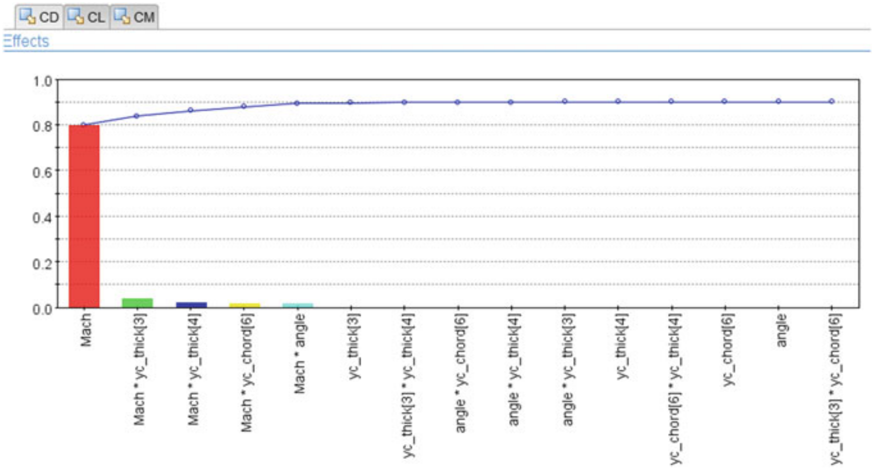


Fig. 3 SS-ANOVA (modeFRONTIER) for each output

Inlet conditions are specified in function of the Reynolds number defined for this test case (Re = 6.5 × 10^6) and of the Mach number specified for this test case (see Table 1).

In the remaining lateral boundaries of the model, an outlet condition is specified, while a symmetry condition is specified on the planes parallel to the flow.

A full turbulent (*Spalart-Allmaras*) model is used; an adaptive meshing procedure is defined, to refine the mesh where the gradient of the pressure is higher.

The first step in the UQ of the test case is the definition of a large DOE (Design of Experiments) using a Latin Hypercube algorithm, considering all the 13 original uncertainties.

For this purpose we have evaluated a series of 105 designs, which is the minimum number of samples to apply a Polynomial Chaos Expansion of order 2 for the UQ, and then repeated the analysis for a larger number (200) of samples.

These samples are also used to apply the SS-ANOVA screening analysis, which indicates the relative effect of each parameter for the selected output (Cd, Cl and Cm). Figure 3 in fact illustrates for one of the outputs (Cd) the relative effect of each uncertain parameter (using the name conventions described above in this section), including in the analysis also the interaction effects.

Considering for each output a cumulative effect of at least 90%, we can conclude that the parameters most important are:

- For CD: *Mach, Y3_thickness, Y4_thickness;*
- For CL: *Mach, angle, Y1_chord, Y5_chord*;
- For CM: *Mach, Y2_chord*.

Table 2 UQ results for test case defined in Table 1

	Cd mean	Cd Sigma	CI mean	CI Sigma	Cm mean	Cm Sigma
200 samples-13 uncertains	1.951E−2	2.12E−3	6.39E−1	1.93E−2	1.035E−1	4.04E−3
100 samples-13 uncertains	1.953E−2	1.96E−3	6.39E−1	1.82E−1	1.035E−1	3.86E−2
	err: 0.1%	**err: 7.6%**	**err: 0%**	**err: 8.4%**	**err: 0%**	**err: 8.6%**
45 samp/es-SS-ANOVA (7 uncertains)	1.951E−2	2.09E−3	6.39E−1	2.01 E−2	1.035E–1	4.11E−3
	err: 0%	**err: 1.4%**	**err: 0%**	**err: 4.1%**	**err: 0%**	**err: 1.7%**
30 samples-stepwise regression	195E−2	2.14E−3	6.39E−1	1.86E−2	1.035E−1	3.96E−3
	err: 0%	**err: 0.6%**	**err: 0%**	**err: 3.5%**	**err: 0%**	**err: 2.0%**

So, globally, the 7 common parameters most important are: *Mach, angle, Y3_thickness, Y4_thickness, Y1_chord, Y2_chord, Y5_chord*.

In other words, we could exclude from the analysis the less significant uncertainties, keeping statistically almost the same information on the UQ of the outputs.

To validate this hypothesis, we have in a second step fixed the 6 not significant uncertainties (to their nominal values), and defined a UQ problem of 7 uncertainties only.

For this reduced problem, a much smaller number of simulations is required and precisely at least 45 samples to apply a Polynomial Chaos expansion of order 2; Table 2 more over will report the results of the UQ analysis.

Applying instead the second methodology (LAR), we have found that only 10 terms are needed to give acceptable errors on a database of 30 samples only, and precisely (the number in the terms notation below refer to the defined order of the variables, which is: *1-Mach, 2-Angle, 3-Y1_camber, 4-Y2_camber, 5-Y3_camber, 6-Y4_camber, 7-Y5_camber, 8-Y6_camber, 9-Y1_thickness, 10-Y2_thickness, 11-Y3_thickness, 12-Y4_thickness, 13-Y5_thickness*; the apex refers to the exponent of the term, and the _ character refers to an interaction between two terms):

- For CD*: 1, 1ˆ2, 11, 2, 1_12, 2_9, 1_3, 10, 1_7, 1_11*;
- For CL: *1, 1ˆ2, 2, 1_7, 1_13, 13, 6, 10, _13, 3*;
- For CM: *1, 1ˆ2, 2, 13, 1_7, 8, 1_13, 11, 10, 1_4*.

Table 2 therefore reports the UQ results for the test case, after the application of Polynomial Chaos expansion (in modeFRONTIER software) of order 2 for the different DOEs analyzed, and in particular: (1) 200 samples with all the uncertainties, (2) 105 samples with all the uncertainties, (3) 45 samples with 7 most important uncertainties (SS-ANOVA application), (4) 30 samples with 10 most important PCE terms.

The results of the test are satisfactory. Applying the first methodology, using 45 samples only it was possible to determine the main momentum averages with a practically absolute accuracy and the standard deviation by an error (computed on the basis of the largest DOE results) between 1 and 4% (higher for lift coefficient and lower for drag).

Applying the second methodology, the results are even slightly improved, since by a lower number of samples, 30, the highest error on standard deviation has been reduced from 4.1 to 3.5%. This second method, in addition, is independent from the significance of the single parameters, that in this particular test case may have given advantage to the first method (having one variable, Mach, predominant in the global variance).

In general, it emerges clearly that a significant reduction of the number of needed samples, reachable by any of the two methodologies, produces an accurate evaluation of the statistical moments, with a contained maximum estimated error.

4 RDO: Classical Versus MINMAX Approach

In order to apply RDO to a problem of industrial relevance, i.e. a problem characterized by a large number of uncertainties and by simulation times which are expansive, it is not just needed to define only an efficient UQ methodology, which can give accurate results with few simulations, but also an efficient optimization approach.

The first approach that we propose in this chapter as the state of the art, is the classical RDO approach [3] based on the definition of a multi-objective optimization problem, consisting generally on the optimization of the mean value of the performances and on the minimization of their standard deviation.

This approach guarantees the definition of a complete Pareto frontier as trade-off of the optimal solutions, in terms of mean performance and in terms of their stability or robustness. This means that at the end of the optimization the designer has the freedom to select the best solutions accordingly to a large variety of possibilities depending on which criteria should be privileged.

The problem of this approach is that a Multi-objective Optimization algorithm is to be chosen, since the definition of a single objective as weighted sum of the different criteria cannot be proposed for the impossibility of knowing a priori the proper weights of the particular optimization problem. Multi-objective Optimization algorithms are in fact generally very robust, but they require a number of simulations generally very much consistent with respect to a single objective optimization case, and for a RDO problem the number of simulations may be not feasible from a practical point of view (this number being multiplied by the sampling size for each design to obtain the overall number of simulations required).

In order to reduce the overall number of simulations for a RDO problem, we propose in this section another approach, described in one of our previous works [4].

The basic idea is to reduce the number of objectives, so that a single-objective algorithm, which requires much less simulations for the convergence, could be applied.

To achieve this purpose the so called *min-max* or *max-min* approach is followed. The idea is to maximize the minimum or worst performance of a distribution function that is to be maximized (for instance the aerodynamic efficiency of a wing), or to minimize the maximum or worst limitation that is to be minimized (for instance the drag coefficient of a wing).

The effect of this approach is the "shift" of the performance distribution in the desired direction, so in a certain sense both the average performance and the stability at the uncertainties are optimized. Considering for instance the drag coefficient distribution of an airfoil: the optimized configuration distribution by this approach will be shifted below the baseline distribution, since we minimize the maximum value of the distribution or the value of its higher tail.

Besides of this one objective, of course other criteria shall be considered (like lift and momentum), but if they can be expressed as constraints, a single-objective algorithm could still be applied.

At this point, before analyzing the possible single-objective algorithms that may be chosen, it is opportune to discuss about the definition of maximum and minimum values of a distribution.

In the case of a Normal distribution of the performance, since it is unlimited, the concept of the extremes may be replaced by a given percentile of the distribution, for instance 95 or 99%. Usually, the reference value is 99.73% because for a Normal distribution it corresponds to the 3Sigma level.

This analysis is also called Sig-Sigma, since six time standard deviation corresponds to the 99.73% of the complete distribution, a value that can be assumed enough representative of the whole distribution.

Now, since by Polynomial Chaos analysis we can compute mean and standard deviation with high accuracy, the computation of the maximum or minimum value with the expression MEAN $\pm 3\sigma$ can be made for each design of the RDO optimization, therefore the objective function can be defined this way.

The limitation of this approach occurs when the performance does not follow a Normal distribution: in this case, the Six-Sigma formulation may not correspond exactly to the correct percentile of the distribution, so from design to design the computation of the objective function could be not accurate. This problem is even more evident for a particular class of RDO problems, the Reliability-based design optimization, where any constraint should be defined accurately on a given percentile of the distribution.

In next chapter we will propose a new methodology, based on the application of Polynomial Chaos on the Reliability-RDO, in order to solve the problem of the accuracy of the min-max approach and make the RDO optimization more efficient. For the moment, as illustration of the state of the art, we follow the Six-Sigma approach for the min-max strategy, compared with the classical two-objectives approach.

In this case we consider a test case derived from the one illustrated in Table 1. For simplicity, we consider only 3 uncertainties for the RAE2822 airfoil, with nominal values equal to 0.734 for free stream Mach number, 2.79° for angle of attach and Reynolds number equal to 6.5E6. The uncertainties are given by a Normal distribution for Thickness-to-chord profile (a single uncertainty factor which multiplies the thickness profile), Mach number and angle of attack, defined by a standard deviation respectively equal to 0.005, 0.005 and 0.1.

The first optimization strategy applied is the multi-objective approach (3.1), considering the following objectives and constraints:

- Obj.1: Minimize mean value of Cd
- Obj.2: Minimize standard deviation of Cd
- Constraint 1: mean value of Cm + 3* standard deviation of Cm < 0.1305
- Constraint 2: mean value of Cl − 3* standard deviation of Cl > 0.9

The last two constraints are imposed by the necessity to guarantee a minimum value of Cl and a maximum value of Cm respectively less and higher than an arbitrary extreme percentile of the baseline distributions, here 99.97%, that can be approximated considering a Normal distribution using the expressions above (Six Sigma rule). A number of 10 sampling points for design was found to be necessary to guarantee an accurate UQ using a Polynomial Chaos expansion of the second order.

The multi-objective approach then consists in the minimization of the mean value and of the standard deviation of the drag coefficient, and we applied a Game Theory algorithm [12] (MOGT in modeFRONTIER) in order to obtain good compromise results by a lower number of simulations than a classical GA algorithm.

Nonetheless, after the evaluation of more than 50 designs (for a total of 500 CFD simulations, that corresponds to about 20 days using a double cpu machine), it was practically impossible to find feasible solutions that improve the original baseline. The optimization approach has been stopped, because the optimization time was considered already excessive for a problem of industrial relevance.

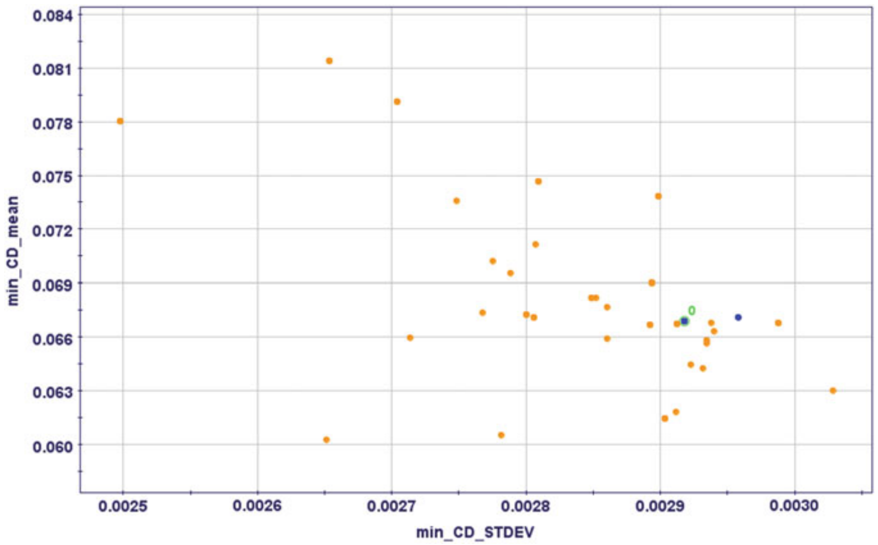


Fig. 4 Optimization results using classical RDO approach

Figure 4 above reports the results obtained following this approach: the two objectives are reported in ordinate (average) and in abscissa (standard deviation), and each point represents a different design proposed during the optimization. The orange color indicates that the design is unfeasible, i.e. that does not respect the constraints, while the blue color indicates that all the constraints are respected.

Beside design 0 (the baseline), only another feasible design has been obtained, without however improving significantly the objectives (standard deviation is higher).

At this point we have then decided to adopt the second methodology, i.e. considering a single objective optimization problem, following *the max-min* approach.

The optimization problem becomes then described as follows:

- Obj.1: Minimize mean value of Cd + 3* standard deviation of Cd
- Constraint 1: mean value of Cm + 3* standard deviation of Cm < 0.1305
- Constraint 2: mean value of Cl − 3* standard deviation of Cl > 0.9

Besides the two constraints on Cl and Cm distributions, mean and standard deviation of Cd have been compacted together into a single objective, which is the minimization of a given high percentile (still 99.97) of the Cd distribution, that can be considered as the “maximum” target value.

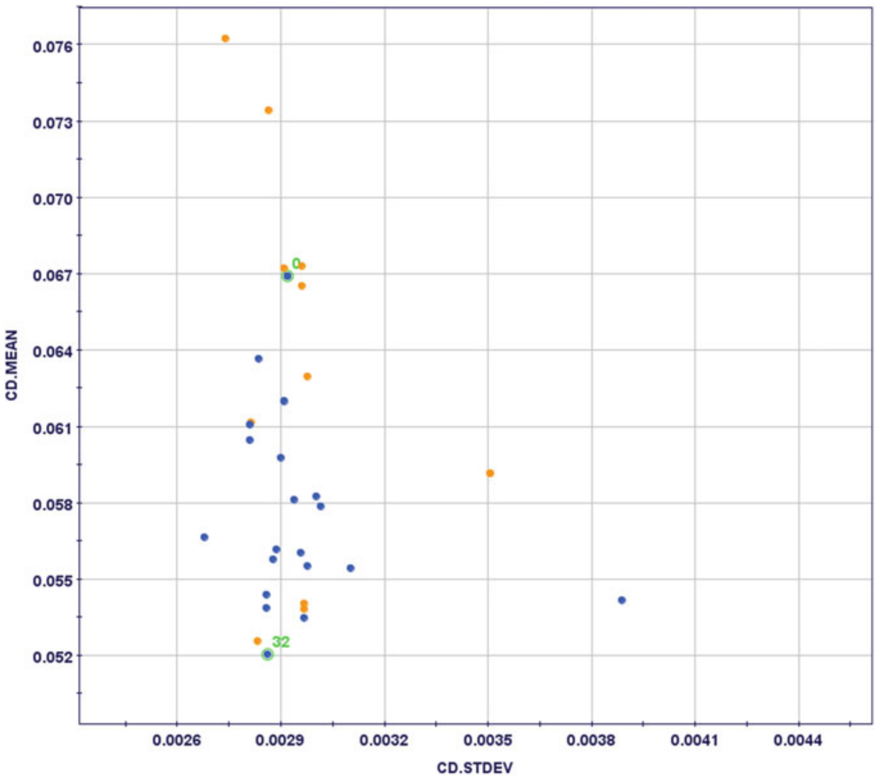


Fig. 5 Optimization results using min-max approach

Table 3 UQ results with different methodologies

Design	Mean Cd	STDV Cd	Mean Cl	STDV Cl	Mean Cm	STDVCm
Baseline	6.693E−2	2.918E−3	9.491E−1	1.575E−2	1.286E−1	6.237E−4
Optimized	5.207E−2	2.860E−3	9.423E−1	1.251E−2	1.127E−1	6.318E−4

To solve efficiently this single-objective optimization, we have applied a *Simplex* algorithm [13], with a global number of simulations not higher than the one considered for the multi-objective case. Figure 5 reports the results obtained, with same parameters (mean and standard deviation) reported in the axis; in green, baseline point and optimized point are highlighted.

The results are in this case much more satisfactory than the previous approach: the percentage of feasible designs is much higher than before, and after just few iterations a possible convergence trend is found, improving in an important way the baseline performance (similar standard deviation, but much lower mean value; see Table 3 for more details).

For the purpose of this comparison, we decided to stop the optimization after less than 40 designs, with a total number of CFD simulations equal to 370, which corresponds to about two weeks of analysis.

Table 2 reports the results obtained following the min-max approach, which are definitively satisfactory.

5 Reliability-Based RDO

We have pointed out in previous section that the main moments of the distribution of the performance of any design can be used to quantify its robustness, i.e. they can be used as criteria for a RDO problem (for instance, one could maximize the mean performance and minimize the standard deviation). Conversely, the min-max approach or more in general a Reliability-based Optimization (RBDO) problem, needs for the optimization criteria the definition of a reliability index or a failure probability. This approach can in fact be used, as noted above, to define accurately a min-max criteria (objective or constraint) also when the output performance is not necessarily of Normal type.

Many methodologies exist in literature to determine the failure probability, such as FORM/SORM [13], which for a RDO optimization could be very expansive from the numerical point of view. For this reason, we propose here a different methodology, based indeed on the Polynomial Chaos polynomial exploitation.

In fact, the evaluation of the performance function in industrial cases can be very demanding, since they often involve expensive CFD or structural numerical simulations. In the approach we propose, these expensive evaluations are required only to determine the coefficients of the PCE (Polynomial Chaos expansion). Once found them, it is possible to express the CDF (cumulative distribution function) of any system response using directly the PCE polynomial, which can be considered as a meta-model of the response, practically free in terms of CPU. Once the CDF is accurately obtained, from the given constraint value we can easily retrieve the corresponding percentage of the distribution, i.e. the *failure probability*.

In this way, a Robust Design Optimization problem can be defined, using as criteria for the optimization the minimization of the *failure probability*: in other words, we search for a new design whose failure probability for the given uncertainties distribution is minimum, either for a new design for which a given percentile (e.g. 99%) of its distribution is minimum. The big advantage of this approach with respect to using FORM/SORM methodologies is the reduced number of sampling points needed to obtain the Polynomial Chaos based meta-model, if compared to the iterations needed to compute the reliability index for each design required by FORM/SORM methodologies.

To validate the efficiency of the methodology proposed in this chapter, we have applied it to the same benchmark case used to describe the State of the Art techniques of RDO in previous section.

Table 4 RDO constraints accordingly to 6a and PCE-RBDO

Constraints	6 sigma definition	PCE-based
Constraint 1: min lift	mean_Cl – 3 * σ Cl = 0.90	0.03% = 0.887
Constraint 2: max drag	mean_Cd + 3 * σ_Cd = 0.0757	99.97% = 0.0747
Constraint 3: max momentum	mean_Cm + 3 * σ_Cm = 0.1305	99.97% = 0.1305

Table 5 Results of *Six-Sigma* based and RBDO method

Design	Cd-max	Cl-min	Cm-max
Baseline	7.47E−2	8.87E−1	1.302E−1
Optimized by 6sigma	*6.06E−2 **(6.33E−2)	*9.04E−1 **(8.94E−2)	*1.145E−1 **(1.129E−1)
Optimized by reliability	6.05E−2	8.95E−1	1.127E−1

In the previous case, we have approximated the needed percentile distributions of the performances (99.97%) by a *Six-Sigma* interval, that is however correct only under the hypothesis, not verifiable a priori, of a Normal distribution of the responses.

Following the new approach proposed, we can instead compute accurately the needed percentile distribution (99.97%) directly from the CDF distribution function defined by the Polynomial Chaos expansion, which is more accurate and valid also if the output distributions are generally different from a Normal distribution.

The definition of the constraints are therefore slightly different from the other approach, and Table 4 above reports the constraints values in the two cases.

In Table 5 we report a comparison of the performances (mean and standard deviation values) of the baseline configuration and of the optimized configuration obtained in each approach (also in this case *SIMPLEX* has been used).

The optimized solution is generally slightly different, considering the performance distributions, following the two approaches, but in both cases the constraints are respected and the selected objective is minimized.

Nevertheless, if in the first approach (*Six-Sigma*) the hypothesis followed is not necessarily correct (the performance distribution of the response does not necessarily follow a Normal distribution) and therefore not necessarily the 99.97% of the distribution really take the values estimated, by the new approach we can estimate with much more accuracy the needed percentile of the distributions, therefore we have can assume with an higher accuracy that the extreme values of the distributions take the values indicated, therefore respecting accurately the constraints.

To prove this assumption, we have re-evaluated the performances of the optimal design found by the *Six-Sigma* approach (* in Table 5), using this time the reliability approach, i.e. extracting the real 99.97%-ile value from the Polynomial Chaos expansion. These corrected values are reported inside brackets in the second row (** in Table 5). As we can note, the performances originally estimated by the *Six-Sigma* approach are in reality worse, and even though the differences are in this case not very large, the results obtained applying the reliability criteria for the whole optimization (third row of Table 5), i.e. following the new methodology proposed in this section, are better, in particular for what concerns the objective function (drag minimization: 6.05E−2 instead of 6.33E−2).

In conclusion, to solve with highest efficiency the *min-max* approach for a RDO problem, a reliability-based approach is needed, and the Polynomial Chaos Expansion approach here described has revealed to be the most efficient approach.

6 Conclusion

In this paper we have illustrated some innovative methodologies for the Robust Design Optimization with large number of uncertainties, which is a typical requirement from the industry.

Two different UQ methodologies have been proposed, one based on SS-ANOVA and one based on a step-wise regression methodology, which can be used to reduce the number of sampling points for an accurate uncertainty quantification (either reducing the number of significant parameters, or reducing the number of Polynomial Chaos terms).

In addition, a methodology for efficient Robust Design Optimization (based on the application of min-max criteria combined with reliability-based optimization formulation and Polynomial Chaos exploitation for percentiles estimation) has been presented.

All the methodologies have been validated by the application of selected test cases in aeronautical field; in the future steps of UMRIDA Project, the proposed methodologies will be applied to an industrial problem of challenging relevance.

Acknowledgements This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. ACP3-GA-2013-605036-UMRIDA.

References

1. Clarich, A., Pediroda, V.: Robust design applications with modeFRONTIER, applying NODESIM-CFD tools. In: NODESIM-CFD Workshop on Quantification of CFD Uncertainties, Bruxelles, 29–30 October 2009
2. Loeven, G., Witteveen, J., Bijl, H.: Probabilistic collocation: an efficient non-intrusive approach for arbitrarily distributed parametric uncertainties. In: 45th AIAA Aerospace Science Meeting and Exhibit, Reno, Nevada (2007)
3. Kalsi, M., Hacker, K., Lewis, K.: A comprehensive robust design approach for decision tradeoffs in complex systems design. J. Mech. Des. **123**(1), 1–10 (2001)
4. Clarich, A., Marchi, M., Russo, R.: Reliability-based optimization applying polynomial chaos expansion. In: 6th European Conference on Computational Fluid Dynamics (ECFD VI), July 20–25, 2014, Barcelona, Spain

5. Gu, C.: Smoothing Spline ANOVA Models. Springer, New York (2002)
6. Allen, D.M.: The relationship between variable selection and data augmentation and a method for prediction. Technometrics **16**, 125–127 (1974)
7. Blatman, G., Sudret, B.: Adaptive sparse polynomial chaos expansion based on Least Angle Regession. In Elsevier Science, June 2010
8. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. Ann. Stat. **32**, 407–499 (2004)
9. https://sites.google.com/a/numeca.be/umrida/home
10. Various Authors: Experimental data base for computer program assessment. Report of the Fluid Dynamics Panel Working Group, AGARD-AR-138 (1979). ISBN 92-835-1323-1
11. Farin, G.E., Hoschek, J., Kim, M.-S.: Handbook of Computer Aided Geometric Design, pp. 4–6. Elsevier (2002). ISBN 978-0-444-51104-1
12. Clarich, A., Pediroda, V., Periaux, J., Poloni, C.: Comparison between different game theory methodologies in robust design optimisation, Santorini, acts of Eccomas2005
13. Nelder, J.A., Mead, R.: A simplex method for function minimization. Comput. J. **7**, 308–313 (1965)

A Novel Method for Inverse Uncertainty Propagation

Xin Chen, Arturo Molina-Cristóbal, Marin D. Guenov, Varun C. Datta and Atif Riaz

Abstract Proposed is a novel method for inverse uncertainty propagation, ultimately aiming to facilitate the wider uncertainty allocation problem. The approach is enabled by techniques for the reversal of computational workflows and for the efficient propagation of uncertainty. The method is validated with analytical and numerical examples. Also a representative aircraft sizing code is used to illustrate the application in a more realistic setting.

1 Introduction

The need for competitive, robustly designed products with optimized margins has led to the establishment in recent decades of uncertainty quantification and management (UQ&M) as a field of its own right. Broadly, it includes identification of the sources of uncertainty, quantification and propagation of this uncertainty through the design computation, and analysis of its effects on the output variables of interest. UQ&M research has already addressed a number of theoretical and computational challenges, especially with regard to uncertainty propagation (e.g. method of moments, polynomial chaos, Monte-Carlo Simulation, etc.). These techniques have been widely adopted in practice [1].

One of the remaining challenges involves the lack of decision-support methods for handling pre-defined uncertainty associated with design objectives (targets) or constraints (requirements). In such cases the designer will need to know where to inversely allocate or specify desired uncertainty for the input variables or parameters,

X. Chen · A. Molina-Cristóbal · M. D. Guenov (✉) · V. C. Datta · A. Riaz
Cranfield University, Cranfield MK43 0AL, UK
e-mail: M.D.Guenov@cranfield.ac.uk

X. Chen
e-mail: xin.chen@cranfield.ac.uk

A. Molina-Cristóbal
e-mail: a.molinacristobal@cranfield.ac.uk

A. Riaz
e-mail: a.riaz@cranfield.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_21

to meet the allowable uncertainty in the outputs. This problem will be referred to as *Uncertainty Allocation* in this paper. There is relatively little published on this subject. A related issue can be found under the topic of 'tolerance allocation' in the manufacturing community, where the individual tolerances of components are decided on the basis of requirements for the overall assembly tolerance [2]. Lööf et al. [3, 4] have developed a method to find the optimal way of allocating tolerances considering the cost and quality loss. This method requires the explicit relationships between different tolerances (e.g. via partial derivatives). However from the design point of view, these relationships may not be directly available due to extensive usage of non-linear or even black-box models. Baumgärtel et al. [5] have developed a method to estimate the input uncertainties based on a given output uncertainty bound, which is referred to as *Inverse Uncertainty Propagation*. This method is based on a forward propagation technique which applies Gaussian processes as surrogates, and uses approximation approaches to calculate the statistical moments [6]. A considerable number of supporting (sampling) points are needed to train the surrogate, which may lead to relatively high computational cost, especially for complex problems. Another related but distinct subject is the *Inverse Problem*. Generally it refers to parameter identification or model reconstruction given existing data of the system response [7]. In the field of uncertainty analysis, it is used to determine the unknown distributions based on measurements or observations. Bayesian Approach and Maximum Likelihood estimation are widely used for this area and it is sometimes called *Inverse Uncertainty Quantification* [8].

In this context, the wider objective of our research as part the European project TOICA (Thermal Overall Integrated Conception of Aircraft) [9] has been to develop an efficient uncertainty allocation method. This should enable the designer to quickly assess how much uncertainty is accepted from different sources to guarantee the results' variance(s). In this paper we deal specifically with the efficient inverse uncertainty propagation problem as an essential enabler. The rest of the paper is organized as follows: In Sect. 2, the problem definition is discussed in more detail. The enablers and proposed methodology are explained in Sects. 3 and 4, respectively. Two simple test-cases are used for validation in Sect. 5. In Sect. 6, a representative aircraft sizing code is used to demonstrate a more practical application of the approach. Conclusions and future work are presented in Sect. 7.

2 Problem Definition

In this paper, the scope is restricted (but not limited) to the model-based design approach where a set of models are used to deduce the design variables and evaluate the performance. The models are assembled as a *computational workflow* (shown in Fig. 1 Left) with its inputs specified by the designer and all dependent variables calculated automatically as the outputs [10, 11].

For a deterministic case, the outputs will be known once all the inputs are given fixed values. In reality however, uncertainty may be induced mainly from three sources:

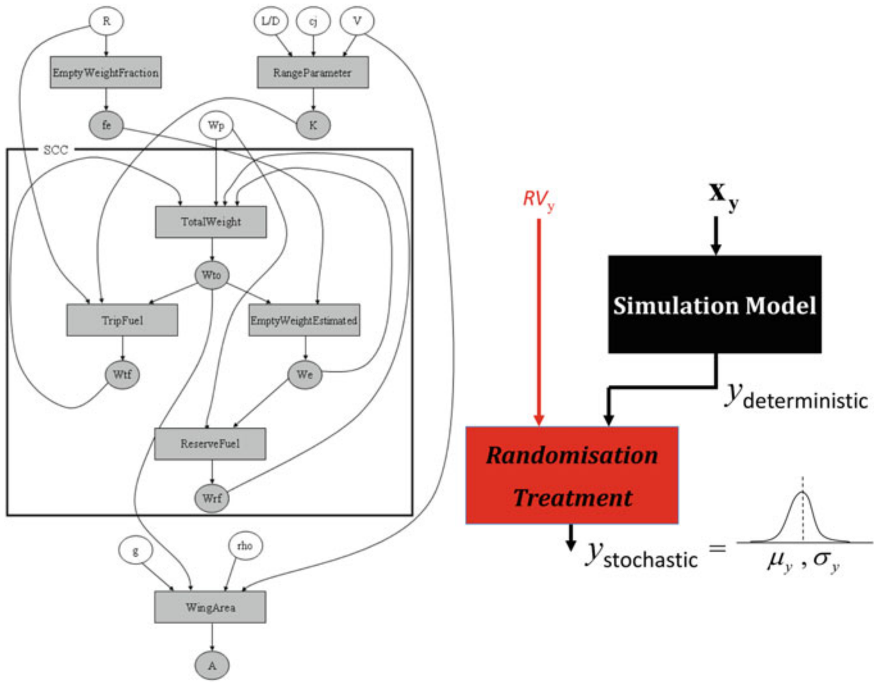


Fig. 1 Left: Example of a simple aircraft sizing workflow [11], Right: Randomization treatment to capture the model uncertainty [12]

1. The input variables may not be strictly defined, which means that they may vary within a certain range.
2. The models may not be fully calibrated; therefore the results produced may contain errors from the true values.
3. The computational code may introduce numerical uncertainty (e.g. noise).

In this work the focus is on the first two types of uncertainty. Here a probability approach is used; and uncertainty will be represented by statistical moments, in particular, standard deviations (*std*) of the input and output variables. Regarding model uncertainty, a method has been developed by Molina-Cristóbal et al. [12], where the output of a deterministic model is assigned with a random variable which makes the result varying according to a specified probability distribution (Fig. 1 Right); therefore the model uncertainty can be represented by the *std* of the associated random variable.

Mathematically this random variable is equivalent to an uncertain input. In this way, the problem could be defined as shown in Fig. 2. Assuming a workflow F with p inputs, m random variables (representing model uncertainty), and q outputs.

$$\mathbf{Y} = F(\mathbf{X}, \mathbf{RV}) \tag{1}$$

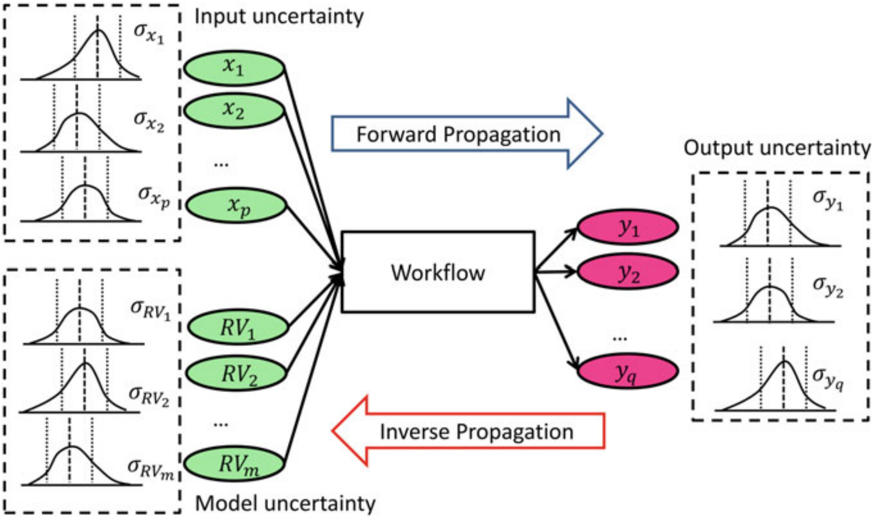


Fig. 2 Forward and Inverse uncertainty propagation

Here **X**, **Y** and **RV** are vectors of inputs, outputs and random variables respectively.

$$\begin{aligned}\mathbf{X} &= (x_1, x_2, ..., x_p)\\ \mathbf{Y} &= (y_1, y_2, ..., y_q)\\ \mathbf{RV} &= (RV_1, RV_2, ..., RV_m)\end{aligned} \tag{2}$$

Their uncertainties are represented by the associated *std* respectively.

$$\begin{aligned}\boldsymbol{\sigma}_{\mathbf{X}} &= (\sigma_{x_1}, \sigma_{x_2}, ..., \sigma_{x_p})\\ \boldsymbol{\sigma}_{\mathbf{Y}} &= (\sigma_{y_1}, \sigma_{y_2}, ..., \sigma_{y_q})\\ \boldsymbol{\sigma}_{\mathbf{RV}} &= (\sigma_{RV_1}, \sigma_{RV_2}, ..., \sigma_{RV_m})\end{aligned} \tag{3}$$

In forward propagation, $\boldsymbol{\sigma}_{\mathbf{Y}}$ is calculated based on pre-defined $\boldsymbol{\sigma}_{\mathbf{X}}$ and $\boldsymbol{\sigma}_{\mathbf{RV}}$. Regarding the inverse uncertainty propagation problem, we will set desired values for $\boldsymbol{\sigma}_{\mathbf{Y}}$ (or part of the vector $\boldsymbol{\sigma}_{\mathbf{Y}}$), and inversely propagate through the workflow to find out the proper values of $\boldsymbol{\sigma}_{\mathbf{X}}$ and $\boldsymbol{\sigma}_{\mathbf{RV}}$ (or part of $\boldsymbol{\sigma}_{\mathbf{X}}$ and $\boldsymbol{\sigma}_{\mathbf{RV}}$) which can produce the targeted $\boldsymbol{\sigma}_{\mathbf{Y}}$.

3 Enablers

Before presenting our proposed methodology, we briefly describe two essential enablers that we have developed in previous work.

3.1 *Univariate Reduced Quadrature (URQ) Method*

URQ is a forward uncertainty propagation method based on the quadrature approach [13]. Compared with Monte-Carlo Simulation (MCS), it enables a much faster and computationally cheaper estimation of the outputs' uncertainty with lower, but still comparable accuracy.

To construct a URQ propagation, the user needs to specify the first four moments: Mean, Standard Deviation, Skewness and Kurtosis (μ_x, σ_x, γ_x, Γ_x respectively) for each stochastic input variable. URQ will then select $2p + 1$ sampling points to be evaluated, where p is the number of the uncertain input variables. If m random variables representing model uncertainty are also considered, then the total number of sampling point should be $2(p + m) + 1$. Based on the $2p + 1$ (or $2(p + m) + 1$) evaluation results, the method approximates the means and variances of the stochastic outputs. More details on the derivation of this method are given by Padulo et al. [13].

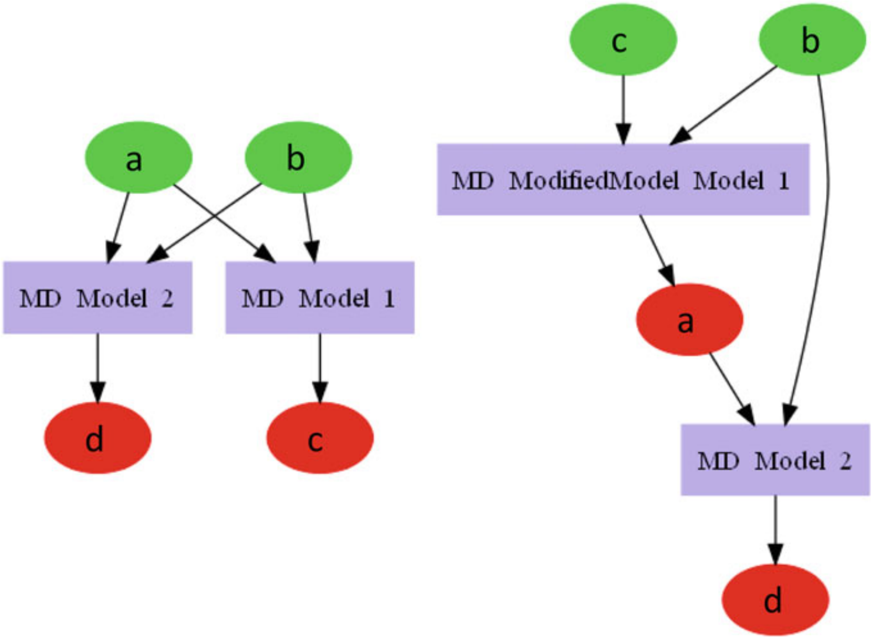


Fig. 3 Example of workflow reversal: original workflow (left) and reversed workflow (right)

3.2 Workflow Reversal

Reversal refers to the capability to swap input and output variables of a workflow. A simple example is shown in Fig. 3, where one of the original (default) inputs: a is swapped with one of the original (default) outputs: c. This technique has been implemented in a prototype tool called AirCADia [14, 15], where the user is able to define such a reversal by specifying the variables he or she wants to swap, and consequently the new reversed workflow is created. This capability allows the designer to ask what-if questions and conduct flexible computational studies. The reader is referred to ref [10, 11, 16, 17] for more details of this capability.

4 Method for Inverse Propagation

Recall the objective of this research: to calculate input and model uncertainty (via their associated std) given the specified std of the selected outputs. The proposed method is comprised of three steps:

1. The first step (optional) is to rank the contribution from different sources of uncertainty, by conducting Global Sensitivity Analysis (GSA) [18]. Ultimately this step is aimed at reducing the number of variables involved in the subsequent steps. Variance-based GSA techniques [19] are preferred for their adaptability to different models (especially black-box models). Such techniques include the Sobol' Indices method [20–22], Fourier Amplitude Sensitivity Test (FAST) [23, 24] etc.
2. The second step involves the creation of an *'Outer Workflow'* which accommodates the standard deviations as parameters. The original workflow and URQ method are embedded inside the outer workflow which takes the four moments of the original inputs as additional inputs, and the two moments of the outputs as additional outputs. Without loss of generality, Fig. 4 shows a workflow with two inputs and two outputs. The Outer workflow takes μ_{x_1}, σ_{x_1}, γ_{x_1}, Γ_{x_1}, μ_{x_2}, σ_{x_2}, γ_{x_2}, Γ_{x_2} as the new inputs which are passed to the URQ method to generate sampling points for repeated executions of the original workflow. Still through the URQ method, evaluation results are used to calculate μ_{y_1}, σ_{y_1}, μ_{y_2}, σ_{y_2} as the new output variables.
3. The third step utilises AirCADia's workflow reversal capability to swap the inputs' standard deviations with those of the outputs. That is, the user assigns the desired values and executes the reversed workflow in AirCADia. The workflow reversal in this step is illustrated in Fig. 5. By reversing the Outer Workflow, σ_{y_1}, σ_{y_2} have now become the inputs and σ_{x_1}, σ_{x_2} have been swapped to become the outputs. The user could set values for σ_{y_1}, σ_{y_2} and calculate σ_{x_1}, σ_{x_2} as the results of the reversed workflow.

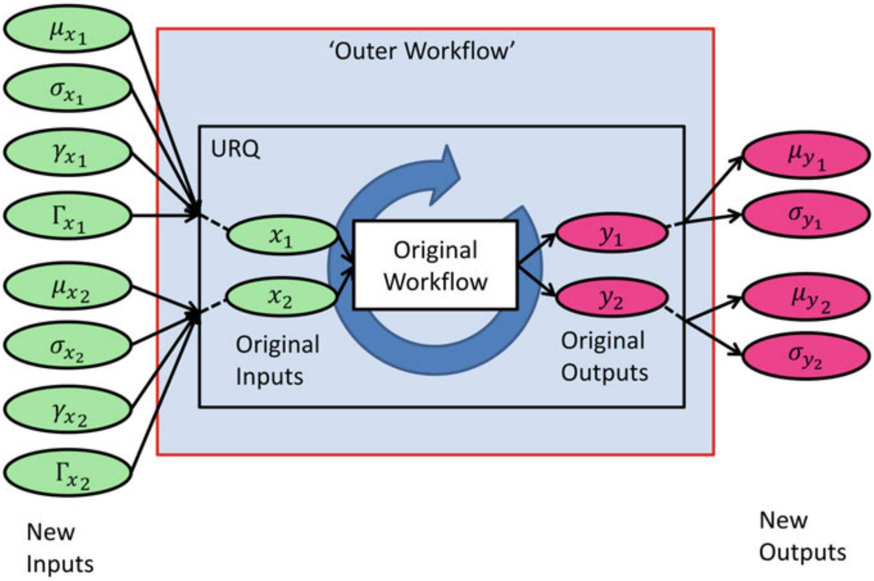


Fig. 4 Set up of an outer workflow

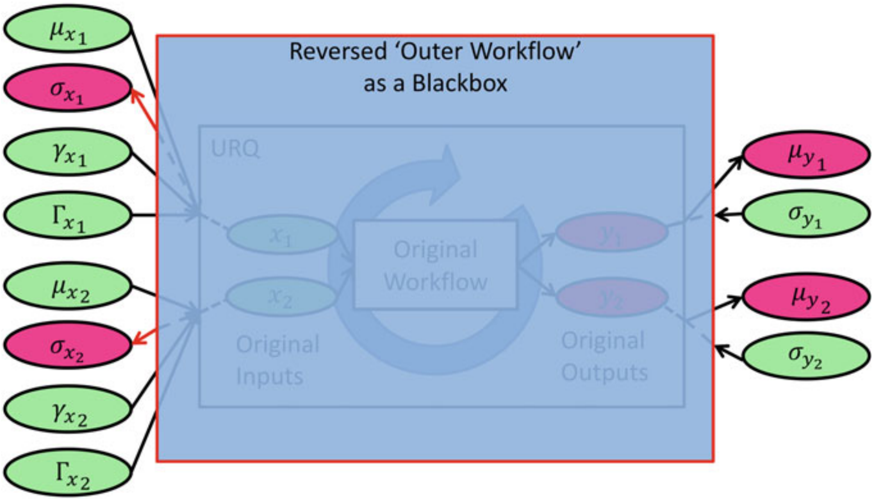


Fig. 5 Reversal of an outer workflow

5 Validation

The methodology proposed in the previous section is tested with progressively more complex examples.

5.1 Test Case 1: Linear Functions

Assuming a simple workflow composed of two equations,

$$\begin{aligned} y_1 &= x_1 + 2x_2 \\ y_2 &= 3x_1 - 4x_2 \end{aligned} \tag{4}$$

x_1 and x_2 are the two input variables which follow the normal distribution independently.

$$\begin{aligned} x_1 &\sim N(\mu_{x_1}, \sigma_{x_1}^2) \\ x_2 &\sim N(\mu_{x_2}, \sigma_{x_2}^2) \end{aligned} \tag{5}$$

Based on the linear combination of independent normal distributions, it could be analytically obtained that,

$$\begin{aligned} y_1 &\sim N(\mu_{x_1} + 2\mu_{x_2}, \sigma_{x_1}^2 + 4\sigma_{x_2}^2) \\ y_2 &\sim N(3\mu_{x_1} - 4\mu_{x_2}, 9\sigma_{x_1}^2 + 16\sigma_{x_2}^2) \end{aligned} \tag{6}$$

Therefore,

$$\begin{aligned} \sigma_{x_1} &= \sqrt{\frac{\sigma_{y_2}^2 - 4\sigma_{y_1}^2}{5}} \\ \sigma_{x_2} &= \sqrt{\frac{9\sigma_{y_1}^2 - \sigma_{y_2}^2}{20}} \\ 2\sigma_{y_1} &\leq \sigma_{y_2} \leq 3\sigma_{y_1} \end{aligned} \tag{7}$$

We tested a series of targeted values for σ_{y_1} and σ_{y_2}. The theoretical values for σ_{x_1} and σ_{x_2} are listed in Table 1 compared with those calculated by the proposed method. It should be mentioned that although μ_{x_1} and μ_{x_2} do not influence the standard deviations, we need to set their values to execute the workflow. Here they are all set as 1. From Table 1, it could be seen that the proposed method is able to provide considerable accuracy for such linear models.

Table 1 Results from testcase 1 (T stands for Theoretical values and A stands for Actual values)

σ_{y_1}	σ_{y_2}	$\sigma_{x_1}(T)$	$\sigma_{x_1}(A)$	$\sigma_{x_2}(T)$	$\sigma_{x_2}(A)$
0.05	0.12	0.0296648	0.0296666	0.0201246	0.0201254
0.7	1.7	0.4312772	0.4312772	0.2756810	0.2756810
1	2.5	0.6708204	0.6708204	0.3708099	0.3708099
2	5	1.3416408	1.3416407	0.7416198	0.7416199
3	8	2.3664319	2.3664319	0.9219544	0.9219544

5.2 Test Case 2: Non-Linear Functions

$$\begin{aligned} y_3 &= x_3^2 + e^{x_4} \\ y_4 &= \sin x_3 + x_4 \end{aligned} \tag{8}$$

x_3 and x_4 are the two input variables which follow the uniform distribution independently.

$$\begin{aligned} x_3 &\sim U(l_{x_3}, u_{x_3}) \\ x_4 &\sim U(l_{x_4}, u_{x_4}) \end{aligned} \tag{9}$$

To simplify the calculation process, we assume that l_{x_3}, u_{x_3}, l_{x_4} and u_{x_4} are all positive; also $(l_{x_3}, u_{x_3}) \in [0, \pi]$. By solving $E((y_i - \mu_{y_i})^2), i = 3, 4$, the analytical relationship between σ_{x_3}, σ_{x_4} and σ_{y_3}, σ_{y_4} is obtained:

$$\begin{aligned} \sigma_{y_3}^2 &= \frac{(\mu_{x_3} + \sqrt{3}\sigma_{x_3})^5 - (\mu_{x_3} - \sqrt{3}\sigma_{x_3})^5}{10\sqrt{3}\sigma_{x_3}} \\ &\quad - \frac{[(\mu_{x_3} + \sqrt{3}\sigma_{x_3})^3 - (\mu_{x_3} - \sqrt{3}\sigma_{x_3})^3]^2}{108\sigma_{x_3}^2} \\ &\quad + \frac{e^{2\mu_{x_4}}(e^{2\sqrt{3}\sigma_{x_4}} - \frac{1}{e^{2\sqrt{3}\sigma_{x_4}}})}{4\sqrt{3}\sigma_{x_4}} - \frac{e^{2\mu_{x_4}}(e^{\sqrt{3}\sigma_{x_4}} - \frac{1}{e^{\sqrt{3}\sigma_{x_4}}})^2}{12\sigma_{x_4}^2} \\ \sigma_{y_4}^2 &= \frac{\sin(2\mu_{x_3} - 2\sqrt{3}\sigma_{x_3}) - \sin(2\mu_{x_3} + 2\sqrt{3}\sigma_{x_3})}{8\sqrt{3}\sigma_{x_3}} \\ &\quad - \frac{[\cos(\mu_{x_3} - \sqrt{3}\sigma_{x_3}) - \cos(\mu_{x_3} + \sqrt{3}\sigma_{x_3})]^2}{12\sigma_{x_3}^2} + \sigma_{x_4}^2 + \frac{1}{2} \end{aligned} \tag{10}$$

Here σ_{x_3}, σ_{x_4}, μ_{x_3}, and μ_{x_4} are calculated from l_{x_3}, u_{x_3}, l_{x_4}, and u_{x_4}.

$$\mu_{x_i} = \frac{l_{x_i} + u_{x_i}}{2}, \sigma_{x_i} = \frac{u_{x_i} - l_{x_i}}{2\sqrt{3}}, i = 3, 4 \tag{11}$$

Table 2 Results from test case 2 (T stands for Theoretical values and A stands for Actual values)

μ_{x3}	μ_{x4}	σ_{y3}	σ_{y4}	$\sigma_{x3}(T)$	$\sigma_{x3}(A)$	$\sigma_{x4}(T)$	$\sigma_{x4}(A)$
1	0.7	1.3365169	0.4803470	0.5	0.50056679	0.4	0.39962203
1.2	1	1.1466395	0.2514633	0.4	0.40020410	0.2	0.19975415
1.4	1.2	2.7178940	0.5461466	0.7	0.70569482	0.5	0.49625501
1.6	0.8	1.7575017	0.3184329	0.5	0.50022576	0.3	0.29935355
2	1.5	3.6073339	0.5755963	0.65	0.65359010	0.5	0.49792451
1.7	3	17.1103914	0.7187758	0.6	0.56182699	0.7	0.70320786
1.3	1.6	3.3116937	0.5587234	0.7	0.76526044	0.5	0.47931307
1.7	2	6.3964317	0.7050625	0.4	Not converged	0.7	Not converged

Since it is difficult to give analytical solutions of these equations, we can firstly set σ_{x_3}, σ_{x_4} to calculate σ_{y_3}, σ_{y_4} from the equations and then pass the values of σ_{y_3}, σ_{y_4} to the inverse propagation method to check if the same σ_{x_3}, σ_{x_4} could be achieved. In this case, μ_{x_3} and μ_{x_4} will also have an impact on the values of σ_{y_3} and σ_{y_4}. The settings are shown in the Table 2.

As could be seen from Table 2, while most of the settings produce satisfactory results, the last three groups show relatively lower accuracy. For the last one, the reversal solver did not converge to the theoretical value. By running a sensitivity analysis, it is noted that for these three settings, the output uncertainty is very sensitive to σ_{x_4} due to the relatively large value of μ_{x_4}. This might create more difficulty for the solver to find a solution. Future work will involve determining the limitations of the solver and proposing adequate solvers for highly non-linear problems.

6 Industrial Testcase

6.1 Testcase Setup

To demonstrate a practical application, we applied the method to an aircraft sizing code, USMAC (Ultra Simplified Model of Aircraft), which was provided by an industrial partner, in the context of the European project VIVACE [25]. Figure 6 shows the workflow of USMAC, where green and red ovals represent input and output variables respectively. Models are represented as purple boxes.

Here we consider five sources of uncertainty shown in Table 3. Among them, the Bypass Ratio (BPR), Sea Level Static Thrust (FNslst) and Temperature at Cruise (Tamb_{crz}) belong to the input uncertainty, which are caused by non-fixed design variables or changing working conditions. The uncertainty of the Drag model and SFC model is treated using the method mentioned in Sect. 2.

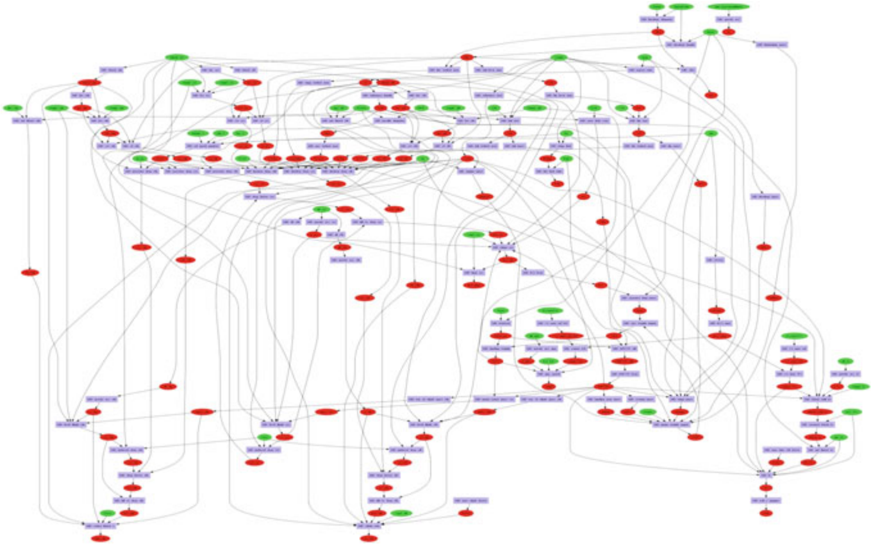


Fig. 6 Workflow of USMAC

Table 3 Sources of uncertainty

Source	Type	Distribution	Standard deviation
Bypass ratio	Input uncertainty	Uniform	0.2886
		$l_{BPR} = 7.5$	
		$u_{BPR} = 8.5$	
Sea level static thrust (N)	Input uncertainty	Uniform	3752.7767
		$l_{FNslst} = 123500$	
		$u_{FNslst} = 136500$	
Temperature at cruise (K)	Input uncertainty	Normal	10.9000
		$\mu_{Tamb_{crz}} = 218$	
		$\sigma_{Tamb_{crz}} = 10.9$	
Drag model	Model uncertainty	Normal	0.0500
		$\mu_{RV_{drag}} = 1$	
		$\sigma_{RV_{drag}} = 0.05$	
SFC model	Model uncertainty	Triangular	0.0122
		$a_{RV_{SFC}} = 0.97$	
		$b_{RV_{SFC}} = 1.03$	
		$c_{RV_{SFC}} = 1$	

Table 4 Results from forward uncertainty propagation

Output variables	Mean	*std*
RA (km)	10003.3178	511.7080
MTOW (kg)	104164.1357	274.6514

6.2 Forward Propagation

The allocation of uncertainty to the outputs of a design computation (e.g. objectives, constraints) can be based on various considerations, including the experience of the analyst. In this case we use the following scenario, leading to the assignment of targets for the output uncertainty.

First, we assume that a Monte Carlo simulation (MCS) was run *a priori* to establish the means and standard deviations of two critical output variables: Range (RA) and Maximum Take-off Weight (MTOW) with no restriction on their values. The results are shown in Table 4.

Second, we now assume that we wish to place constraints on RA to be larger than 9500 km and on MTOW to be lower than 104460 kg. The inferred probabilities of meeting these constraints are 84.357% and 83.887%, respectively (Fig. 7). These are calculated from the ratio of sampling points meeting the requirements to the total sampling points of the MCS. If we wished to increase both probabilities, for example, up to 90%, we need to reduce the output uncertainty, which in this case is represented by standard deviations. Padulo and Guenov [26] developed a method for mapping the expected probability into desired standard deviations. With this method the target *std* of outputs are estimated as shown in Table 5.

6.3 Inverse Propagation

As discussed in Sect. 4, and following the assignment (allocation) of output uncertainty to MTOW and RA, we start with the first (optional) step of the method, sensitivity analysis, to identify the most influential uncertainty sources. In this particular case FAST is utilised via a MATLAB toolbox developed by Cannavó [27].

According to the results shown in Fig. 8 we could determine that the uncertainties in input variable: FNslst, and Drag model are the main contributors to the uncertainties in outputs. Now a new outer workflow is set up which includes the standard deviations as variables. This outer workflow is then reversed with σ_{MTOW} and σ_{RA} as inputs; σ'_{FNslst} and $\sigma'_{RV_{drag}}$ as outputs. With the desired values in Table 5, we execute this reversed outer workflow which calculates σ'_{FNslst} and $\sigma'_{RV_{drag}}$ as 2924.5705 and 0.0236, respectively. The results indicate that, to achieve the target *std* in Table 5, σ_{FNslst} should be lower than 2924.5705 and $\sigma_{RV_{drag}}$ should be lower than 0.0236. This process is illustrated in Fig. 9.

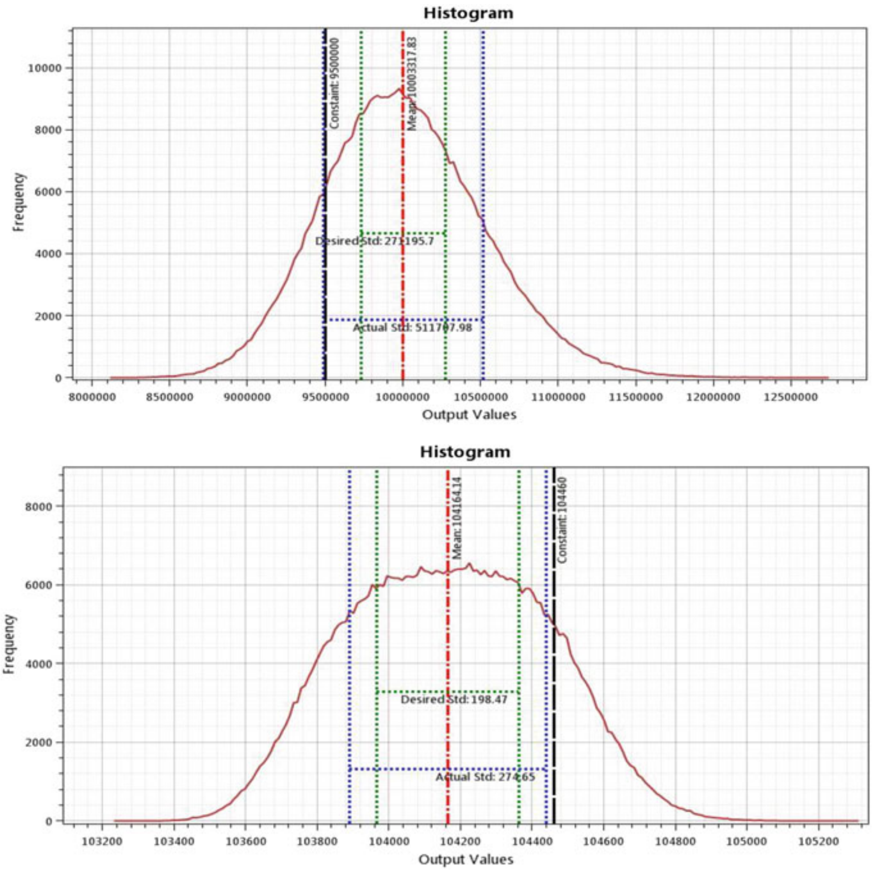


Fig. 7 Histogram plot of RA (upper) and MTOW (lower). Black lines represent for constraints; blue and green lines represent the actual and desired output *std* respectively

Table 5 Original and target standard deviations of the outputs

Output variables	Original *std*	Target *std*
RA (km)	511.7080	271.1957
MTOW (kg)	274.6514	198.4718

To demonstrate the efficiency, we also conducted a benchmarking study, where a 'brute-force' strategy, shown in Fig. 10 was adopted.

In this strategy, Monte-Carlo simulation is implemented in an optimization loop, in which the input *std* are updated by the solver repeatedly to minimize the gap between actual output *std* and the target ones. This setup was conducted in MATLAB environment with its built-in solver: *fminsearch*. The computational cost is the number of the optimization iterations (limited up to 400) multiplied by the number

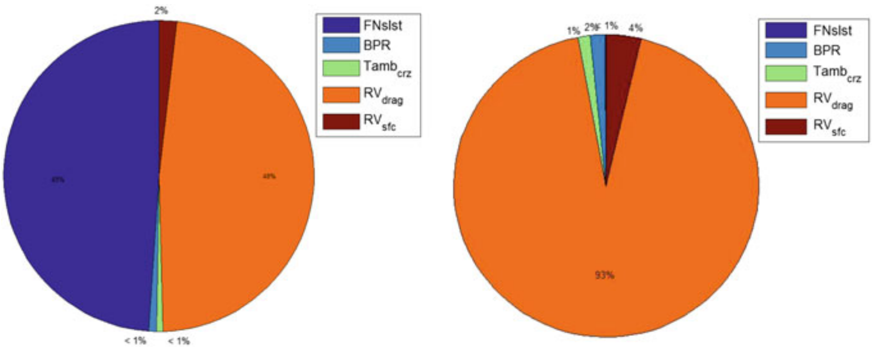


Fig. 8 First order sensitivity indices of different inputs for MTOW (left) and RA (right)

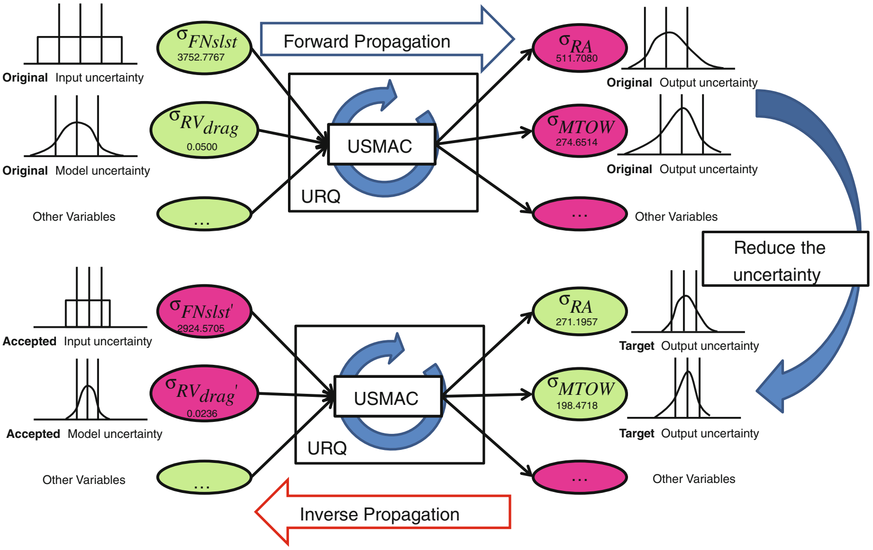


Fig. 9 Overview of the process in the testcase

of MCS runs in each loop. We tested a number of different settings and the results are shown in Table 6.

For validation, the new values of σ'_{FNslst} and $\sigma'_{RV_{drag}}$ are put back into Monte-Carlo forward propagation (1000000 runs) to check if the desired output uncertainty can be achieved. The results are shown in Table 7.

It could be seen that the proposed method provides not only better efficiency, but in this particular case also higher accuracy. By definition the 'brute-force' method should have achieved a higher accuracy. One possible explanation which needs a further investigation is that by using the URQ, the problem is smoothened. URQ could be taken as a deterministic approach of calculating the statistical moments

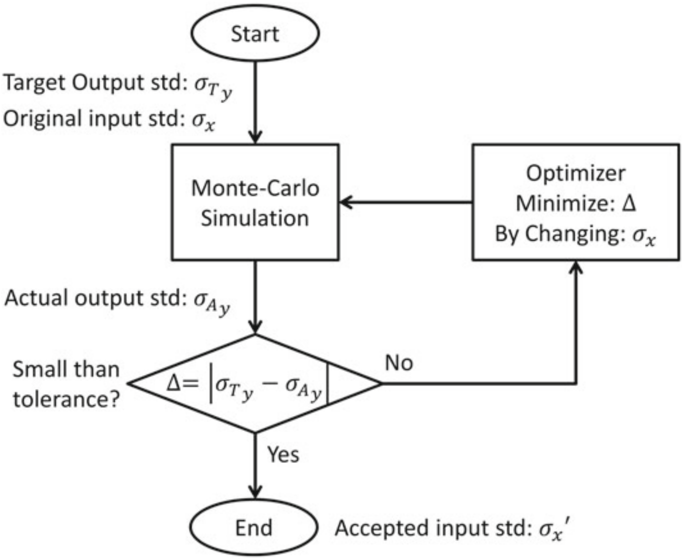

Start
Target Output std: σ_{T_y}
Original input std: σ_x
Monte-Carlo Simulation
Optimizer
Minimize: Δ
By Changing: σ_x
Actual output std: σ_{A_y}
Small than tolerance?
$\Delta = |\sigma_{T_y} - \sigma_{A_y}|$
No
Yes
End
Accepted input std: $\sigma_x{}'$

Fig. 10 A brute-force’ strategy to realize inverse propagation

Table 6 Comparison between different settings for ‘brute-force’ strategy and the proposed method

Test no.	Num of simulations in each opt. iteration	Total time (s)	σ'_{FNslst}	$\sigma'_{RV_{drag}}$
1	MCS: 100	Solver not converged		
2	MCS: 1000	45.94	2905.7106	0.0229
3	MCS: 10000	428.33	2900.8375	0.0236
4	MCS: 20000	871.59	2896.805	0.0237
5	MCS: 30000	1312.87	2918.4910	0.0234
6[a]	URQ: 11	3.57	2924.5705	0.0236

[a] Proposed Method in AirCADia

Table 7 Validation of the results

Test no.	Actual[a] σ_{MTOW}	Error (%)	Actual[b] σ_{RA}	Error (%)
1	N.A	N.A	N.A	N.A
2	196.9392	−0.7722	265.4866	−2.105
3	197.1362	−0.6730	271.7015	0.1865
4	196.8078	−0.8384	272.0898	0.3297
5	198.1202	−0.1772	270.0255	−0.4315
6	198.6092	0.0692	271.4027	0.0763

[a] Target σ_{MTOW} is 198.4718
[b] Target σ_{RA} is 271.1957

of the outputs. Once the four moments of the inputs are specified, the sampling points are 'decided' accordingly, therefore the outputs' means and variances could be calculated as fixed values. On the other hand, MCS are based on random sampling points, the means and variances been calculated are still random variables, which may be different every time running the whole MCS (although they will converge to the true values as the number of sampling points increases). This limits the performance of the solver. (Actually the limitation of 400 optimization iterations was reached in each of the tests 1–5). Accuracy might be improved by refining the solver settings which is beyond the scope of this research.

7 Conclusion and Future Work

Presented in this paper is a novel method for inverse propagation of uncertainty. It incorporates enablers for the reversal of computational workflows, and for the efficient propagation of uncertainty. This contributes towards the identification and allocation of desired variances to design inputs. The methodology is illustrated with representative analytical and numerical examples. The results demonstrate its accuracy and efficiency.

Future work will investigate further the robustness of the reversal process and the limitations of many-to-many reversals (regarding the existence of a solution). Also the scalability of the proposed method with regard to application to the wider uncertainty allocation problem will be investigated.

Acknowledgements The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2013-2016, TOICA project) under grant agreement $n°$ 604981.

References

1. Mangeant, F.: Joined initiatives around uncertainty management. Ann. Telecommun.-ann. des téléCommun. **66**(7–8), 397–407 (2011)
2. Chase, K.W.: Tolerance allocation methods for designers. ADCATS Rep. **99**(6), 1–28 (1999)
3. Lööf, J., Hermansson, T., Söderberg, R.: An efficient solution to the discrete least-cost tolerance allocation problem with general loss functions. In: Davidson, J.K. (ed.) Models for Computer Aided Tolerancing in Design and Manufacturing. 9th CIRP International Seminar on Computer-Aided Tolerancing, Arizona, Apr 2005, p. 115. Springer, Netherlands (2007)
4. Lööf, J., Söderberg, R.: Discrete Tolerance Allocation for Product Families. Eng. Optim. **44**(1), 75–85 (2012)
5. Baumgärtel, P., Endler, G., Wahl, A.M., Lenz, R.: Inverse uncertainty propagation for demand driven data acquisition. In: Tolk, A., Yilmaz, L., Diallo, S.Y., Ryzhov, I.O. (eds.) Proceedings of the 2014 Winter Simulation Conference, Georgia, December 2014, p. 710. IEEE Press, Piscataway (2014)

6. Girard, A., Murray-Smith, R.: Gaussian processes: Prediction at a noisy input and application to iterative multiple-step ahead forecasting of time-series. In: Murray-Smith, R., Shorten, R. (eds.) Switching and Learning in Feedback Systems. European Summer School on Multi-agent Control, Maynooth, September 2003. Lecture Notes in Computer Science (Revised Lectures and Selected Papers), vol. 3355, p. 158. Springer, Heidelberg (2005)
7. Tarantola, A.: Inverse problem theory and methods for model parameter estimation. Soc. Ind. Appl. Math. (2005). https://doi.org/10.1137/1.9780898717921
8. Litvinenko, A., Matthies, H.G.: Inverse problems and uncertainty quantification. arXiv:1312.5048v2 (2014). Accessed 06 July 2015
9. Thermal Overall Integrated Conception of Aircraft. http://www.toica-fp7.eu/ (2015). Accessed 06 July 2015
10. Balachandran, L.K., Guenov, M.D.: Computational workflow management for conceptual design of complex systems. J. Aircr. **47**(2), 699–703 (2010)
11. Nunez, M., Datta, V.C., Molina-Cristóbal, A., Guenov, M.D., Riaz, A.: Enabling exploration in the conceptual design and optimisation of complex systems. J. Eng. Des. **23**(10–11), 852–875 (2012)
12. Molina-Cristóbal, A., Nunez, M., Guenov, M.D., Laudan, T., Druot, T.: Black-box model epistemic uncertainty at early design stage, an aircraft power-plant integration case study. In: 29th Congress of the International Council of the Aeronautical Sciences, St Petersburg, 7–12 Sept 2014
13. Padulo, M., Campobasso, M.S., Guenov, M.D.: Novel uncertainty propagation method for robust aerodynamic design. AIAA J. **49**(3), 530–543 (2011)
14. Guenov, M.D., Nunez, M., Molina-Cristóbal, A., Datta, V.C., Riaz, A.: Aircadia—an interactive tool for the composition and exploration of aircraft computational studies at early design stage. In: 29th Congress of the International Council of the Aeronautical Sciences, St Petersburg, 7–12 Sept 2014
15. Guenov, M.D., Nunez, M., Molina-Cristóbal, A., Sripawadkul, V., Datta, V.C., Riaz, A.: Composition, management, and exploration of computational studies at early design stage. In: Vasile, M., Becerra, V.M. (eds.) Computational Intelligence in Aerospace Sciences, pp. 415–460. Progress in Astronautics and Aeronautics, American Institute of Aeronautics and Astronautics, Virginia (2014)
16. Balachandran, L.K.: Computational workflow management for conceptual design of complex systems: an air-vehicle design perspective. Dissertation, Cranfield University (2007)
17. Datta, V.C.: Interactive computational model-based design: a blackbox perspective. Dissertation, Cranfield University (2014)
18. Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S.: Global Sensitivity Analysis: the Primer. Wiley, Chichester (2008)
19. Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., Tarantola, S.: Variance based sensitivity analysis of model output: design and estimator for the total sensitivity index. Comput. Phys. Commun. **181**(2), 259–270 (2010)
20. Sobol', I.M.: Sensitivity analysis for nonlinear mathematical models (in english). Math. Model. Comput. Exp. **1**, 407–414 (1993)
21. Sobol', I.M.: Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. Math. Comput. Simul. **55**(1), 271–280 (2001)
22. Saltelli, A.: Making best use of model evaluations to compute sensitivity indices. Comput. Phys. Commun. **145**(2), 280–297 (2002)
23. Cukier, R.I., Fortuin, C.M., Shuler, K.E., Petschek, A.G., Schaibly, J.H.: Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. I theory. J. Chem. phys. **59**(8), 3873–3878 (1973)
24. Schaibly, J.H., Shuler, K.E.: Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. II applications. J. Chem. Phys. **59**(8), 3879–3888 (1973)
25. VIVACE: Value improvement through a virtual aeronautical collaborative enterprise. https://cordis.europa.eu/result/rcn/47814_en.html. (2015). Accessed 23 April 2018

26. Padulo, M., Guenov, M.D.: Worst-case robust design optimization under distributional assumptions. Int. J. Numer. Methods Eng. **88**(8), 797–816 (2011)
27. Cannavó, F.: Sensitivity analysis for volcanic source modeling quality assessment and model selection. Comput. Geosc. **44**, 52–59 (2012)

Uncertainty Sources in the Baseline Configuration for Robust Design of a Supersonic Natural Laminar Flow Wing-Body

Domenico Quagliarella and Emiliano Iuliano

Abstract An aerodynamic configuration of a supersonic business jet wing-body is proposed as baseline for a robust aerodynamic shape design problem. This configuration has been analyzed to identify the main dependencies and interactions of the parameters that describe the uncertainty sources in the robust design problem. Subsequent steps of the research activity will be related to the robust natural laminar flow design optimization of this configuration.

1 Introduction

This work is related to a research activity carried out within the framework of the UMRIDA (Uncertainty Management for Robust Industrial Design in Aeronautics) research project that has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement n° ACP3-GA-2013-605036.

Within this project an aerodynamic configuration of a supersonic business jet wing-body was defined and proposed as baseline for a robust aerodynamic shape design problem. This configuration, while being of industrial interest, is not covered by copyright or confidentiality clauses and can be used for benchmarks and comparisons even outside of the UMRIDA consortium.

Here a preliminary parametric analysis has been performed to identify, using the Analysis of Variance (ANOVA), the main dependencies and interactions of the parameters that describe the uncertainty sources in the robust design problem. In the subsequent steps of the research activity a robust design will be performed whose setup will be determined through the results of this preliminary analysis.

D. Quagliarella (✉) · E. Iuliano (✉)
Fluid Mechanics Department, Multidisciplinary Analysis and Optimization Group,
CIRA—Italian Aerospace Research Centre, Via Maiorise, 81043 Capua, Italy
e-mail: d.quagliarella@cira.it

E. Iuliano
e-mail: e.iuliano@cira.it

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_22

For the sake of completeness the baseline description and the design problem definition, available in the database of the UMRIDA project, are first of all reported.

2 Design Problem Description

The reference configuration is the optimized wing–body shape produced by CIRA within the SUPERTRAC (Supersonic transition control) European Union (EU) [2] project. This shape, optimized for natural laminar flow, was originated by another configuration produced by Dassault Aviation within the framework of the Supersonic Business Jet project, and made available within SUPERTRAC project. The inboard wing has a 65° leading-edge sweep angle, and the outboard-wing sweep is 56°. The wing semispan is 9.35 m and the aspect ratio is 3.5. The cruise flight Mach number is 1.6.

Within SUPERTRAC project, wing section airfoils and twist angle were optimized in order to maximize the laminar flow region while monitoring and controlling the pressure (vortex and wave) drag. Compared to SUPERTRAC baseline the optimized wing showed an improved extent of laminar flow.

A redesign of the wing using robust or reliability based optimization tools is proposed, as the NLF configuration is the baseline configuration for the robust design problem defined in the UMRIDA project.

2.1 Geometry and Design Problem Definition

Figure 1 reports the UMRIDA baseline in isometric and orthographic projection, while wing-body geometric features are summarized in Table 1.

The flow conditions for the optimization problem are those related to the main cruise design point and are reported in Table 2.

2.2 Optimization Problem

The detailed description of the original optimization problem is reported in [2]. A synoptic view of design condition, constraints and objective as set up in the present investigation is summarized in Table 3. The objective function here proposed

$$G(C_L, C_D, C_M, ler, tea, \Delta x_{\mathrm{lam}})$$

has been changed to account for the new characterization of Mach and lift coefficient as uncertain parameters, and it is defined as:

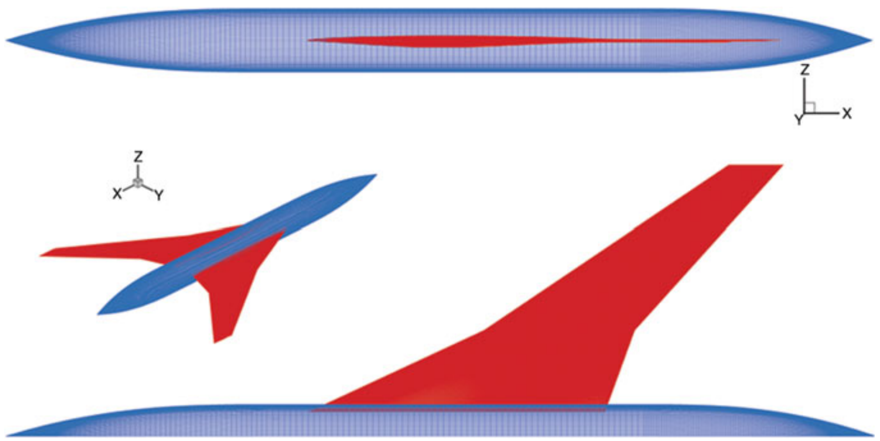


Fig. 1 UMRIDA baseline view

Table 1 Wing body geometric features

Parameters	Values
Inboard sweep	65°
Outboard sweep	56°
Semi-span length	9.35 m
Aspect ratio	3.5
Wing area	50 m^2

Table 2 Design flow conditions

Parameters	Values
Mach	1.6
Reynolds	51.8×10^6
L_{ref}	6.27 m
AOA	3.65°
C_L	0.182

$$
\begin{aligned}
G \;= K \frac{C_D + C_{D,M} + C_{D,L}}{C_L} \frac{\tilde{C}_L}{\tilde{C}_D} \\
+ (1-K)\frac{\Delta x_{\mathrm{lam}}}{\widetilde{\Delta x}_{\mathrm{lam}}} + \ell P\left(1 - \frac{ler}{\widetilde{ler}}\right) + tP\left(1 - \frac{tea}{\widetilde{tea}}\right)
\end{aligned}
$$

with the terms below aimed at accounting the contribution to drag due to trim:

Table 3 Problem definition

Design variables	
Wing twist	−3°; +3°
Wing section shape	User choice
Design point	
Mach number	1.6
Reynolds number	51 millions
Reference chord	6.27 [m]
Altitude	44000 [ft]
Lift coefficient	0.182
Design constraints	
Lift coefficient	$C_L \geq 0.180$
Pitching moment	$C_M \geq -0.05$
Trailing edge angle	$tea \geq \widetilde{tea} =$ 0.050 [rad]
Leading edge radius	$lea \geq \widetilde{lea} =$ 0.0020 [m]
Laminar extent—suction side	$\overline{X_{\rm tr}/c} = 0.35$
Laminar extent—pressure side	$\overline{X_{\rm tr}/c} = 0.45$
Laminar separation	$\overline{X_{\rm sep}/c} = 0.60$
Objective	
$G(C_L, C_D, C_M, ler, tea, \Delta x_{\rm lam})$	To be minimized

$$C_{D,M} = \max[0, 0.05(\widetilde{C_M} - C_M)]$$
$$C_{D,L} = \max[0, 1.0(\widetilde{C_L} - C_L)]$$

and where $\widetilde{C_L}$ and $\widetilde{\Delta x}_{\rm lam}$ are, respectively, the lift coefficient and the laminar extension indicator related to the baseline, and K, ℓ, and t are constant values that define the relative importance of the corresponding aerodynamic/geometric performance component. Suggested values for these constants are: $K = 0.25$, $\ell = 100$, and $t = 100$. The quadratic penalty function is activated only when its argument is positive. Hence, P has the following expression:

$$P(x) = \begin{cases} x^2 \text{ if } x > 0 \\ 0 \text{ if } x \leq 0 \end{cases}$$

The function $\Delta x_{\rm lam}$ is introduced to estimate the transition and laminar separation position on the whole wing, and it is defined as

$$\Delta x_{\rm lam} = \sum_{i=1}^{n} (\Delta x_{\rm lu} + \Delta x_{\rm ll} + \Delta x_{\rm su} + \Delta x_{\rm sl})$$

where

$$\Delta x_{\text{lu}} = \max\left(0,\, X^i_{\text{tr}} - \overline{X}^i_{\text{tr}}\right)_{\text{upper}}$$
$$\Delta x_{\text{ll}} = \max\left(0,\, X^i_{\text{tr}} - \overline{X}^i_{\text{tr}}\right)_{\text{lower}}$$
$$\Delta x_{\text{su}} = \max\left(0,\, X^i_{\text{sep}} - \overline{X}^i_{\text{sep}}\right)_{\text{upper}}$$
$$\Delta x_{\text{sl}} = \max\left(0,\, X^i_{\text{sep}} - \overline{X}^i_{\text{sep}}\right)_{\text{lower}}$$

Here, $\overline{X}^i_{\text{tr}}$ and $\overline{X}^i_{\text{sep}}$ are the computed values of transition and separation point at span section i, X^i_{tr} and X^i_{sep} are the desired values of transition and separation point at span section i, and n is the number of streamwise stations defined along the wingspan. Separation point is the chordwise abscissa at which the laminar boundary-layer calculation stops for each section: this can occur either because of a laminar separation (e.g., caused by a separation bubble) or because the boundary-layer solution does not converge at that point for some reason. Separation is taken into account in the Δx_{lam} objective function because, when it occurs upstream of the transition location, transition is automatically switched, even if the N factor has not yet reached the critical value. This approach is used in order to delay the laminar separation point as much as possible. The constraint value on inviscid drag and the set of desired transition locations have been assigned following preliminary studies and past experiences in laminar wing design. In particular, the drag penalty is activated with a huge weight when the inviscid drag exceeds the baseline value (195 drag counts). On the other hand, the transition specifications that have been imposed in the design problem represent a sort of utopia point, i.e., the actual threshold above which laminar flow would become really beneficial on aircraft performances and emissions.

3 Computational Model

The physical and computational model adopted for objective function computation is thoroughly described in [2]. A scheme of the computational setup is reported in Fig. 2 for the sake of completeness.

4 Definition of Uncertainties

The definition of uncertain parameters must take into account the design problem at hand. The aim of the designer is to obtain a configuration that has a satisfactory level of laminar flow even in the presence of parameters and working conditions not completely deterministic and controllable. A further difficulty is due to the epistemic uncertainty inherently included in the computational model for transition. This can

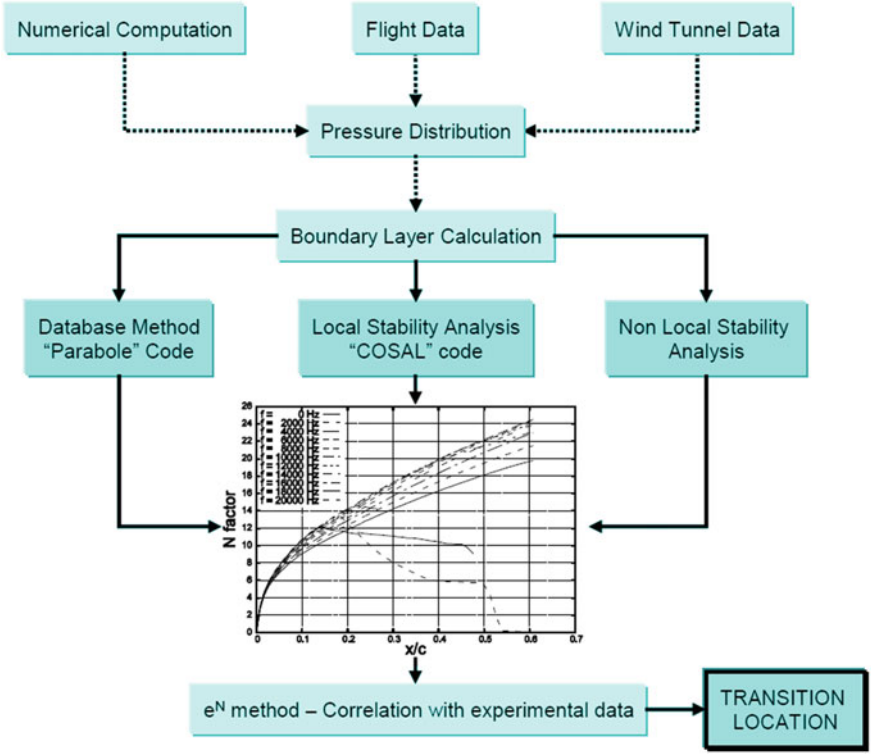


Fig. 2 Computational setup for the objective function computation

significantly affect the performance calculation of the new configurations. Consequently, a good design should be quite robust even with respect to this latter type of uncertainty sources.

4.1 Geometrical Uncertainties

Natural laminar flow is mostly sensitive to the shape of the leading edge region. This is due to its effect on pressure coefficient gradient which, in turn, is one of the factors that have more influence on the transition. The designer has complete freedom in defining the deterministic parameterization of the shape of the wing, but the statistical parameters that define the uncertainty of the shape of the wing are univocally defined. They are the radius of the leading edge and the airfoil section thickness at ten percent of the chord, and should be considered along the whole wing. It is the task of the designer to ensure that the variations of the radius of the leading edge and of the thickness of the wing section are well harmonized in the context of the parameterization chosen to describe the wing. The minimum requirement to be fulfilled is that these variations do not introduce discontinuities in the description of the wing section up to the second derivative, and that do not introduce inflections and oscillations in the geometry. The nominal range of variation for these parameters is 15% of the radius of the leading edge of the initial configuration and 10% of the initial thickness in the assigned position. A uniform probability distribution should be considered for both parameters. However, an inverse approach may be considered where these uncertain parameters are regarded as unknowns to be determined within the design process. As a consequence, the robustness of the obtained results may be made dependent upon the kind and ranges of probability distributions and, hence, the designer should determine which probability distribution provides the best compromise results in terms of robustness. In other terms, the designer can describe the robustness of the results indicating the probability distributions for which those results remains acceptable. In this case, the previously defined variation intervals for the random parameters have to be considered as a design target. This inverse approach may be very useful to quantify the machining and production tolerances that must be required for the production of a NLF wing.

4.2 Operational Uncertainties

Operational uncertainties are related to Mach number and lift coefficient (C_L). Mach and C_L are modeled as four parameter beta distributions. For the sake of completeness we recall that the probability density function f for $a \leq x \leq b$ and shape factors α, β is given by

$$f(x; \alpha, \beta, a, b) = \frac{(y-a)^{\alpha-1}(b-y)^{\beta-1})}{(b-a)^{\alpha+\beta-1}B(\alpha, \beta)}$$

and by $f = 0$ otherwise, with $B(\alpha, \beta) = \int_0^1 u^{\alpha-1}(1-u)^{\beta-1}du$. The table below summarizes the parameters that define the Mach and CL random variables (Table 4):

4.3 Model Uncertainties (epistemic)

One of the challenges that have to be faced when approaching the numerical design of natural laminar flow wings is the reliable estimation of the point where transition from

Table 4 Uncertain operational parameters

Parameter	α	β	a	b
Mach	4	4	1.55	1.65
C_L	2.5	2.5	0.180	0.184

laminar to turbulent flow occurs. A significant uncertainty in the determination of transition location is inherent to the methods for numerical transition prediction and in particular to the e^N method. A robust design approach should take into account this epistemic uncertainty source. If the e^N method is chosen, then the uncertain parameter to be considered is the N_{critical} factor. Its nominal value is fixed at 18 for deterministic runs, while it will be modeled by a uniform distribution in the interval [16, 20] when considered as uncertain parameter.

5 Preliminary Parametric Analysis

A preliminary parametric analysis was performed to identify, using ANOVA, the main dependencies and interactions of the parameters that describe the uncertainty in the problem.

For the sake of simplicity, the effect of uncertain parameters was not studied on the original objective function, but only on one component of it, namely the extension of laminar flow on the lower surface of the wing (LEXT_LO). Similarly, the external flow field was analyzed using rather coarse grids.

R [5] and DAKOTA (Design Analysis Kit for Optimization and Terascale Applications) [1] codes have been used to analyze results.

5.1 Effect of Uncertainty of N_{critical} Factor

The first analysis step was devoted to the evaluation of the computational model parameter N_{critical} on the location of laminar to turbulent flow transition on the lower surface of the wing. Therefore, in this analysis run, only the parameter N_{critical} was sampled according to its uniform distribution in [16, 20]. The population sample is composed of 400 members and includes the baseline.

The results, summarized in Table 5 and illustrated in Figs. 3 and 4, show that, at least for the limited region of parameter space in proximity to the baseline, the overall effect of N_{critical} is almost perfectly linear.

Table 5 Analysis of variance for the N_{critical} linear model fit

Response: LEXT_LO

	Df	Sum Sq	Mean Sq	F value	Pr (>F)
N_{critical}	1	0.48266	0.48266	126694	$<2.2 \times 10^{-16}$
Residuals	398	0.00152	0.00000		

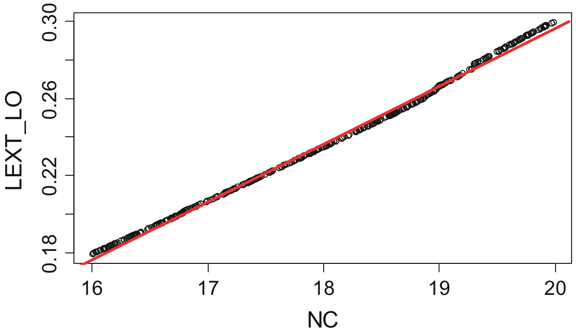


Fig. 3 Influence of N_{critical} (NC) on lower surface laminar flow extension

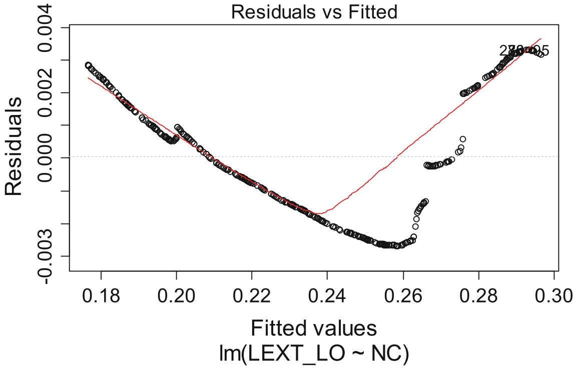


Fig. 4 Residual after the linear model fit of N_{critical}

5.2 C_L and Mach Contrast Effects on N_{critical}

Nevertheless, when we try to model the effects introduced by changing C_L and Mach, we realize that a linear model is no longer adequate. ANOVA, performed for models of increasing complexity, shows us that it is necessary to consider at least the terms of the second degree in C_L and Mach along with the first order interactions (C_L* Mach).

A Montecarlo sample of 1600 population elements has been performed to evaluate the simultaneous effects of C_L, Mach and N_{critical}. The ANOVA results, reported in Table 6, show that the effect of N_{critical} is still dominant, but we need to model the contrast produced by C_L and Mach using quadratic and interaction terms. The result of ANOVA between the different fitting models is reported in Table 7, while Fig. 5 reports a plot of the residual related to Model 3.

Table 6 Analysis of variance for the N_{critical} (N_C), C_L and Mach (M) model fit

Response: LEXT_LO

Model: LEXT_LO $\sim M + C_L + N_C + M * C_L * N_C + M^2 + C_L^2$

	Df	Sum Sq	Mean Sq	F value	Pr (>F)
M	1	0.00006	0.00006	1.4247e+01	0.0001662
C_L	1	0.00372	0.00372	8.7408e+02	<2.2e−16
N_C	1	1.97212	1.97212	4.6362e+05	<2.2e−16
M^2	1	0.00005	0.00005	1.1522e+01	0.0007048
C_L^2	1	0.00045	0.00045	1.0694e+02	<2.2e−16
$M : C_L$	1	0.00004	0.00004	8.5157e+00	0.0035704
$M : N_C$	1	0.00054	0.00054	1.2726e+02	<2.2e−16
$C_L : N_C$	1	0.00048	0.00048	1.1236e+02	<2.2e−16
$M : C_L : N_C$	1	0.00001	0.00001	2.6849e+00	0.1015029
Residuals	1590	0.00676	0.00000		

Table 7 Comparison of different fitting models for N_{critical} (N_C), C_L and Mach (M)

Response: LEXT_LO

Model 1: LEXT_LO $\sim M + C_L + N_C$

Model 2: LEXT_LO $\sim M + C_L + N_C + M * C_L * N_C$

Model 3: LEXT_LO $\sim M + C_L + N_C + M * C_L * N_C + M^2 + C_L^2$

	Res. Df	RSS	Df	Sum of Sq	F	Pr (>F)
1	1596	0.0083343				
2	1592	0.0071956	4	0.00113870	66.924	<2.2e−16
3	1590	0.0067634	2	0.00043216	50.798	<2.2e−16

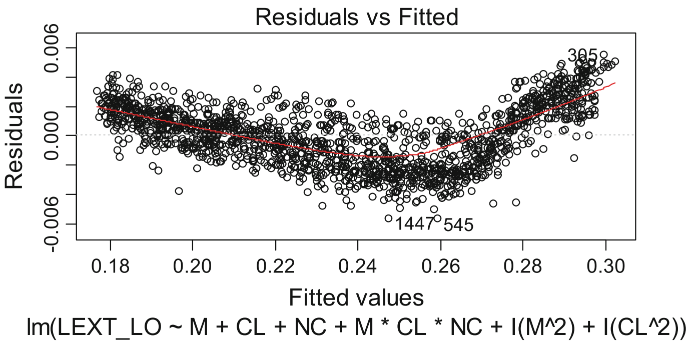


Fig. 5 Residual obtained using Model 3 for fitting N_{critical}, C_L and M

5.3 NURBS Parameterization to Model Uncertainties in Leading Edge Shape

The criticality of a parameterization that effectively models the uncertainty in the shape of the leading edge of a wing lies mainly in the fact that these parameters must simultaneously act on a small portion of the wing. At the same time, they should not produce discontinuities such to prevent the use of the fluid dynamic solver or interphere with the parameterization of the deterministic variables that control the shape of the whole wing. The method here chosen is based on Non Uniform Rational Basis-Splines (NURBS). It uses a grid of 21 × 7 NURBS control points (CPs) defined on the whole wing surface and a subset of it (21 CPs) is used to control the wing leading edge shape. In particular, 7 CPs modify the wing leading edge shape in streamwise direction using a 3rd order basis functions, and 3 CPs, with 2nd order basis functions, operate spanwise. The Design variables are the vertical displacements of the CPs (Figs. 6 and 7).

To evaluate the sensitivity to input variables a Latin Hypercube sampling (1000 samples) has been performed considering:

- 21 geometrical variables, uncertain with uniform distribution;
- 2 aerodynamic variables (MACH and CL), uncertain with beta distribution.

The responses, as in previous tests, is the laminar extent on the pressure side. The correlation analysis has considered the following response to inputs:

- Simple correlation: x_i versus f, e.g. considering the effects of varying individual response.
- Partial correlation: x_i versus f by removing the effect of the remaining x_j input variables.

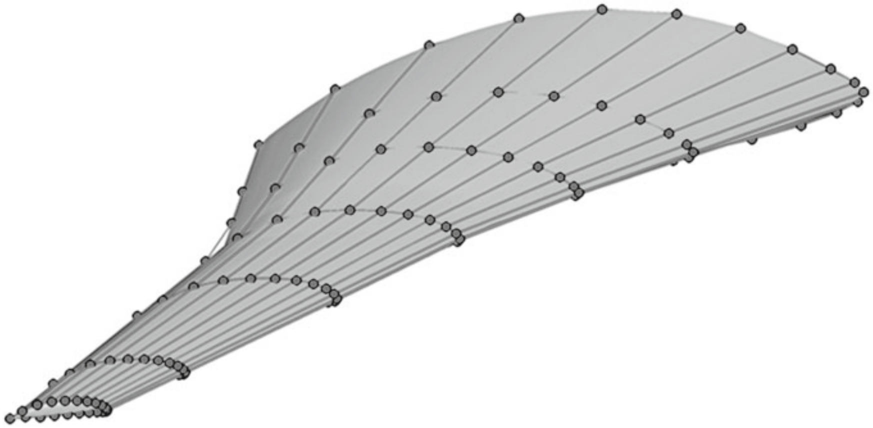


Fig. 6 NURBS Control Points to control the whole wing shape (21 × 7)

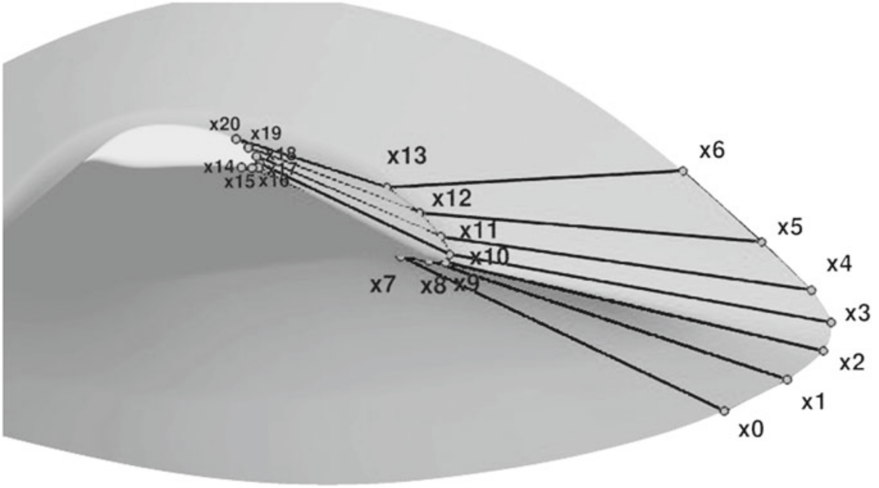


Fig. 7 Active NURBS Control Points to control local leading edge shape modifications (7 × 3)

- Rank correlation: "ranked" values are correlated instead of variable values. This kind of rank correlates the values of the variables, but highest correlation rank is where monotony is more respected, regardless of how you correlate the value pairs.

Figure 8 reports the scatter plot matrix related to the effect of variables x_1, x_8 and x_{15} (most correlated ones) on the response function. The diagonal shows the frequency histograms for variables x_1, x_8 and x_{15} and the response function. Effect of Mach and lift distribution on response function are, instead, reported in the scatter plot of Fig. 9. The diagonal shows the frequency histograms for C_L, and M and the response function. The observation of both figures evidences how the distributions are well described in input, and how the response function is negatively correlated with main geometric and aerodynamic parameters. Anyway, the response function mildly tends to increase with leading edge de-cambering or thickening, and slight increments can be observed also with decreasing Mach and lift coefficient. The green lines are the simple regression lines for each variable. It can be noted as the effect of M and C_L is comparable.

6 Sensitivity Analysis

The combined effects of geometry and operating condition uncertainties are here considered using sensitivity analysis. This task is aimed to discover the global sensitivities of the objective function related to the baseline configuration with respect to the above reported uncertainty sources.

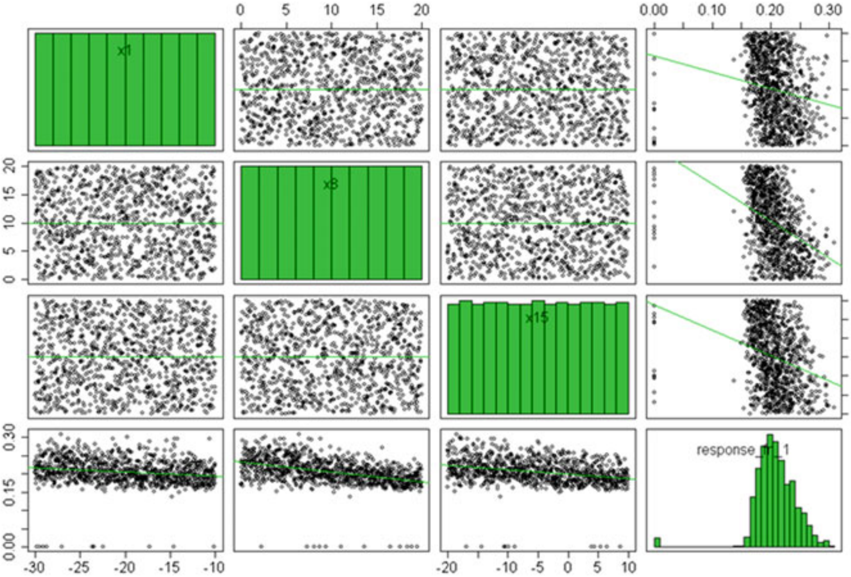


Fig. 8 Scatter plot related to the effect of variables x_1, x_8 and x_{15} on response function

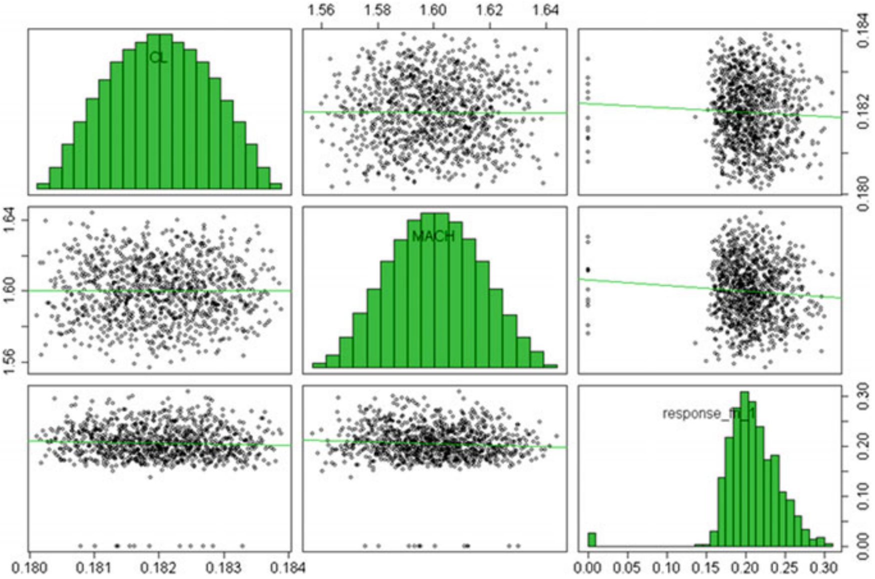


Fig. 9 Scatter plot related to the effect of variables C_L and M on response function

The sensitivities are computed in the hypothesis that uncertain variables are characterized by an uniform distribution on their whole range. The expected goal of this screening activity is the identification of the most important uncertain variables, relatively to their effect on the complete objective function, to allow a down-selection for the subsequent robust optimization step.

6.1 Variance-Based Decomposition and Sobol Indices

The parametric study was performed computing the Sobol indices via variance-based decomposition, a form of global sensitivity analysis that decomposes the variance of the model output into fractions which can be attributed to single input variables and into other parts related to sets of interacting input variables. This kind of analysis is able to deal with nonlinear responses and to evaluate the effect of interactions in non-additive systems. A basic assumption is that inputs are independently and uniformly distributed within the unit hypercube.

The total variance of the sampled response function is decomposed as:

$$\mathrm{Var}(Y) = \sum_{i=1}^{d} V_i + \sum_{i<j}^{d} V_{ij} + \ldots + V_{12\ldots d} \tag{1}$$

where V_i is the Variance due to the pure variation of factor X_i:

$$V_i = \mathrm{Var}_{X_i}\left(E_{X_{\sim i}}\left(Y|X_i\right)\right) \tag{2}$$

and V_{ij} is the variance due to the pure variation of both factors X_i, X_j:

$$V_{ij} = \mathrm{Var}_{X_{ij}}\left(E_{X_{\sim ij}}\left(Y|X_{ij}\right)\right) \tag{3}$$

with ~i notation indicating the effect of all variables except X_i.

The Sobol index S_i is defined as

$$S_i = \frac{V_i}{\mathrm{Var}(Y)} \tag{4}$$

and it measures the main (first order) effect, e.g. the effect of varying X_i alone, but averaged over variations in the other input parameters. The following relations hold:

$$\sum_{i=1}^{d} S_i \leq 1, \qquad \sum_{i<j}^{d} S_i + \sum_{i<j}^{d} S_{ij} + \ldots + S_{12\ldots d} = 1 \tag{5}$$

The index S_{Ti} measures instead the total effect, e.g. the contribution of all terms in the variance decomposition which do include X_i. Thus the first order effect plus the interactions are accounted by:

$$S_{Ti} = \frac{E_{X_{\sim i}}\left(\mathrm{Var}_{X_i}\left(Y|X_{\sim i}\right)\right)}{\mathrm{Var}(Y)} = 1 - \frac{\mathrm{Var}_{X_{\sim i}}\left(E_{X_i}\left(Y|X_{\sim i}\right)\right)}{\mathrm{Var}(Y)} \tag{6}$$

The interaction effect between e.g. X_i and X_j is counted in both S_{Ti} and S_{Tj}, hence we have

$$\sum_{i=1}^{d} S_{Ti} \geq 1 \tag{7}$$

6.2 Design Space Sampling

The goal of this sensitivity analysis is to understand the contribution of the uncertainty sources on the whole design problem and, hence, the full objective function value is considered, which may be rewritten as a weighted mixture of aerodynamic efficiency and of a measure of laminar portions on the wing surfaces:

$$G = -w_1 \left[\frac{C_{L,0}}{C_{D,0}} \left(\frac{C_D + \Delta C_{D,m} + \Delta C_{D,l}}{C_L} \right) \right] - w_2 S_{\mathrm{lam,up}} - w_3 S_{\mathrm{lam,low}} \tag{8}$$

with $\Delta C_{D,m} = \max[0.05(C_{M,0} - C_M), 0.0]$ and $\Delta C_{D,l} = \max[1.0(C_{L,0} - C_L), 0.0]$.

An Orthogonal Array Latin Hypercube sampling strategy [8] was chosen as it has both orthogonality and stratification features and is therefore a good choice for computing Sobol indices. The Sobol indices were computed using a corrected version of Saltelli's formula [6, 7] which is implemented within DAKOTA package. Two sample sizes were chosen to fill the hypercube, namely a small sampling that required 13,225 CFD evaluations and a big one that required 21,025 evaluations.

6.3 Failure Handling

The geometry modification process has been set up by empirically fixing the variation ranges of the uncertain geometry variables in order to obtain a global uncertainty in the description of the leading edge shape that was within the order of 5% of the wing section local thickness. This has made unavoidable some amount of mesh generator/CFD solver failures for particular combinations of design variables. The observed probability of failure was 0.006% for the big sampling case, and 0.009% for the small sampling. These percentages, although quite limited in numbers, may have the capacity to impair the computation of variance contributions and, hence, they have to be handled appropriately to not reduce the reliability of the computed sensitivity indices.

In this work, an exploratory approach has been adopted to the problem of handling failed samples, by varying the value that is assigned to the objective function in case of failures in the calculation chain. Three different failure responses were experimented here:

1. Assign a large response penalization to failed samples.
2. Assign response values near the database mean.
3. Assign a value slightly larger than the database maximum.

In the subsequent sections the results obtained with each of these approaches will be reported and discussed. Thanks to DAKOTA restart capabilities, assigning new response values to peculiar samples is very easy and does not require the recomputation of the objective function database.

6.4 Sensitivity Analysis Results

The S_i (main effect) and S_{Ti} (total effect) indices related to the big sampling are reported in Figures 10 and 11 respectively. It can be observed from Fig. 10 that the main effect is predominantly concentrated on variables 8, 12, 13 and 15 that control the pressure side shape on the mid-outboard wing. These variables have the highest influence on the objective function, but the main effect of Mach and C_L (variables 21 and 22) is also significant. These results are in good agreement with preliminary, coarse CFD grid analyses not reported here. It appears also evident that adding a large penalty to failed computations might lead to mask the true influence of the design parameters on the objective function. To avoid this problem different strategies to assign a score to failed computations were considered. In particular, it was found that assigning to the failed computations an objective value equal to the average or the maximum value from successful computations does not seem to affect the results quality in terms of relative sensitivities. Figure 11 is related to the variable total effect, e.g. including the interactions with the other variables, and it tells a different story. Here the mean value and the maximum value curves evidence that all the parameter shape values show significant interactions. Inboard wing variables (0–6) and Mach/C_L variables have statistically less importance with respect to mid-outboard ones. The large penalty curve shows how the interaction between variables 9, 11, 16 and 18 (controlling the leading edge radius and shape on the mid-outboard wing) play a major role in generating failed samples. The sum of main and total indices values are reported below to allow a comparison of the importance order of these effect when the objective function values related to failed configurations are changed:

$$\sum_{i=1}^{d} S_i = \begin{cases} 0.006 & \text{(large penalty)} \\ 0.39 & \text{(mean value)} \\ 0.24 & \text{(max value)} \end{cases} \tag{9}$$

$$\sum_{i=1}^{d} S_{Ti} = \begin{cases} 3.32 & \text{(large penalty)} \\ 6.68 & \text{(mean value)} \\ 5.34 & \text{(max value)} \end{cases} \tag{10}$$

Fig. 10 Big sampling, main effect

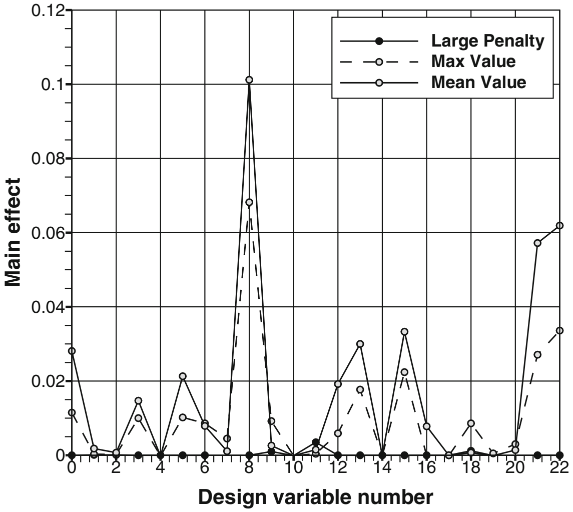
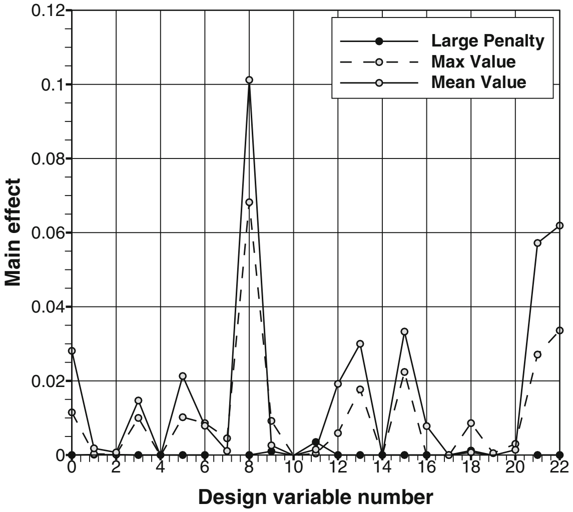


Fig. 11 Big sampling, total effect



Finally, Figs. 12 and 13 report the main and total effect for the small sampling set. In Fig. 12, the main effect, when computed using the mean value for failed samples, is roughly comparable with the one obtained using the big sample. Instead, the large penalty curve evidence a main effect in variables 9 and 18 that the big sample set was not able to spot. Finally, the mean value curve of Fig. 13 is roughly comparable with the results of Fig. 11. The large penalty curve evidences, instead, the role of interactions of variables 8, 9, 11, 15, 16 and 18 in generating geometries that will lead to failed computations. The sum of main and total indices values for the small sample set are reported below for the sake of completeness:

Fig. 12 Small sampling, main effect

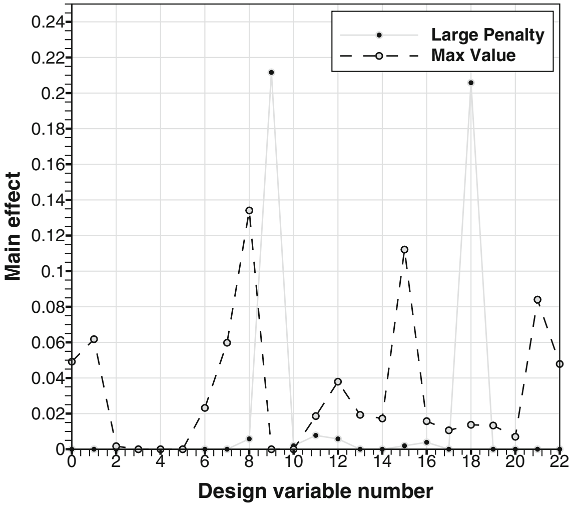
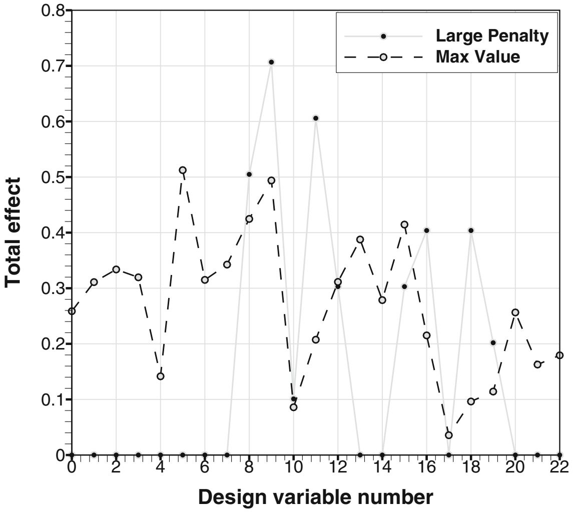


Fig. 13 Small sampling, total effect



$$\sum_{i=1}^{d} S_i = \begin{cases} 0.44 & \text{(large penalty)} \\ 0.73 & \text{(mean value)} \\ na & \text{(max value)} \end{cases} \tag{11}$$

$$\sum_{i=1}^{d} S_{Ti} = \begin{cases} 3.53 & \text{(large penalty)} \\ 6.20 & \text{(mean value)} \\ na & \text{(max value)} \end{cases} \tag{12}$$

7 Conclusions and Future Prospects

The analysis of the basic configuration has shown very clearly that the computational model used to calculate the aerodynamic performance in the presence of transition from laminar to turbulent flow presents critical points that the subsequent process of design and optimization will have to take into account. First of all, the computational model used to calculate the aerodynamic performance in the presence of transition consists of three heterogeneous software modules, namely an Euler/Navier-Stokes equation solver, a 3D boundary layer and an algorithm for linear stability, whose interactions are complex and not easy to track. However, at least for the baseline under the design conditions, the most important parameter, namely N_{critical}, showed an almost linear influence on the objective function. Accordingly, at least this part of the epistemic dependence of computational model can be easily considered in the optimization process. Vice versa, the dependence on operating parameters, Mach and C_L, has proven more complex and had to consider a regression model with a quadratic and interaction terms included to be able to model acceptably well the effects. Nevertheless, even in this case it is not too difficult to model these dependencies and insert them in the robust optimization loop. On the contrary, the effect of uncertainty on the wing leading edge has proved much more complicated to model and to take into account. What immediately jumps to the eye is the non-linearity of the effects and the interdependence of the parameters that define the uncertainty on the geometry. To analyze these effects we made use of an approach based on variance decomposition and Sobol indices. There have been two series of tests, the first with a coarser, but still significant, sample. The sensitivity analysis was then repeated with an increased the number of samples. The comparison of the two sensitivity analyses showed that, although not overlapping, the results are consistent. From the point of view of optimization, at least with the chosen parameterization, it is not easy to decrease the number of variables or simplify the model. However, it was possible to show that the correct management of failures in the calculations has a crucial role in the setup of the optimization problem. Finally, although not easy to achieve in practice, it would be desirable to reduce the variation range of the above parameters.

References

1. Adams, B., Bauman, L., Bohnhoff, W., Dalbey, K., Ebeida, M., Eddy, J., Eldred, M., Hough, P., Hu, K., Jakeman, J., Swiler, L., Vigil, D.: Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.4 user's manual. Technical report. Sandia Technical Report SAND2010-2183, Sandia National Laboratories, Livermore, CA (2009), Updated Apr 2013
2. Iuliano, E., Quagliarella, D., Donelli, R.S., El Din, I.S., Arnal, D.: Design of a supersonic natural laminar flow wing-body. J. Aircr. **48**(4), 1147–1162 (2011). https://doi.org/10.2514/1.C031039
3. Quagliarella, D., d'Ippolito, R., Iuliano, E., Donelli, R., El Masri, N., Donders, S.: Variability analysis and optimization of airfoil design under laminar flow conditions. In: LSAME.08—

Leuven Symposium on Applied Mechanics in Engineering. Katholieke Universiteit Leuven, Leuven, Belgium (2008)

4. Quagliarella, D., Iuliano, E., Donelli, R.S., de Rosa, D., d'Ippolito, R., El Masri, N.: Optimization of a natural laminar flow wing using robustness-based evolutionary optimization techniques. In: Proceedings of EUROGEN 2009 (2009)
5. R Core Team: R: a Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2015). http://www.R-project.org/
6. Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., Tarantola, S.: Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. Comput. Phys. Commun. **181**(2), 259–270 (2010)
7. Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S.: Global Sensitivity Analysis. The Primer. Wiley (2008). https://doi.org/10.1002/9780470725184
8. Tang, B.: Orthogonal array-based latin hypercubes. J. Am. Stat. Assoc. **88**(424), 1392–1397 (1993). https://doi.org/10.2307/2291282

Robust Airfoil Design in the Context of Multi-objective Optimization

Lisa Kusch and Nicolas R. Gauger

Abstract We apply the concept of robustness to multi-objective optimization for finding robust Pareto optimal solutions. The multi-objective optimization and robustness problem is solved by using the ε-constraint method combined with the non-intrusive polynomial chaos approach for uncertainty quantification. The resulting single-objective optimization problems are solved with a deterministic method using algorithmic differentiation for the needed derivatives. The proposed method is applied to an aerodynamic shape optimization problem for minimizing drag and maximizing lift in a steady Euler flow. We consider aleatory uncertainties in flight conditions and in the geometry separately to find robust solutions. In the case of geometrical uncertainties we apply a Karhunen-Loeve expansion to approximate the random field and make use of a dimension-adaptive quadrature based on sparse grid methods for the numerical integration in random space.

1 Introduction

Realistic engineering design involves the optimization of different competing objectives. Here, the aim is to find a set of solutions that fulfil the concept of Pareto optimality. A feasible design x is Pareto optimal if there does not exist any feasible design $\overline{x}$ such that $f_i(\overline{x}) \leq f_i(x)$ for every objective function f_i with $i \in \{1, ..., k\}$ and $f_j(\overline{x}) < f_j(x)$ for at least one $j \in \{1, ..., k\}$. A further significant step to realistic multi-objective designs is to take into account uncertainties for finding robust optimal solutions. Robust optimal solutions are solutions, that are optimal and robust with respect to perturbations.

In aerodynamic shape optimization one has to consider aleatory uncertainties in the flight conditions, that may arise due to turbulences, or in the geometry itself,

L. Kusch (✉) · N. R. Gauger
Chair for Scientific Computing, TU Kaiserslautern, Kaiserslautern, Germany
e-mail: lisa.kusch@scicomp.uni-kl.de

N. R. Gauger
e-mail: nicolas.gauger@scicomp.uni-kl.de

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_23

caused by manufacturing tolerances or temporary factors like for example icing [15]. Taking into account aleatory uncertainties during the optimization process may therefore help to improve the design process.

The challenge for algorithms that explore the Pareto optimal front is to find an evenly distributed set of objective vectors for approximating the front. Among these methods one can distinguish between scalarization methods that reduce the multi-objective optimization problem to several single-objective optimization problems, and direct Pareto approaches that find a set of representative solutions on the Pareto optimal front in one optimization procedure. Exemplary for direct approaches are multi-objective evolutionary algorithms. As we have the opportunity to obtain derivatives for deterministic optimization methods in an efficient way by making use of algorithmic differentiation [5] (AD), we apply a scalarization approach, namely the ε-constraint, method [10] together with a gradient-based optimization strategy for the resulting single-objective optimization problems. Another reason for the use of a constraint method is that it will be easier to use efficient one-shot techniques [2] for the single-objective optimization in future projects. One-shot methods have the potential to reduce the computational effort by one order of magnitude.

In the context of multi-objective optimization there can be seen growing interest in finding robust Pareto-optimal solutions when considering parameter or design uncertainties in the model. Multi-objective robust optimization problems are mainly treated in an evolutionary context: Deb and Gupta [3] introduce different types of multi-objective robust solution using a mean effective objective function. Instead of finding a mean effective objective function by means of sampling the approach can be extended to the use of statistical quantities like the expected value and the variance [12]. Other concepts to measure robustness include for example the use of sensitivity regions based on a local sensitivity analysis [6] or the use of a probabilistic domination operator to compare different designs under modelling errors [16]. We apply the ε-constraint method to solve for robust Pareto-optimal solutions using statistical measures like expected value and variance to quantify robustness.

To obtain statistical quantities, the uncertainties have to be propagated through the model. As the costs of a multi-objective optimization are already very high, it is important to use efficient approaches. In Schillings et al. [14], a non-intrusive polynomial chaos approach is used for single-objective aerodynamic robust design. In this approach the stochastic objective function is expanded in terms of polynomials that are orthogonal with respect to the density function of the input random variables. The non-intrusive approach results in a multiple set-point problem when used in the context of a semi-infinite robust design problem. The computational effort can be reduced by using sparse grids for the quadrature points. In Schillings et al. [14] the method is combined with one-shot optimization. A dimension-adaptive quadrature method is applied in the context of geometrical uncertainties. It is shown that the used approach is efficient for expensive aerodynamic design problems.

The main goal of our paper is to make use of a non-intrusive polynomial chaos approach for uncertainty quantification and to extend the ideas of Schillings et al. to multi-objective optimization using the ε-constraint method. We apply our suggested methodology for robust multi-objective design to robust airfoil design.

The underlying partial differential equations are the 2D Euler equations for steady flow. The considered performances are the lift and the drag coefficient.

In Sect. 2 we present all components needed for the robust multi-objective optimization. We introduce the notion of robust Pareto-optimal solutions, the non-intrusive polynomial chaos approach and the optimization method. The application to aerodynamic shape optimization is shown in Sect. 3. In Sect. 3.1 we consider uncertainties in the flight conditions and in Sect. 3.2 uncertainties in the geometry itself.

2 Robust Multi-objective Optimization

In single-objective optimization problems we consider a solution to be robust if it is not very sensitive to uncertainties ω. In multi-objective optimization problems the main difference to single-objective robust design is that one has to measure a combined effect of sensitivities for all objective functions represented by a point in the objective space. Additionally, one has to find a set of robust solutions instead of only a single robust solution. From this problem arises the question of how to define robust Pareto optimal designs.

2.1 Robust Pareto-Optimal Solutions

There exist different types of robustness measures that are used to describe robustness in a multi-objective context. We define a robust Pareto optimal design according to the first type of robustness measure in Deb and Gupta [3] using the expected value as a statistical quantity and call a solution robust Pareto optimal, if it is a solution to the problem

$$\begin{aligned} \min_{y,u} \; & \operatorname{Exp}(F(y,u,x(\omega))) \\ \text{s.t.} \; & c(y,u,x(\omega)) = 0, \\ & h(y,u,x(\omega)) \geq 0, \end{aligned} \tag{1}$$

where F is the objective vector. We will model the uncertainty by means of random variables with know probability density functions. The variables y and u are the state and design variables that fulfil the state equation $c(y,u) = 0$ and additional inequality constraints for a realization $x(\omega)$ with $\omega \in \Omega$. The minimization as well as the expectation operator have to be understood component-wise.

Instead of only looking at the expected value as a performance measure, one can incorporate the variance of the objective functions to take into account perturbations. This can for example be done by introducing an additional constraint resulting in the optimization problem

$$\begin{aligned} \min_{y,u} \quad & \operatorname{Exp}(F(y,u,x(\omega))) \\ \text{s.t.} \quad & ||\operatorname{Var}(\mathbf{F}(y,u,x(\omega)))|| \le \nu, \\ \text{s.t.} \quad & c(y,u,x(\omega)) = 0, \\ & h(y,u,x(\omega)) \ge 0. \end{aligned} \tag{2}$$

Note, that the existence of a solution to this problem depends on the choice of the threshold ν. As a result, one defines robustness as the Pareto optimal solution to a multi-objective optimization problem with additional constraints measuring the sensitivity. Another idea would be to incorporate the expected value and the variance as new objective functions instead of adding them in the form of constraints. This might become useful if the user is not able to specify the thresholds for the constraints. In the following we will use the robust formulation based on the expected value. For measuring the sensitivities we make use of probabilistic concepts for uncertainty quantification.

2.2 *Uncertainty Quantification*

There exist different methods to propagate uncertainties ω in the model. We make use of a non-intrusive polynomial chaos approach, which is also referred to as pseudo-spectral approach. In this approach the stochastic objective function is expanded in terms of polynomials Φ_i that are orthogonal with respect to the probability density function of the input random variables $x(\omega)$, such that

$$f(y,u,x(\omega)) = \sum_{i=1}^{\infty} f_i(y,u)\Phi_i(x(\omega)), \tag{3}$$

with $f_i(y,u) = \gamma_i^{-1}\operatorname{Exp}(f(y,u,x(\omega))\Phi_i(x(\omega))$ and $\operatorname{Exp}(\Phi_i\Phi_j) = \gamma_i\delta_{ij}$.

When applied to find statistical quantities the infinite expansion is truncated. The Fourier coefficients are approximated by first using stochastic collocation with quadrature points and then employing a quadrature rule that is suitable for the used polynomials.

2.2.1 Karhunen-Loève Expansion

In the case of geometrical uncertainties for the points ξ on the airfoil Γ it is necessary to approximate a random field $\psi(\xi,\omega)$ as the uncertainties depend on the shape itself. The random field is described by its mean $\psi_0(\xi)$ and its covariance function. As the random field results in an infinite-dimensional probability space we make use of the Karhunen-Loève (KL, [7, 9]) Expansion. The random field is decomposed into a linear combination of the eigenfunctions v_i of the covariance matrix such that it can be expressed as

$$\psi(\xi,\omega) = \psi_0(\xi) + \sum_{i=1}^{\infty} \sqrt{\lambda_i} v_i(\xi) X_i(\omega), \tag{4}$$

where λ_i are the corresponding eigenvalues sorted in decreasing order and the random variable X_i is described by

$$X_i(\omega) = \frac{1}{\sqrt{\lambda_i}} \int_{\Gamma} \psi(\xi,\omega) v_i(\xi) d\xi.$$

The KL expansion is truncated to get an approximation of the random field in the form of a finite-dimensional random space such that non-intrusive polynomial chaos can be applied to obtain the needed statistical quantities.

2.2.2 Dimension-Adaptive Quadrature

Especially when using geometrical uncertainties tensorized quadrature rules become computationally to expensive. The computational effort can be reduced by using sparse grids instead of fully tensorized grids for the quadrature points. Furthermore, generalized sparse grids can be produced with a dimension-adaptive strategy [4], in which the important dimensions are identified with the help of error estimators to refine the grid in these dimensions. The process is done adaptively starting with the coarsest sparse grid. Indices are added if the index set is admissible and the estimated error is reduced. The method can be understood as a sparse grid approach for a function f in a d-dimensional domain of integration with a generalized admissible index set $\mathscr{I}$, such that the quadrature formula reads

$$\tilde{Q}^{(d)}_{\mathscr{I}} f = \sum_{k \in \mathscr{I}} (\Delta_{k_1} \otimes \cdots \otimes \Delta_{k_d}) f \tag{5}$$

with the difference formula $\Delta_{k_i} = (Q_{k_i} - Q_{k_{i-1}-1}) f$ and $Q_0 f = 0$ for a quadrature rule Q. The generalized formula includes the formula for the full tensor grid as well as for the original sparse grid. The error indicator of an index $k \in \mathscr{I}$ is computed from $(\Delta_{k_1} \otimes \cdots \otimes \Delta_{k_d}) f$ and the algorithm adds the index with the maximum error indicator. In the case of more than one objective function we make use of the maximum l^2-norm of the error indicators.

A clear advantage of the dimension-adaptive strategy is that it enables the use of problem dependent quadrature formulas like the Gauss-Hermite formula. In Schillings et al. [14] the dimension-adaptive strategy is applied for the calculation of the expected value of the drag coefficient under the consideration of geometrical uncertainties.

2.3 Multi-objective Optimization Method

The formulation of robust Pareto optimal solutions results in a multi-objective optimization problem. We solve it by using the ε-constraint method. The concept of this method is to optimize one objective function f_s while imposing inequality constraints on the remaining competing objective functions. For the robust multi-objective optimization the objective function is a statistical quantity. The constraints as well as the objective function to be optimized are varied in each step of the algorithm to find different Pareto optimal solutions that are evenly distributed. The resulting minimization problem for the j-th step of the algorithm applied to a general multi-objective PDE-constrained optimization problem is

$$\begin{aligned} \min_{y,u} \quad & f_{s_j}(y,u) \\ \text{s.t.} \quad & c(y,u) = 0, \\ & f_i(y,u) \leq f_i^{(j)} \;\; \forall\, i \in \{1, ..., k\} : \; i \neq s_j. \end{aligned} \tag{6}$$

The inequality constraints for the different steps are distributed equidistantly. The outlines of the front can be found by minimizing the objective functions individually without imposing additional constraints. The procedure is depicted in Fig. 1 for a problem with two objective functions. Here, the dashed lines indicate the constraint values $f_i^{(j)}$.

It can be shown that all unique solutions to the resulting single-objective optimization problem (6) are globally Pareto optimal for any upper bound $f_i^{(j)}$ [11].

The choice of the algorithm for solving the single-objective optimization problems (6) that result from the ε-constraint method is very important. In Kusch et al. [8] a hybrid algorithm is applied for the single-objective optimization problems to enhance the chance of finding a global optimum.

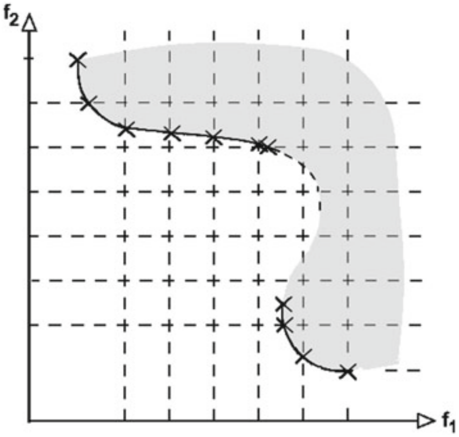


Fig. 1 Scanning the Pareto optimal front with the ε-constraint method

3 Application and Results

We apply the proposed method to an aerodynamic shape optimization problem for a NACA 0012 airfoil. The performances to be optimized are the drag coefficient c_d and the lift coefficient c_l. The flow is transonic and inviscid with a Mach number of 0.8 and an angle of attack of 1.25. Additional inequality constraints for the thickness of the airfoil and the resulting moment are used in the optimization process to restrict the number of feasible designs. In the robust optimization approach under consideration of uncertainties in the flight conditions these additional constraints are treated as always feasible. The airfoil is parametrized with the help of 38 Hicks-Henne functions.

The underlying steady Euler equations are solved with the Stanford University Unstructured (SU2) code [13] using a Jameson-Schmidt-Turkel scheme. SU2 is an open source software suite specialised on computational fluid dynamics, that can be used for performing PDE constrained optimization. A framework for algorithmic differentiation in SU2 was already developed by Albring et al. [1] and can be used to calculate derivatives for the gradient-based optimization.

All constrained single-objective optimization problems are solved with the deterministic, gradient-based interior point optimizer Ipopt [17]. IPOPT solves constrained nonlinear optimization problems with the help of an interior point method for the inequality constraints and a filter method for the resulting optimization problem with equality constraints.

In the following we investigate on robustness with respect to uncertainties in the flight conditions and geometrical uncertainties separately.

3.1 Uncertainties in the Flight Conditions

The scalar-valued uncertainties in the flight conditions are modelled by using random variables with an assumed probability density function. We start with an uncertain Mach Number and assume a normal distribution $Ma \sim N(0.8, 0.01)$. The associated orthogonal polynomials for the normal distribution f_x and the random Mach number x are the Hermite polynomials H_k with $(H_k, H_j)_{f_x} = k!\delta_{kj}$.

The expected value of c_d, for example, is then given by

$$\begin{aligned}\text{Exp}(c_d(y,u,x(\omega))) &\approx \text{Exp}(\sum_{k=0}^{m} c_{d,k}(y,u)H_k(x(\omega)))\\ &= 0!c_{d,0}(y,u) = \int_{-\infty}^{\infty} c_d(y,u,x)H_0(x)f_x(x)dx\\ &\approx \sum_{i=1}^{n} c_d(y,u,x_i)w_i.\end{aligned}$$

Note, that we have made use of the orthogonality of the polynomials and that we approximate the resulting integral with a quadrature formula with weights w_i and points x_i. For the normally distributed random variables we apply a Gauss-Hermite quadrature with four quadrature points.

The robust multi-objective optimization problem based on the expected value is given by

$$\begin{aligned} \min_{y,u} \quad & (\mathrm{Exp}(c_d), \mathrm{Exp}(-c_l)) \\ \text{s.t. } & c(y,u,x(\omega)) = 0 \; \forall \omega \in \Omega, \\ & h(y,u,x(\omega)) \geq 0 \; \forall \omega \in \Omega. \end{aligned} \tag{7}$$

When applying the above expression for the expected values and the ε-constraint method we obtain a multiple set-point problem

$$\begin{aligned} \min_{y,u} \quad & \sum_{i=1}^{n} c_d(y,u,x_i) w_i \\ \text{s.t. } & c(y,u,x_i) = 0 \; \forall i = 1, ..., n, \\ & h(y,u,x_i) \geq 0 \; \forall i = 1, ..., n, \\ & -\sum_{i=1}^{n} c_l(y,u,x_i) w_i \leq c_{l,j}, \end{aligned} \tag{8}$$

for each iteration step j for the constraint $c_{l,j}$ on the lift coefficient. The same applies to the constraints on the drag coefficient. The additional inequality constraints and the PDE constraint shall be feasible for all quadrature points x_i.

For minimizing drag and maximizing lift, the deterministic multi-objective optimization problem without any uncertainties included and the robust multi-objective optimization problems are solved for finding eight Pareto-optimal points.

In Fig. 2 the found points on the deterministic Pareto optimal front are marked with crosses and points on the expected Pareto optimal front that we will refer to as robust Pareto optimal solutions are marked with full diamonds. The empty diamonds show the expected points of the deterministic optimization and the small points are admissible points found during the deterministic optimization that shall give an idea of the admissible objective space. The points indicate a convex Pareto optimal front for each problem. One can clearly see the improvement gained by the robust optimization. On the right of the figure are the designs that correspond to the found points from top to bottom. The dashed shapes belong to the optimal designs and the solid shapes belong to the robust optimal designs.

The robustness is shown exemplary for one robust optimal design, that is given in Fig. 3. The expected value of the optimal design are $\mathrm{Exp}(c_d) = 0.00274$ and $\mathrm{Exp}(c_l) = 0.306$ and the expected values of the robust optimal design are $\mathrm{Exp}(c_d) = 0.00172$ and $\mathrm{Exp}(c_l) = 0.309$.

Figure 4 shows the behaviour of the drag coefficient around the mean Mach number. While the drag coefficient of the optimal design (dashed line) is sensitive around

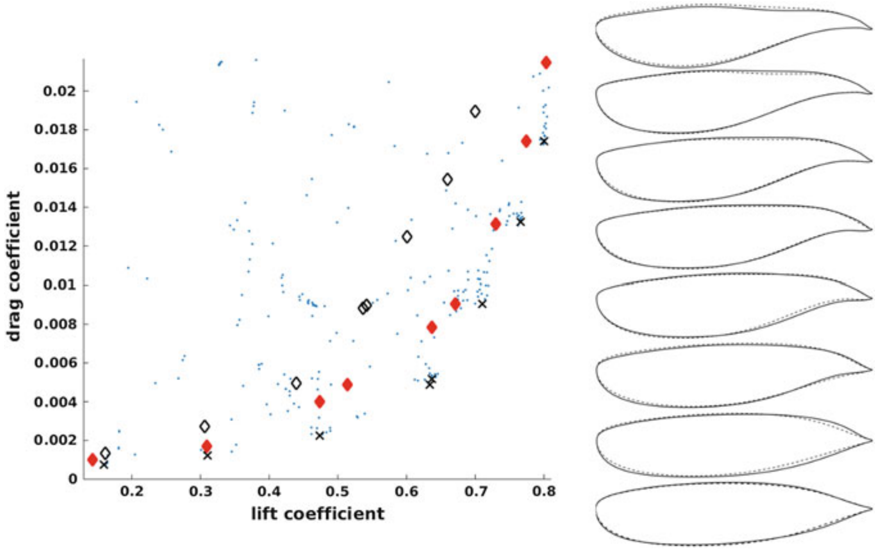


Fig. 2 Deterministic (crosses) and robust Pareto optimal points (full diamonds) and expected points of the deterministic optimization (empty diamonds) for the aerodynamic shape optimization problem and corresponding designs (right)

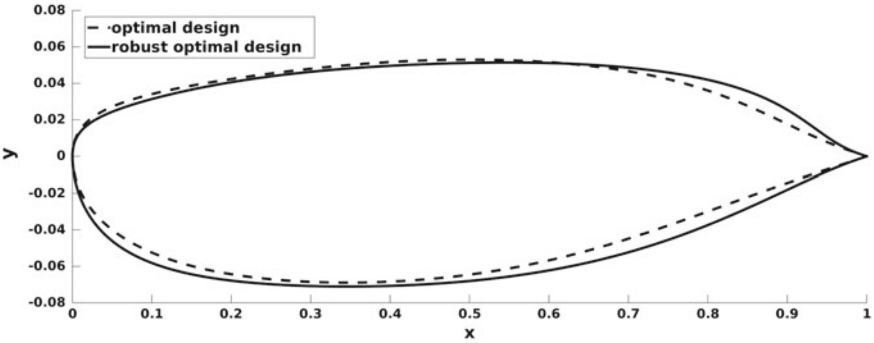


Fig. 3 Exemplary robust optimal airfoil design (solid line) and optimal airfoil design (dashed line)

the Mach number of 0.8, it is not very sensitive for the robust optimal design. The strong increase of the drag coefficient is also shifted to a higher Mach number. Similar results can be obtained for the lift coefficient.

The angle of attack α can be considered as an additional uncertainty that is normally distributed with a mean of 1.25 and a standard deviation of 0.01. For the computation of the two-dimensional random integral we make use of a full tensor grid with 16 Gauss-Hermite quadrature points. The resulting robust Pareto optimal

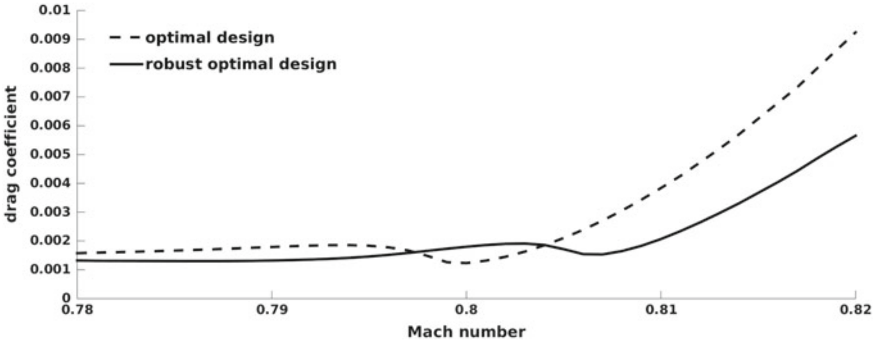
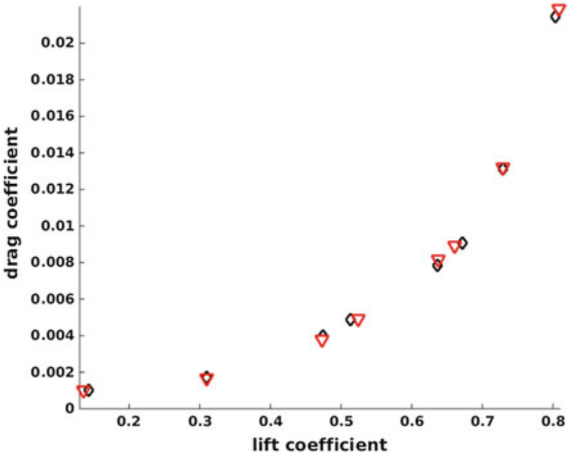


Fig. 4 Behaviour of the drag coefficient for the deterministic and the robust optimal design

Fig. 5 Robust Pareto optimal points for uncertainties in the Mach number and in the angle of attack (triangles) in comparison to robust Pareto optimal points for uncertainties in the Mach number (diamonds)



points marked by triangles in Fig. 5 are very similar to the points found when only considering an uncertainty in the Mach number (diamonds), which is also reflected in the designs.

3.2 Geometrical Uncertainties

As a next step we assume geometrical uncertainties of the airfoil. We restrict uncertainties to 80 percent of the airfoil Γ defined by $\tilde{\Gamma} = \{\xi \in \Gamma : \xi_x \leq 0.8\}$, which neglects the trailing edge. The coordinates of the perturbed airfoil are given by

$$\tilde{\xi} = \xi + \psi(\xi, \omega) \cdot n(\xi) \quad \forall x \in \tilde{\Gamma}, \omega \in \Omega \tag{9}$$

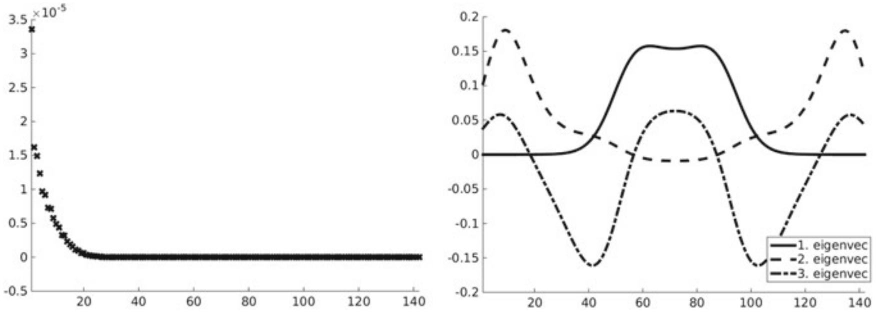


Fig. 6 Eigenvalues (left) and first three eigenvectors (right) of the covariance matrix of the original design

with the normal vector n. The expected value of the random process is $\text{Exp}(\psi) = 0$ and the covariance function is given by

$$\text{Cov}(\xi_i, \xi_j) = 0.001^2 \exp\left(-\frac{||\xi_i - \xi_j||_2}{0.1^2}\right) \quad \forall \xi_i, \xi_j \in \tilde{\Gamma}, \tag{10}$$

leading to a covariance matrix of size (142 × 142) for the discretized airfoil. Figure 6 shows the decaying eigenvalues on the left side and the first three eigenvectors on the right side for the NACA airfoil. In each optimization step a basis of the first three eigenvectors is chosen as an approximation of the random process.

The integral in the three-dimensional random space is approximated with the help of dimension-adaptive sparse grids based on Gauss-Hermite quadrature. The maximum order of quadrature points is 7. A full tensor grid quadrature would result in 343 quadrature points. The dimension-adaptive sparse grid for the original shape is shown on the left in Fig. 7. With a prescribed tolerance of 10^{-5} for the estimated error the dimension-adaptive strategy leads to 20 grid points. The dimension belonging to the third eigenvector does not seem to be important. The grid on the right consists of 27 grid points and is an example for a design in which the third eigenvector is of higher importance.

For the multi-objective optimization we prescribe the same constraints for the objective functions as for the deterministic optimization. The geometric constraint shall hold for the unperturbed design. Figure 8 shows the resulting Pareto optimal points represented by the full diamonds. The empty diamonds are the expected outcomes of the designs found by the deterministic optimization.

The found robust Pareto optimal points are very close to the original deterministic front, although the robust designs differ from the deterministic designs which can be seen in the right part of the figure. The solid shapes represent the robust designs and the dashed shapes are the designs resulting from the optimization without considering any uncertainties.

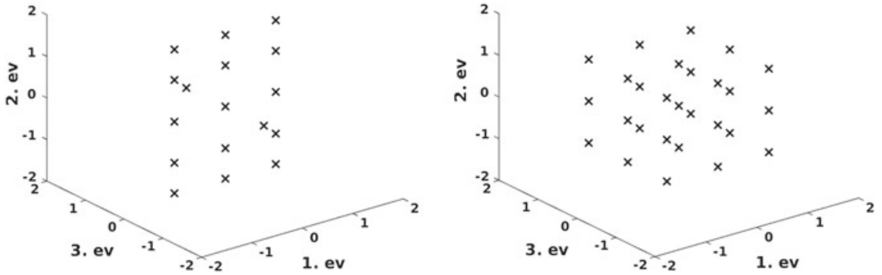


Fig. 7 Dimension-adaptive sparse grid for original design with 20 grid points (left) and for different design with 27 grid points (right)

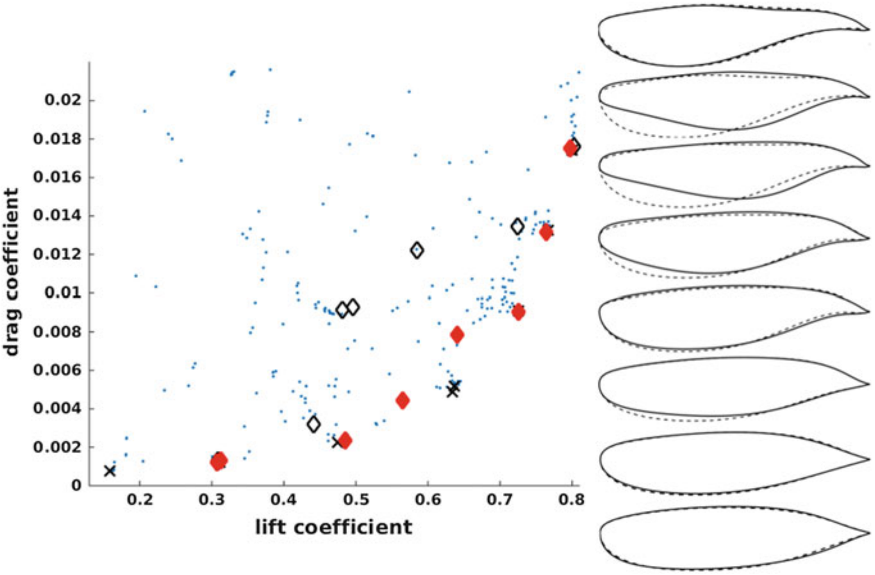


Fig. 8 Deterministic (crosses) and robust Pareto optimal points (full diamonds) and expected points of the deterministic optimization (empty diamonds) for the aerodynamic shape optimization problem and corresponding designs (right)

4 Summary and Outlook

We make use of the ε-constraint method combined with a non-intrusive polynomial chaos approach for uncertainty quantification to find robust Pareto optimal solutions. We apply the strategy to aerodynamic shape optimization to optimize for the drag and the lift coefficient. When assuming geometrical uncertainties we employ a Karhunen-Loeve expansion to approximate the random process and we make use of a dimension-adaptive strategy for numerical integration in random space.

In the future we plan to apply the proposed procedure to multidisciplinary optimization with SU2. Additional aims involve the application of a one-shot approach for solving the single-objective optimization problems and the investigation of further measures of robustness.

Acknowledgements We would like to thank our colleague Tim Albring from TU Kaiserslautern for assistance with SU2, and Claudia Schillings from University of Warwick for providing the code and the support for the dimension-adaptive quadrature.

References

1. Albring, T., Zhou, B.Y., Gauger, N.R., Sagebaum, M.: An aerodynamic design framework based on algorithmic differentiation. ERCOFTAC Bull **102**, 10–16 (2015)
2. Bosse, T., Gauger, N.R., Griewank, A., Günther, S., Schulz, V.: One-shot approaches to design optimzation. Int. Ser. Num. Math. **165**, 43–66 (2014)
3. Deb, K., Gupta, H.: Searching for robust pareto-optimal solutions in multi-objective optimization. Lect. Notes Comput. Sci. **3410**, 150–164 (2005)
4. Gerstner, T., Griebel, M.: Dimension-adaptive tensor-product quadrature. Computing **71**(1), 65–87 (2003)
5. Griewank, A., Walther, A.: Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation, Siam (2008)
6. Gunawan, S., Azarm, S.: Multi-objective robust optimization using a sensitivity region concept. Struct. Multidiscip. Optim. **29**(1), 50–60 (2005)
7. Karhunen, K.: Zur Sprektraltheorie stochastischer Prozesse. Suomalaisen Tiedeakatemian toimituksia. Math. Phys. Ser. A: 1 **34** (1946)
8. Kusch, L., Gauger, N., Spiller, M.: Efficient calculation of pareto-optimal points for shape optimization. In: Evolutionary and Deterministic Methods for Design, Optimization, and Control with Applications to Industrial and Societal Problems - EUROGEN 2013 (ISBN 978-84-617-2141-2), Universidad de Las Palmas de Gran Canaria, Spain (2014)
9. Loéve, M.: Probability Theory. Springer, New York (1978)
10. Marglin, S.A.: Public Investment Criteria. Allen & Unwin London (1967)
11. Miettinen, K.: Nonlinear multiobjective optimization. In: International Series in Operations Research & Management Science, vol. 12. Kluwer Academic Publishers, Dordrecht (1999)
12. Molina-Cristobal, A., Parks, G., Clarkson, P.: Finding robust solutions to multi-objective optimisation problems using polynomial chaos. In: Proceedings of the 6th ASMO UK/ISSMO Conference on Engineering Design Optimization, Citeseer (2006)
13. Palacios, F., Colonno, M.R., Aranake, A.C., Campos, A., Copeland, S.R., Economon, T.D., Lonkar, A.K., Lukaczyk, T.W., Taylor, T.W., Alonso, J.J.: Stanford university unstructured (su2): an open-source integrated computational environment for multi-physics simulation and design. AIAA Pap. **287**, 1–60 (2013)
14. Schillings, C., Schmidt, S., Schulz, V.: Efficient shape optimization for certain and uncertain aerodynamic design. Comput. Fluids **46**(1), 78–87 (2011)
15. Schulz, V., Schillings, C.: Problem formulations and treatment of uncertainties in aerodynamic design. AIAA J. **47**(3), 646–654 (2009)
16. Teich, J.: Pareto-front exploration with uncertain objectives. In: Evolutionary Multi-criterion Optimization, pp. 314–328. Springer, Berlin, Heidelberg (2001)
17. Wächter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math. Program. **106**(1), 25–57 (2006)

An Alternative Formulation for Design Under Uncertainty

F. Fusi, P. M. Congedo, G. Geraci and G. Iaccarino

Abstract A novel formulation for design under uncertainty is presented, which is based on the computation of the mean value and the minimum of the function. The aim of the method is to exert a stronger control on the system output variability in the optimization loop at a moderate cost. This would reduce post-processing analysis of the PDF of the resulting optimal designs, by converging rapidly to the interesting individuals. In other words, in the set of designs resulting from the optimization, the new approach should be capable of discarding poor-performance design. Also, no a priori assumption of optimal PDF is made. The preliminary results presented in the paper proves the benefit of the new formulation.

1 Motivation and Objectives

Optimization problems seek the design that improves a quantity of interest the most, according to a set of objectives and constraints. In an uncertainty-based framework, the variables or the model of the system under consideration are affected by a certain level of uncertainty, which will eventually affect the performance of the system as well. Thus, the quantity of interest depends not only on the design variables, but also on the uncertain variables of the system. The objectives of the optimization problem are formulated in order to take into account the statistical behavior of the performance

F. Fusi (✉)
Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano,
via La Masa 34, 20156 Milan, Italy
e-mail: francesca.fusi@polimi.it

P. M. Congedo (✉)
CARDAMOM Team INRIA Bordeaux Sud-Ouest, 200 Avenue de la Vieille Tour,
33405 Talence Cedex, France
e-mail: pietro.congedo@inria.fr

G. Geraci · G. Iaccarino
Mechanical Engineering and Institute for Computational and Mathematical
Engineering, Stanford University, Stanford, CA 94043, USA
e-mail: ggeraci@stanford.edu

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_24

due to the input uncertainty. For instance, Taguchi-like robust design [1] looks for a design that is minimally sensitive with respect to changes in the uncertain variables, by maximizing the mean value of the performance while minimizing its variance. Thus, a problem of uncertainty-based optimization can be formulated in several ways depending on the application and final goal. “Classical” formulations typically relies on the computation of the mean value, the variance, or a function of the two. These quantities however may not be capable of describing in a comprehensive way the Probability Density Function (PDF) of the output, which represents the behavior of the system. In recent years efforts have been made to find formulations which exert a control on the PDF of the performance inside the optimization loop [2, 3]. Nevertheless, the computation of the PDF inside the optimization loop may increase the computational cost, and defining an attainable target PDF may often be difficult, because the PDF is a feature of the problem and it is not known a priori in most cases.

Another example is the work presented in Ref. [4], which considers the influence of high-order statistics during the optimization process. The aim here is to provide some useful indications for obtaining a good trade-off between the high-quality information given by high-order statistics and the feasibility of the whole optimization loop. In particular, a multi-objective optimization method taking into account high-order statistic moments, such as the third and fourth-order statistic moments, i.e. skewness and kurtosis, respectively, is considered. From this work it appears that the skewness may not be very useful in the definition of the optimization objectives to control of the PDF, because it is difficult to handle in the optimization loop and it does not limit the tails of the PDF.

In this work, an alternative formulation for design under uncertainty is developed, which is based on the computation of the mean value and the minimum of the function. The aim of the method is to exert a stronger control on the system output variability in the optimization loop at a moderate cost. This would reduce post-processing analysis of the PDF of the resulting optimal designs, by converging rapidly to the interesting individuals. In other words, in the set of designs resulting from the optimization, the new approach should be capable of discarding poor-performance design. Also, no a priori assumption on the optimal PDF is made.

2 Optimization Problem

In an uncertainty-based optimization, the objectives are formulated to drive the variability of the performance f with respect to changes in the uncertain variables $\boldsymbol{\xi}$ to a desirable behavior, by acting on the design variables $\mathbf{x}$ [5]. In a probabilistic framework, the uncertain variables are treated as continous random variables in the stochastic space Ξ and they are assigned a probability density function p_{ξ}. The objectives of the optimization are measures of the response of the system under consideration to changes in the random variables. For instance, for classical robust design, the mean μ_f and the variance σ_f^2 are employed, and the problem then reads

$$\underset{\mathbf{x}\in\Sigma}{\text{optimize:}}\begin{cases}\mu_f(\mathbf{x})=\int_{\Xi} f(\mathbf{x},\boldsymbol{\xi})\;p_{\xi}(\boldsymbol{\xi})d\xi\\ \sigma_f^2(\mathbf{x})=\int_{\Xi}\big(f(\mathbf{x},\boldsymbol{\xi})-\mu_f(\mathbf{x})\big)^2\;p_{\xi}(\boldsymbol{\xi})d\xi,\end{cases} \tag{1}$$

where the design variables $\mathbf{x}$ vary in the design space Σ. The performance f is typically a function of the solution of the equations describing the system under consideration which can be defined as follows

$$\mathscr{L}\left(\mathbf{x},\boldsymbol{\xi};\varphi(\mathbf{x},\boldsymbol{\xi})\right), \tag{2}$$

where $\mathscr{L}$ is the mathematical operator of the equations and φ is the solution vector and $f=f(\varphi)$.

An uncertainty-based optimization requires the coupling of an outer optimization loop and an inner uncertainty propagation technique. The former provides the design to be tested at each iteration; the latter computes the statistics of the performance for that particular design, starting from the uncertainty affecting the system variables. The statistics of the performance represent the objectives, which are passed on to the optimization loop to generate a new design.

Choosing the objectives in an uncertainty-based optimization basically means choosing the statistics of the performance that one wishes to optimize. In this work, a different formulation is presented with the goal of inserting comprehensive information about the PDF of the performance in the optimization loop at a moderate cost, and without conditioning the result with constraints or targets. The novel strategy is also explored to assess if a particular choice of the objectives could influence the global cost of the optimization problem, for instance by reducing the number of the final optimal designs. In the new approach, the mean value of the performance is accompanied by the minimum of the performance. By maximizing both the mean value and the minimum, the optimization addresses the problem of finding a design that guarantees a desirable minimum performance while maintaining high mean performances. In addition, this approach could be suited to tackle reliability-based design problems, where the goal of the optimization is to reduce the occurrence of undesired system response (e.g. failures). Although the current work only explores problems related to enhancement of the performances, reliability-based design is deemed a possible application of the novel formulation and it will be explored in future work.

In the next section, the computation of the minimum in the uncertainty quantification method is discussed.

3 Computation of the Minimum

The computation of the minimum is performed in three different ways. The first, straightforward approach is the computation of the minimum value among the samples obtained for each point of a given quadrature formula (in this case, a Gaussian quadrature). This solution is referred to as Quadrature Point (QP) minimum computation and it is expressed as follows

$$\min_{\boldsymbol{\xi}_i \in \mathcal{N}_{QP}} v\left(\boldsymbol{\xi}_i\right), \tag{3}$$

where v is the output of the system, i.e. the quantity of interest, given a particular design and $\boldsymbol{\xi}_i$ are the samples drawn in the n_ξ-dimensional stochastic space belonging to the set $\mathcal{N}_{QP}$ of n_{QP} quadrature points.

The other two methods compute the minimum by leveraging the analytical approximations provided by two uncertainty quantification methods: (i) the Polynomial Chaos expansion, and (ii) the Simplex Stochastic Collocation. The former provides a meta-model of function v over the entire stochastic space, starting from a set of samples $\mathbf{v}$ [6]. Specifically, the meta-model is obtained by considering a spectral projection of function $v(\boldsymbol{\xi})$ onto a stochastic space spanned by a complete set of orthogonal polynomials Ψ that are functions of the random variables $\boldsymbol{\xi}$

$$v(\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \alpha_k \Psi_k\left(\boldsymbol{\xi}\right). \tag{4}$$

where Ψ_k are the PC orthogonal polynomials and α_k the coefficients of the expansion [6]. In practice, the series has to be truncated to a finite number of terms, which is determined from the number of uncertain variables and the order of the univariate polynomial expansion $\phi_i(\xi_i)$ from which the multivariate polynomials $\Psi_k(\boldsymbol{\xi})$ are obtained via tensorization, i.e.

$$\Psi_k\left(\boldsymbol{\xi}\right) = \prod_{i}^{n_\xi} \phi_i(\xi_i).$$

The polynomial basis $\phi_i(\xi_i)$ is chosen according to the Wiener-Askey scheme to select orthogonal polynomials with respect to the probability density function p_ξ. In this work, because a uniform distribution is considered, Legendre polynomials are employed. The orthogonality property can be advantageously used to compute the PC coefficients of the expansion α_k in a non-intrusive PC framework, i.e. the so-called Non-Intrusive Spectral Projection [7]. The projection requires an integration of the polynomials, which is obtained by means of a quadrature formula; the quadrature points are the same n_{QP} points previously mentioned for the QP method.

On the other hand, SSC is a multi-element method which relies on the adaptive refinement of the sampling in the stochastic space [8, 9]. The domain of the uncertain variables is divided into n_E elements, and each vertex in element Ξ_j represents a sample for which the exact function v is evaluated. Based on the set of samples $\mathbf{v}$ obtained for all vertexes, the approximation $w(\boldsymbol{\xi})$ of function $v(\xi)$ is built. In particular, a polynomial approximation is performed either on the whole domain (when the global extremum diminishing condition is satisfied [8]) or on each element Ξ_j

in which the grid is divided into. In the latter case, a piecewise polynomial representation is obtained. At each iteration of the method, the grid is refined according to the current error between the approximation w and function v in the new samples of the grid. A piecewise polynomial is obtained for each of the n_E elements in which the stochastic domain is divided into.

From the analytical approximation w (either the truncated PC expansion or the piecewise polynomial in the SSC), the gradient of the response for each design is computed and it is employed in a descent method based on Newton-Raphson iterations. From the location $\boldsymbol{\xi}_k$ at the k-th iteration, the new minimum location $\boldsymbol{\xi}_{k+1}$ is obtained using the descent step size γ and the descent direction $\mathbf{d}_k$ computed from the gradient $\nabla w(\boldsymbol{\xi}_k)$ and the Hessian $H(\boldsymbol{\xi}_k)$ of the analytical function in the current point:

$$\begin{aligned} \boldsymbol{\xi}_{k+1} &= \boldsymbol{\xi}_k + \gamma \mathbf{d}_k \\ H(\boldsymbol{\xi}_k)\mathbf{d}_k &= \nabla w(\boldsymbol{\xi}_k). \end{aligned} \tag{5}$$

If the Hessian is singular, the descent direction is taken proportional to the gradient. In the case of the PC expansion, the loop is started with the sample with lower value of function v, i.e. the minimum of the QP approach. With the SSC method, if a global interpolation is employed, the strategy used with PC still holds. Instead, when a local interpolation is employed, the descent method described in Eq. (5) is applied to each element Ξ_j in the stochastic grid and then the lower value of the local minima found on each element is computed.

It is worth noting that the underlying idea on building a meta-model of the function v is that the cost of sampling Eq. (2) is very high from a computational point of view. For instance, in the case of aerodynamic design, a Computational Fluid Dynamics (CFD) code is used to implement the equations of state, and a single CFD run can easily be 1000 times slower than the operations of a single optimization iteration. It follows that the cost of drawing a new sample is far higher than the cost of solving the iterative method in Eq. (5) to compute the minimum, because this procedure does not involve further sampling of Eq. (2).

4 Results

The new optimization strategy is based on the mean and minimum values. While the mean value is a typical quantity in uncertainty quantification, an assessment of the computation of the minimum is required as this is a key feature of the new approach. Thus, the first section presents the result concerning the computation of the minimum. In the second part, the impact of the minimum on the optimization loop is assessed. Thus, the results include first the assessment of the techniques presented in the preceding section to compute the minimum. Secondly, the impact of the minimum on the optimization loop is discussed.

With regard to the optimization method, the optimization is performed by means of the Non-dominated Sorting Genetic Algorithm (NSGA) in the formulation by Deb [10]. Main tuning parameters of the algorithm are the population size, the number of generations, the crossover and mutation probabilities pc, pm and the so-called sharing parameter r used to take into account the relative isolation of an individual along a dominance front. Typical values for pc, pm are, respectively, 0.9 and 0.1; values of r are defined following a formula given in Ref. [10] that takes into account the population size and the number of objectives. Finally, the population size and the number of generations for the convergence of the genetic algorithm should be chosen according to the number of parameters and objectives of the particular optimization problem. In this case, the population consists of 40 individuals and the number of generations is set to 60. Although genetic algorithms are very expensive from a computational point of view, they permit a wide exploration of the design space and the treatment of multi-objective optimization, which results in a rich, exhaustive set of optimal designs.

4.1 Computation of the Minimum

With regard to the first issue, the computation of the minimum is applied to the sine function

$$f(\xi) = \sin(2\pi\xi), \tag{6}$$

where ξ is a uniformly distributed variable in the interval [0, 1]. The variance and the minimum of the function are computed by means of QP, PC and SSC methods and the percentual errors of these estimates with respect to the analytical values are presented in Fig. 1. The QP estimate of the minimum presents some oscillations and the error remains higher with respect to the estimates obtained with the PC and SSC methods. In these cases, the convergence of the minimum is comparable to that obtained for the variance, although some oscillations are present, which are likely to be caused by a lack of accuracy in the representation of the function. As a matter of fact, since the minimum is a point value, it is generally more difficult to be well captured with respect to an integral value. However, the results prove that computing the minimum has a comparable cost with respect to computing the variance and that it is better to resort on functional approximation instead of using quadrature points.

The computation of the minimum is also applied to the Michalewicz function, a very challenging testcase for optimization algorithms, which presents one global minimum, many local minima and wide, flat regions [11]. The function $f(\xi_1, \xi_2)$ is considered here in the two-dimensional stochastic space, and it reads

$$f(\xi_1, \xi_2) = -\sum_{i=1}^{2} \sin(\xi_i)\ \sin^{20}\left(\frac{i\ \xi_i^2}{\pi}\right) \tag{7}$$

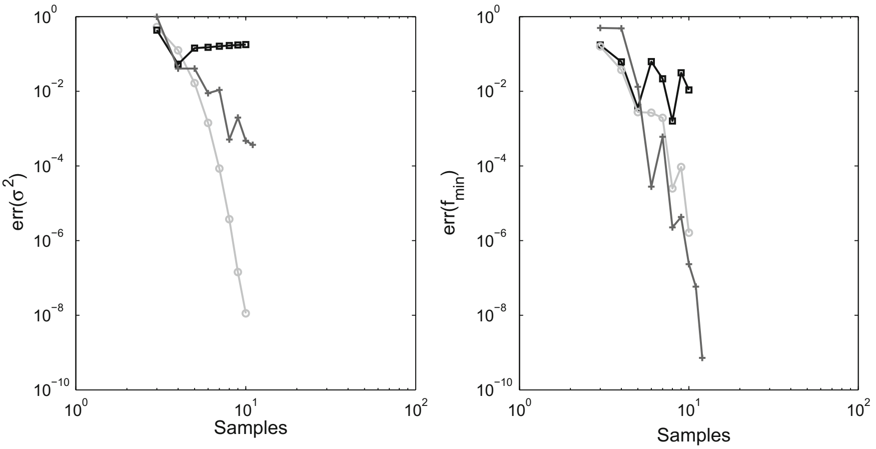


Fig. 1 Errors of the variance (left) and minimum (right) for one-dimensional sine function computed with PC (light grey circles), QP (black squares) and SSC (grey plus markers)

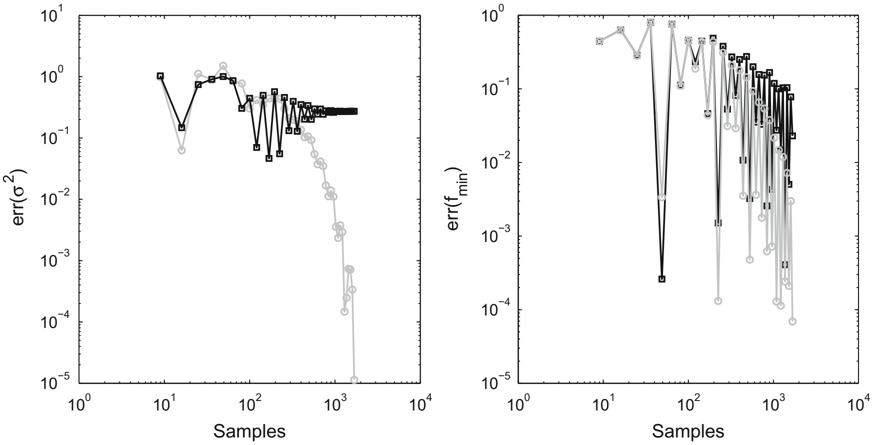


Fig. 2 Errors of the variance (left) and minimum (right) for two-dimensional Michalewicz function computed with PC (grey circles) and QP (black squares)

where each uncertain variable has uniform distribution in the interval $[0, \pi]$. This testcase is tackled by means of QP and PC approaches and the results of the errors for the variance and the minimum are presented in Fig. 2. In this case, the oscillations in the convergence of the minimum estimate are much larger, owing to the complexity of the function. Inspite of the oscillations, it is possible to note that the overall trend of convergence of the minimum is comparable to that of the variance. In summary, it is always better to compute the minimum of the functional approximation instead of using quadrature points.

4.2 Optimization with Minimum

To assess the impact of the computation of the minimum a modification of the Michalewicz function [11] is considered. The function depends on two uniformly distributed uncertain variables in the interval $[0, \pi]$, and on two design variables: $\mathbf{x} \in \Sigma = [0.1, 1]^2$. The modified function reads

$$\begin{aligned} f(\mathbf{x}, \boldsymbol{\xi}) &= M(\mathbf{x}, \boldsymbol{\xi}) + G(\mathbf{x}, \boldsymbol{\xi}) + P(\mathbf{x}, \boldsymbol{\xi}) \\ M(\mathbf{x}, \boldsymbol{\xi}) &= -\sum_{i=1}^{2} \sin(\xi_i + x_1)\ \sin^{20}\left(\frac{i\ (\xi_i + x_1)^2}{\pi}\right) \\ G(\mathbf{x}, \boldsymbol{\xi}) &= \frac{1}{2\pi x_2} \exp^{-\frac{(\xi_1-\pi/2)^2}{2x_2} - \frac{(\xi_2-3\pi/4)^2}{2x_2}} \\ P(\mathbf{x}, \boldsymbol{\xi}) &= 0.2\ (\xi_1 - \pi/2)\ (\xi_2 - \pi/2) \end{aligned} \tag{8}$$

Term M is a modification of the Michalewicz function with a phase angle equal to the first design variable, term G represents a bell function employed to uncorrelate the mean and the minimum, and the function P is added to avoid the flat regions of the original function that would yield a high peak of the PDF at zero. The optimization problem seeks the set of design that maximize both the mean value and the minimum

$$\max_{\mathbf{x} \in \Sigma} \left(\mu(f), \min(f)\right). \tag{9}$$

Figure 3 presents the Pareto fronts obtained using the minimum computation based on a Polynomial Chaos representation of 50-th order, the corresponding Quadrature Points, and a random sampling on the analytic function with 500000 samples which is taken as a reference. The SSC is not considered here for this explorative case. The solution found by the PC-based approach compares well with the reference result, while the approach based on the QP ends up on a different front. On the other hand, the approach based on the QP ends up on a different front and a different set of optimal designs, as presented in Fig. 4. Because the computation of the minimum based on the PC expansion proves robust and effective, it will be the method of choice in the following analysis.

The results of optimization (9) are now compared to the results obtained for the same problem and with the formulation of robust design that maximizes the mean value and minimizes the variance σ^2:

$$\max_{\mathbf{x} \in \Sigma} \ \mu(f) \ \wedge \ \min_{\mathbf{x} \in \Sigma} \sigma^2(f). \tag{10}$$

The Pareto fronts obtained with the two different formulations are presented in the μ, min-plane in Fig. 5 and in the μ, σ^2-plane in Fig. 6. It appears that both strategies are capable of finding the sets of design with higher mean values, where both fronts completely overlap. However, the minimum-based optimization discards the branch

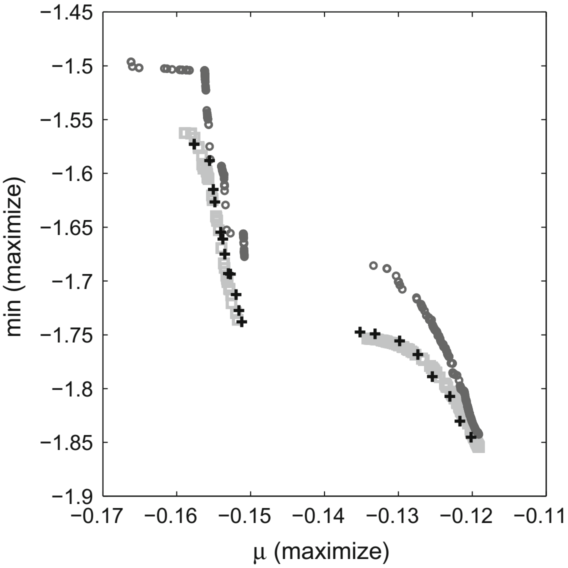


Fig. 3 Pareto fronts obtained with PC (light grey squares), QP (grey circles) and sampling on analytical function (black plus markers)

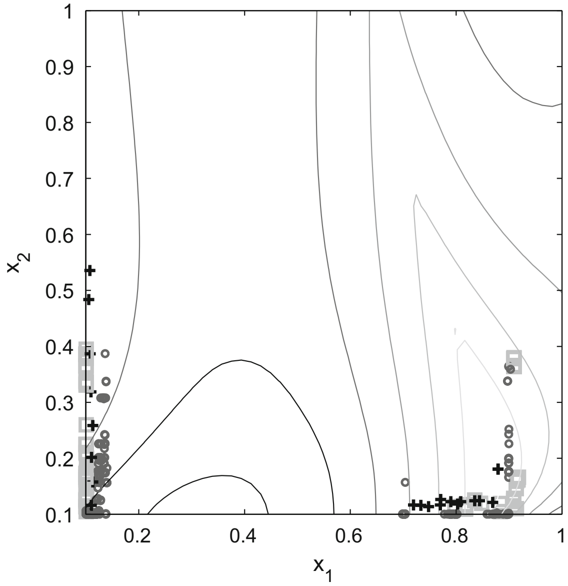


Fig. 4 Contour of the minimum with Pareto front designs and optimal Pareto designs obtained with PC (light grey squares), QP (grey circles) and sampling on analytical function (black plus markers)

Fig. 5 Reference design (black diamond) and Pareto fronts in μ, min-plane: minimum-based optimal solutions (light grey squares) and variance-based optimal solutions (black crosses)

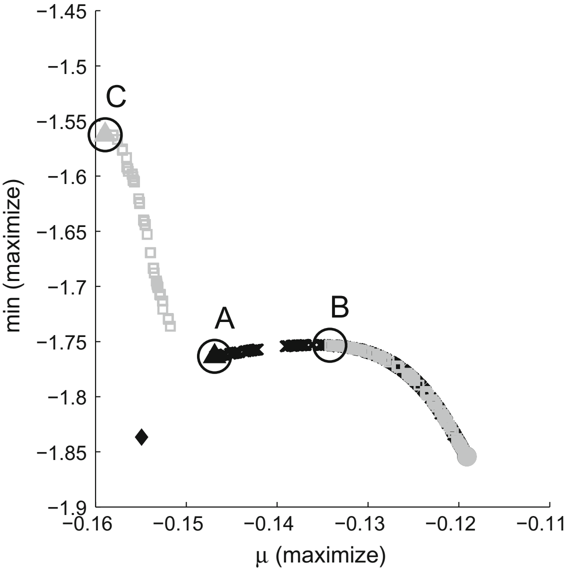
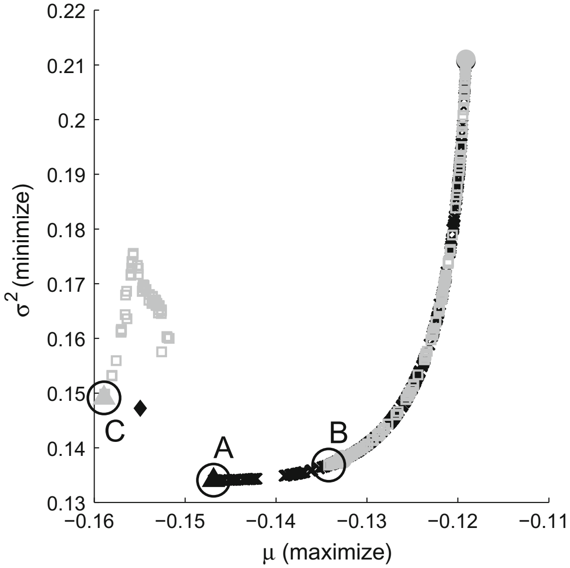


Fig. 6 Reference design (black diamond) and Pareto fronts in μ, σ^2-plane: minimum-based optimal solutions (light grey squares) and variance-based optimal solutions (black crosses)



composed of the designs with minimum variance, because they are dominated in the μ, min-plane. On the other hand, the front obtained by maximizing the mean and minimum values includes a branch with higher minimum values, which, in turn, the robust optimization cannot find. It is thus evident that the optimization based on the mean value and the minimum results in the subset with higher mean performance of the front obtained with the classical robust approach, but it also contains a branch with improved minimum values. Please note that in many applications, the solutions with very low variance comes at the expense of very poor performance values, to the extent that these solutions may be rejected in the decision making stage. In these cases, it is clear the benefit of a method that detects solutions with high mean value in the μ, σ^2-front.

Three optimal solutions from the non-overlapping branches are compared: from the optimization based on the mean value and variance, the design with optimal variance (A) is selected and, from the optimization based on the minimum, the designs with optimal variance (B) and optimal minimum value (C) are chosen. For these designs the PDF of function f is plotted in Fig. 7. The functions share a similar shape, although the PDF of design B has higher mean value. It is interesting to note that for this particular case, despite a lower mean value of the performance, the design with greatest minimum may be an interesting solution for design purposes, because it possesses not only zero probability of low performances, but also the highest maximum value of the performance. This solution would never show up in the classical optimization based on the mean value and the variance. The optimal designs have been obtained without any assumptions on the final shape of the PDF, proving that the method is well suited for exploration of the optimization problem.

As a final remark, the size of the optimal set of designs obtained with the new formulation is comparable to that obtained with the classical approach. In fact, because the function of the testcase in Eq. (8) is designed in order to uncorrelate the mean value, the result is a wide Pareto front. However, in the engineering case tackled in Ref. [4] the Pareto front obtained with the novel approach is much smaller with respect to the other; the difference in the size of the optimal set speeds up convergence of the genetic algorithm to the final results, reducing the computational effort of the novel formulation (Table 1).

Table 1 Performance of the selected optimal designs

	$\mu(f)$	$\sigma^2(f)$	$\min(f)$
A	−0.1469	0.1341	−1.763
B	−0.1342	0.1368	−1.753
C	−0.1589	0.1491	−1.562

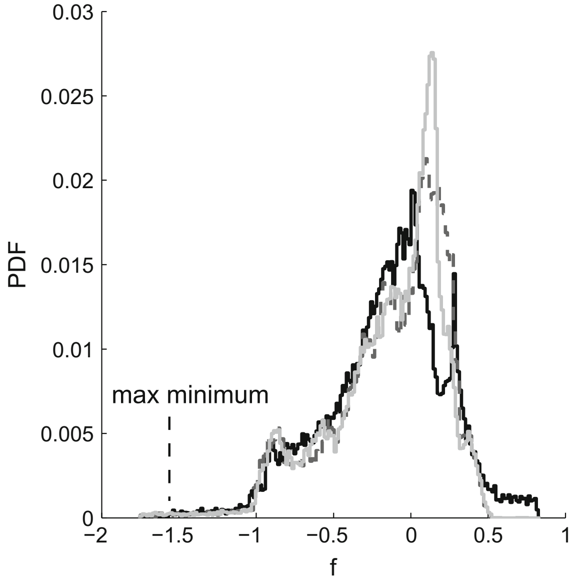


Fig. 7 PDF of selected optimal designs: design A (light grey solid line), design B (grey dashed line) and design C (black solid line)

5 Conclusions

In this work a novel formulation for design under uncertainty have been developed and assessed, which is aimed at maximizing both the mean value and the minimum of the performance. Three methods for computing the minimum are presented, which are based on the Polynomial Chaos expansion and the Simplex Stochastic Collocation. The method based on the quadrature points of the PC expansion appears less efficient with respect to the other strategies which directly exploit the continous analytical approximations provided by these methods. The computation based on the PC expansion is effective and because it is the most straightforward and widely-employed method for uncertainty quantification, it may be the method of choice for this new formulation. With regard to the impact on the optimization loop, the method has proven effective in selecting only the designs with higher mean performance from the front obtained with the classical robust approach, and in finding new interesting designs with improved minimum values. As a matter of fact, this explorative study represents one of the first result in which a set of optimal designs is found which is discarderd with a classical approach.

Future activities will be devoted to consolidate the computation of the minimum. In particular, the implementation of a refinement strategy based on the minimum in the SSC algorithm will be developed and assessed for different test cases with the aim of further reducing the computational cost of the new formulation. In addition, testcases of engineering interest will be solved with the new optimization formulation and compared to existing strategies. In this case, because reliability-based analysis seems to be a natural application of this formulation, test cases of this problem will be considered.

Acknowledgements The authors acknowledge the associated team (Inria-UQ Lab Stanford) AQUARIUS for the financial support.

References

1. Taguchi, G., Phadke, M.S.: Quality engineering through design optimization. In: Quality Control, Robust Design, and the Taguchi Method, pp. 77–96. Springer (1989). https://doi.org/10.1007/978-1-4684-1472-1-5
2. Petrone, G., Iaccarino, G., Quagliarella, D.: Robustness criteria in optimization under uncertainty. EUROGEN 2011, p. 2011. Italy, Capua (2011)
3. Seshadri, P., Constantine, P., Iaccarino, G., Parks, G.: Aggressive design under uncertainty. In: 16th AIAA Non-Deterministic Approaches Conference. AIAA SciTech (2014)
4. Congedo, P.M., Geraci, G., Iaccarino, G.: On the use of high-order statistics in robust design optimization. In: 5th European Conference on Computational Mechanics (ECCMV), 20–25 July 2014, Barcelona, Spain (2014)
5. Schuëller, G.I., Jensen, H.A.: Computational methods in optimization considering uncertainties—an overview. Comput. Methods Appl. Mech. Eng. **198**(1), 2–13 (2008). https://doi.org/10.1016/j.cma.2008.05.004
6. Xiu, D., Karniadakis, G.E.: The wiener–askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput. **24**(2), 619–644 (2002). https://doi.org/10.1137/S1064827501387826
7. Congedo, P.M., Corre, C., Martinez, J.-M.: Shape optimization of an airfoil in a BZT flow with multiple-source uncertainties. Comput. Methods Appl. Mech. Eng. **200**(1–4), 216–232 (2011). https://doi.org/10.1016/j.cma.2010.08.006
8. Witteveen, J.A.S., Iaccarino, G.: Refinement criteria for simplex stochastic collocation with local extremum diminishing robustness. SIAM J. Sci. Comput. **34**(3), A1522–A1543, (2012a). https://doi.org/10.1137/100817498
9. Witteveen, J.A.S., Iaccarino, G.: Simplex stochastic collocation with random sampling and extrapolation for nonhypercube probability spaces. SIAM J. Sci. Comput. **34**(2), A814–A838 (2012). https://doi.org/10.1137/100817504
10. Srinivas, N., Deb, K.: Multiobjective function optimization using nondominated sorting genetic algorithms. Evol. Comput. **2**, 221–248 (1995)
11. Yang, X.-S.: Test problems in optimization. In: Engineering Optimization: an Introduction with Metaheuristic Applications. Wiley (2010)

Polynomial Representation of Model Uncertainty in Dynamical Systems

Massimiliano Vasile

Abstract This chapter introduces an approach to capture unmodelled components in dynamical systems through a hierarchical polynomial expansion in the state space. This approach is reminiscent of the empirical acceleration approach commonly used in precise orbit determination to account for unmodelled components in the force model.

1 Introduction

In orbit determination, and more generally in the propagation of uncertainty in dynamical systems, one problem is to capture uncertainties in the dynamical model itself. Although dynamical models are normally dependent on a number of parameters that can be calibrated using observations, the functional form of the dynamical model can be incomplete.

A commonly used approach, in precise orbit determination, is to introduce *empirical accelerations* as additional components of the dynamics. The value of these empirical accelerations can be defined in a number of different ways exploiting the available measurements.

It is customary to use time series expansions in polynomial or trigonometric form whose coefficients need to be found by matching the prediction of the model with the observations [1]. Another approach is to treat empirical accelerations as stochastic processes that can be reconstructed by a form of sequential filtering [2, 3]. Other more recent approaches treat the unmodelled components as a stochastic process that is represented with a Gaussian mixture [4].

All these techniques generally work satisfactorily and allow one to use a reduced dynamics without the need for extremely high fidelity models. On the other hand they do not immediately furnish a functional representation of the missing components.

M. Vasile (✉)
Aerospace Centre of Excellence, University of Strathclyde, 75 Montrose Street,
Glasgow G11XJ, UK
e-mail: massimiliano.vasile@strath.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_25

Even the use of time series expansions, which are valid within the interval in which the measurements are available, to extrapolate the behaviour of the dynamical system does not always lead to the desired results. Furthermore, time series do not provide information on the dependency of the empirical accelerations on any of the state variables.

For this reason, in this chapter, it is proposed the use of polynomial expansions, with unknown coefficients, of the unmodelled components in the state variables. If the state variables and the observations are treated as stochastic variables, then so are the coefficients of the polynomial expansion. It will be shown that this formulation can effectively capture missing components in simple dynamical systems, including hypersensitive ones, both in the case of a reduced number of observations and in the case of observations affected by uncertainty.

The chapter first introduces the general formulation of the problem and the polynomial expansion of the uncertain components. It then presents an optimisation process, that is required to calculate the coefficients of the expansion, and the concept of uncertainty distance as a metric in the space of the unknown coefficients. Some examples follow, that illustrate the results that can be obtained with this approach.

2 Polynomial Expansion of Unmodelled Components

Consider the two functions $f : S \times P \times [t_0 : t_0 + T] \longrightarrow \mathbb{R}^n$ and $\upsilon : S \times B \times [t_0 : t_0 + T] \longrightarrow \mathbb{R}^n$ with $S \subseteq \mathbb{R}^n$ and the initial value problem:

$$\begin{cases} \dot{s} = f(s, p, t) + \upsilon(s, b, t) \\ s(t_0) = s_0 \end{cases} \tag{1}$$

where s is the state vector. The function $\upsilon(s, b, t)$ represents some unknown function of the states that is capturing all unmodelled components, $p \in P \subseteq \mathbb{R}^{m_p}$ a set of uncertain model parameters, $b \in B \subseteq \mathbb{R}^{m_b}$ some unknown parameter vector of the unmodelled components, and t the time coordinate. In this paper, we will study only the case in which the unmodelled components are not a function of time (the case with time dependence is easily obtained from the time independent formulation) and the missing component is added to the known component. Furthermore, let us consider the special case in which the function $\upsilon(s, b)$ can be expressed as follows:

$$\upsilon(s, b) = \begin{cases} 0 \\ Q = \nabla_r U_r(s, b) + \nabla_v U_v(s, b) \end{cases} \tag{2}$$

where $Q : S \times B \times [t_0 : t_0 + T] \longrightarrow \mathbb{R}^q$, with $q < n$, and U_r and U_v are two continuous and differentiable scalar uncertainty functions that can be expanded in the following hierarchical form:

$$
\begin{aligned}
&U_r(s,b) \simeq \sum_i^{2N} a(b)_i^r \xi_i(s_i) + \sum_i^{2N}\sum_j^{2N} a(b)_{ij}^r \xi_{ij}(s_i,s_j) + \\
&\sum_i^{2N}\sum_j^{2N}\sum_k^{2N} a(b)_{ijk}^r \xi_{ijk}(s_i,s_j,s_k) + \cdots \\
&U_v(s,b) \simeq \sum_i^{2N} a(b)_i^v \xi_i(s_i) + \sum_i^{2N}\sum_j^{2N} a(b)_{ij}^v \xi_{ij}(s_i,s_j) + \\
&\sum_i^{2N}\sum_j^{2N}\sum_k^{2N} a(b)_{ijk}^v \xi_{ijk}(s_i,s_j,s_k) + \cdots
\end{aligned}
\tag{3}
$$

with $n = 2N$ the dimension of the state space. If Eq. (1) describes the time evolution of a dynamical system, then Q can be seen as a generalised force whose hth component, with $h = 1, ..., q$, is:

$$
\begin{aligned}
&Q_h(s,b) = \tfrac{\partial U_r}{\partial r_h} + \tfrac{\partial U_v}{\partial v_h} \simeq c_0 + \sum_i^{2N} c(b)_i \zeta_i(s_i) + \sum_i^{2N}\sum_j^{2N} c(b)_{ij} \zeta_{ij}(s_i,s_j) + \\
&\sum_i^{2N}\sum_j^{2N}\sum_k^{2N} c(b)_{ijk} \zeta_{ijk}(s_i,s_j,s_k) + \cdots
\end{aligned}
\tag{4}
$$

If ζ are monomial bases then the generalised forces read:

$$
\begin{aligned}
&Q_h(s,b) \simeq c_0 + \sum_{i \neq k}^{2N} c(b)_i \Delta s_i + \sum_i^{2N}\sum_j^{2N} c(b)_{ij} \Delta s_i \Delta s_j + \\
&\sum_i^{2N}\sum_j^{2N}\sum_k^{2N} c(b)_{ijk} \Delta s_i \Delta s_j \Delta s_k + \cdots
\end{aligned}
\tag{5}
$$

The vector c has dimension l. Note that the vector function Q can be directly expanded in polynomial series without going through a scalar function U. In fact, in the most general case in which the force field has no potential, it is easier to directly expand Q. However, in the case in which a U function can be found, the uncertainty, from which the missing component is derived, can be expressed in a more compact form. Hence, although in the following we will use only Q, in this section we presented also the idea of deriving Q from a scalar function U.

2.1 Problem Statement

Given Q and a set of observations, one can obtain an approximated representation of the unmodelled components by finding the value of c that best fits the measurements. Then, the value of the coefficients of expansion (4) can be obtained as the solution of an optimisation problem. The nature of the optimisation problem slightly differs depending on the integration scheme used to solve Eq. (1). If $N_o = l$ exact and distinct measurements are available then one needs to solve the following set of constraints:

$$s(t_i, c) - s_o(t_i) = 0; \quad \imath = 1, ..., N_o \tag{6}$$

where $s(t_i, c)$ is the propagated state at time t_i and $s_o(t_i)$ is the observed state at time t_i. If the number of observations N_o is equal to the number of coefficients in the expansion (4), one could argue that the solution of problem (6) provides the exact values of all the coefficients c. If the number of exact measurements is lower than the number of coefficients c, a suitable smoothing function is required and the following problem needs to be solved:

$$\begin{gathered} \min_c \; J(s, c) \\ s.t. \\ s(t_i, c) - s_o(t_i) = 0; \quad \imath = 1, ..., N_o \end{gathered} \tag{7}$$

where $J : S \times C \longrightarrow \mathbb{R}$ is a function of states s and coefficients c. Note that, in general, problem (7) can have more than one solution for c even when $N_o = l$. In fact, consider the simple second order differential linear equation:

$$\ddot{x} = -kx + d \tag{8}$$

where $k > 0$ and $d > 0$. The general solution has the form:

$$x = A\cos(\omega t + \phi) + B \tag{9}$$

Given that Eq. (8) has two unknowns, one could think that a single observation of both velocity and position would suffice. However, given the initial conditions $x(t = 0) = 0$ and $\dot{x}(t = 0) = 0$, for the observation $x = 0$ and $\dot{x} = 0$ at time $t = 2\pi$, a different solution exists for every $\omega = j \in \mathbb{N}^+$.

2.2 Treatment of Stochastic Observations

In the case of observations affected by an error, one cannot obtain a prediction of the exact value of the parameters c. In this case it is reasonable to assume that the initial conditions are also uncertain as they come from previous observations. If the expected values of the state vector, coming from observations, are enforced as hard constraints the result might not capture the actual missing components as the trajectory is forced to satisfy constraints that do not come from the natural dynamics but are dependent on the errors in the observations. One option is to consider the most probable value for each observation and a cost function that maximises the likelihood of correct identification. The other option is to quantify the uncertainty in the observations and initial conditions as confidence intervals on the observed states. More formally, consider the uncertainty space $(\Gamma, \mathscr{L}, \mathscr{M})$, with Γ a non empty set, $\mathscr{L}$ a σ-algebra over Γ, and $\mathscr{M}$ an uncertain measure. Then the observed state s_o is an uncertain variable $s_o : (\Gamma, \mathscr{L}, \mathscr{M}) \longrightarrow \mathbb{R}^n$. If the distribution of s_o is available one can draw N_p samples and solve problem (7) N_p times to derive a distribution of the coefficients c. Alternatively, if no distribution is available for s_o but Σ is the collection of all the confidence intervals for all the observations, including the initial conditions, such that:

$$Pr(s_o \in \Sigma) > \varepsilon \tag{10}$$

then one can formulate the following optimisation problem:

$$\begin{aligned} &\min_{c \in \mathscr{C}} \; J(s, c) \\ &s.t. \\ &s(t_i) \in \Sigma \quad i = 0, ..., N_o \end{aligned} \tag{11}$$

where ε is an arbitrary value. The main advantage of this formulation is that no statistical moments are required and no exact distribution needs to be known a priori. Note that the initial conditions $s(t_0)$ are treated as an observed state.

2.3 Uncertainty Distance

The coefficients of the polynomial expansion define the motion of a physical system. Therefore, one assumption is that the dynamics will follow a minimum action principle. This is not necessarily always true as the missing components might correspond to a transient state. At the same time one can assume that the function Q is a stochastic function that define the probability of the system to be in a particular state. As a consequence the function J in (7) can be expressed in different forms. The one that is proposed in this paper assumes that the system is at a minimum energy state which means that the function Q introduces the minimum level of noise compatible with the observations.

In this sense, the objective function in (7) can be interpreted as a distance in the metric vector space $\mathscr{C}$ of the parameters c. In this space, the origin represents the solution with no model uncertainty and any point at distance $\sqrt{c^T c}$ from the origin has uncertainty vector Q and uncertainty distance:

$$d_u = \int Q^T Q \; dt \tag{12}$$

Note that by analogy, one can formulate problem (7) as the constrained optimal control problem:

$$\begin{aligned}
&\min_u \ J(u) = \tfrac{1}{2}\int u^2 dt \\
&s.t. \\
&\dot{s} = f(s, p, t) + u \\
&and \\
&s(t_i) \in \Sigma \quad i = 0, ..., N_o
\end{aligned} \tag{13}$$

where $u = Q$. It is important to note that other definitions of J are possible, although the results in this paper seems to suggest that the J function derived from (12) and defined as:

$$J = c^T c \tag{14}$$

provides good results, at least for the cases here analysed.

2.4 Solution Through Optimisation

Problem (11) can be solved using a simple multistart approach. Starting from a Latin Hypercube grid of randomly selected starting points we use the Matlab function `fmincon` to find a constrained local minimum from each of the points in the grid. The dynamics was integrated with the Matlab function `ode45`, with both the absolute and the relative tolerances set to 1e-9. In all the examples in this paper, this simple procedure was sufficient to find acceptable solutions. Given the global nature of the problem, more complex dynamical systems might require more sophisticated procedures. It should be noted that the convergence of `fmincon` was dependent on the scaling of the coefficients c. A wild choice for the first guess of c or setting too broad boundaries for the $\mathscr{C}$ space can result in an integration failure of `ode45`.

3 Examples

This section contains a number of simple examples and the results that the proposed method can provide.

3.1 Linear Elastic Dynamics with Friction

The first example considers the following simple dynamical system with an elastic component and friction:

$$\begin{aligned}
\dot{v} &= -x - 0.5v \\
\dot{x} &= v
\end{aligned} \tag{15}$$

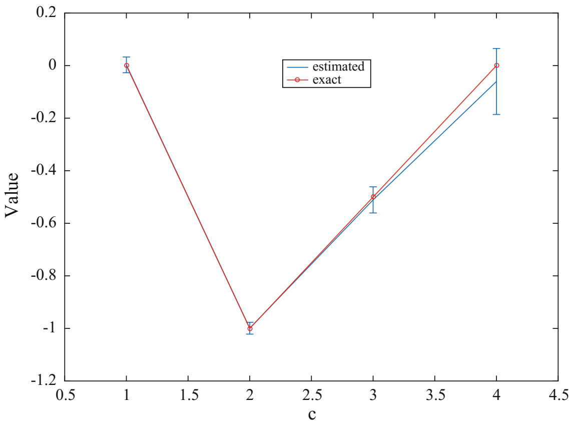


Fig. 1 Reconstructed coefficients for the linear elastic dynamics with friction. The dotted red curve is the exact value of the coefficients. The x-axis is the index of the coefficients and the y-axis their values

We assume that both elastic and friction components are unmodelled and we use the representation:

$$\begin{aligned} \dot{v} &= c_1 + c_2 x + c_3 v + c_4 v^2 \\ \dot{x} &= v \end{aligned} \tag{16}$$

By introducing two measurements at $t = \pi$ and $t = \pi/2$, assuming a uniform distribution of the measurements over an interval that is $\pm 10\%$ of the exact value of the states, and solving problem (7) with cost function $J = c^T c$ we get the result in Fig. 1. The figure shows in blue the mean and confidence interval of the value of the components of the c vector. The same figure shows in red the exact value that the coefficient should have to reproduce the exact dynamics. As one can see the exact value is contained in the confidence interval and the estimated mean value is quite close to the exact one. In particular the first three coefficients are a good match. The fourth one has a much higher variance suggesting that more than one dynamics is compatible with the measurements. This is reasonable in this case as a small drag component, $c_4 v^2$, would produce an effect similar to a larger friction component, $c_3 v$, over the time span considered in this example.

3.2 Orbital Motion with Unmodelled Drag

The second example is an orbital motion with unknown drag component. The gravity component of the model is fully known but the observations show an additional component that is not modelled. The real dynamics is assumed to be governed by the following system of differential equations in polar coordinates:

$$
\begin{array}{l}
\dot{v_r} = -\frac{\mu}{r^2} + \frac{v_t^2}{r} - \frac{1}{2}\rho C_d v v_r \\
\dot{v_t} = -\frac{v_t v_r}{r} - \frac{1}{2}\rho C_d v v_t \\
\dot{r} = v_r \\
\dot{\theta} = \frac{v_t}{r}
\end{array} \tag{17}
$$

We assume a unitary area to mass ratio, and a constant density ρ such that the product of the density times the drag coefficient C_d is $1/2\rho C_d = 10^{-6}\,\text{kg/m}^3$. Furthermore, we assume that the expected trajectory, given the known dynamic components, is a circular orbit with $v_r(t = 0) = v_{r_0} = 0$ and $v_t(t = 0) = v_{t_0}$. The orbital period, without drag, is $T = 2\pi\sqrt{r^3/\mu}$. If one expands the modulus of the velocity v in Taylor series up to the first order, the differential equations with the drag term can be approximated as:

$$
\begin{array}{l}
\dot{v_r} = -\frac{\mu}{r^2} + \frac{v_t^2}{r} - \frac{1}{2}\rho C_d v_t v_r \\
\dot{v_t} = -\frac{v_t v_r}{r} - \frac{1}{2}\rho C_d v_t^2 \\
\dot{r} = v_r \\
\dot{\theta} = \frac{v_t}{r}
\end{array} \tag{18}
$$

In order to capture the unmodelled component of the dynamics, we assume the following expansion with terms up to order 2 in velocity and position:

$$
\begin{array}{ll}
\dot{v_r} = -\frac{\mu}{r^2} + \frac{v_t^2}{r} + c_1 + c_3 r + c_5 r^2 + \\
\qquad c_7 r\theta + c_9 v_r + c_{11} v_r^2 + c_{13} v_r v_t \\
\dot{v_t} = -\frac{v_t v_r}{r} + c_2 + c_4\theta + c_6\theta^2 + \\
\qquad c_8 r\theta + c_{10} v_t + c_{12} v_t^2 + c_{14} v_r v_t \\
\dot{r} = v_r \\
\dot{\theta} = \frac{v_t}{r}
\end{array} \tag{19}
$$

If the linear effects in Eq. (18) are dominant over a given time span Δt, then the prediction given by Eq. (19) should be of the form:

$$
\begin{array}{l}
\dot{v_r} = -\frac{\mu}{r^2} + \frac{v_t^2}{r} + c_{13} v_r v_t \\
\dot{v_t} = -\frac{v_t v_r}{r} + c_{12} v_t^2 \\
\dot{r} = v_r \\
\dot{\theta} = \frac{v_t}{r}
\end{array} \tag{20}
$$

We can now introduce observations at time $t = T$ and $t = T/2$, for a total of 8 constraint equations and 14 parameters, and solve problem (7) with cost function $J = c^T c$.

The estimated coefficients in the case of exact measurements are represented in Fig. 2 (blue line) and compared to the expected values assuming a linear model (denoted by red circles). The resulting prediction of the trajectory over two orbits is shown in Figs. 3 and 4. Note that the values of all the coefficients in the figures were scaled up by 10^{-6} to make them comparable to the value of general orbit

Fig. 2 Example of reconstructed gravity-drag dynamics for exact measurements

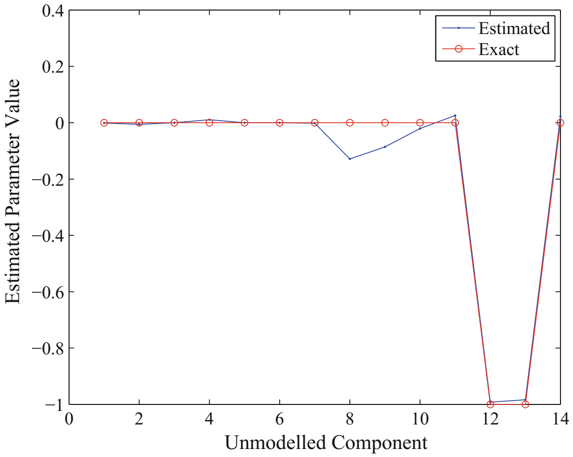


Fig. 3 Prediction over $2T$—radius and true anomaly

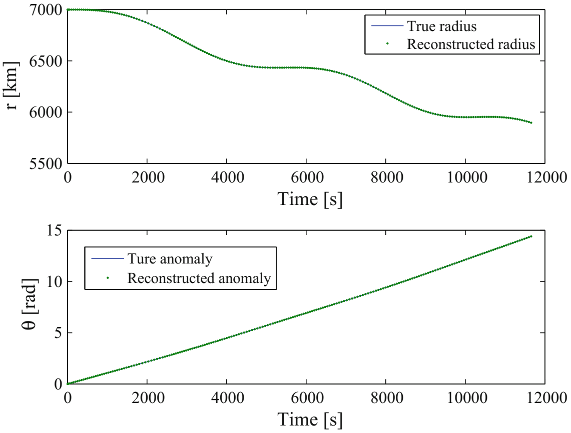
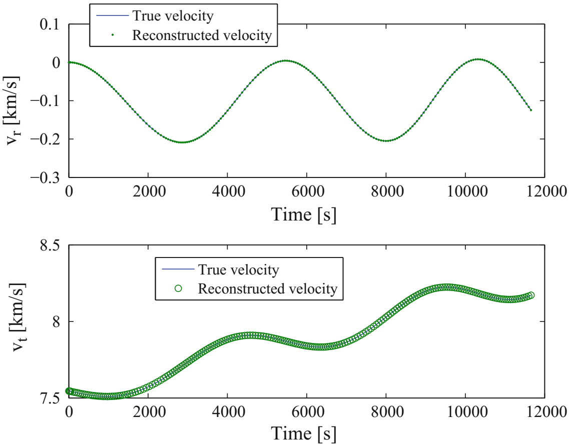


Fig. 4 Prediction over $2T$—velocity components



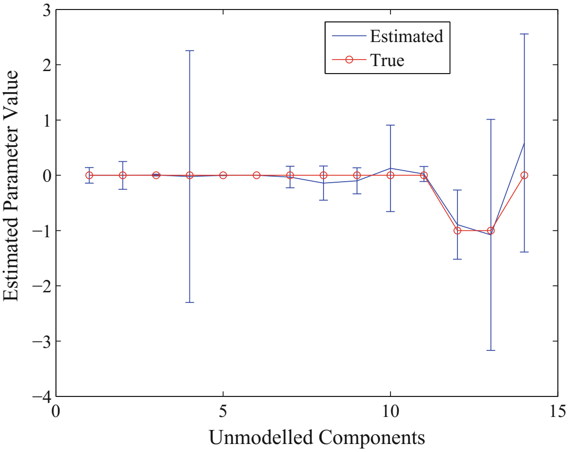


Fig. 5 Example of reconstructed gravity-drag dynamics with confidence intervals

perturbations. The $\mathscr{C}$ space in this case has boundaries $[-10^{-5}, 10^{-5}]$ for all the coefficients. As one can see, even if a linear model is assumed, the prediction of the trajectory is very good over a horizon that is two times the interval over which the measurements are available.

If now one introduces the assumption that measurements are affected by an error, problem (11) needs to be solved under some assumptions on the initial conditions. The assumption in this paper is that the initial conditions are distributed uniformity over a given interval. The size of the confidence interval for the measurements is 10^{-4} of the measured value; accordingly the confidence interval on the initial conditions is set to the same value.

The estimated c parameters are represented in Fig. 5 together with their associated confidence intervals. As one can see, the expected value is close to the true solution. One thing that has to be taken into consideration is that the dynamics that are simulated and measured are the true dynamics, not the linearised equations. Therefore, some components that are not in the linear model might be different from zero.

The other interesting result is that some components are nearly zero for every initial condition while other components, c_4 for example, have a wide variability. This result suggests that some components are irrelevant as they do not contradict the observations no matter which initial conditions are taken, while others substantially affect the evolution of the trajectory. Starting from this first iteration, one can then update the confidence intervals on the parameters c and eventually converge to the correct missing components. Indeed since the uncertain function is based on a truncated series some components of the expansion might absorb the truncation error.

3.3 Chaotic and Hypersensitive Systems

As an example of chaotic dynamics and potentially hypersensitive optimal control problems, we analyse Duffing's equation:

$$\begin{aligned}\dot{v} &= -ax - dv - bx^3 + g\cos(\omega t)\\ \dot{x} &= v\end{aligned} \tag{21}$$

with the following parameters: $a = 1$, $d = 0.1$, $b = 0.1$, $g = 1$, $\omega = 1$. In this case the estimation is over $\Delta t = \pi/4$ with the following expansion:

$$\begin{aligned}\dot{v} &= c_1 + c_2 x + c_3 x^2 + c_4 v + c_5 v^2 + c_6 x^3 + g\cos(\omega t)\\ \dot{x} &= v\end{aligned} \tag{22}$$

In other words, the forcing term is known and is an input to the system, with the system itself a black box that needs to be identified. We first consider the case in which there are enough deterministic observations to compute all the c parameters. In this case we directly solve problem (6) with 6 equispaced measurements. The result is represented in Fig. 6. The dynamics is correctly reconstructed and the optimisation converges to a unique solution.

Considering the case in which there are only 4 deterministic measurements, we solve problem (7) with uncertainty distance metric $J = c^T c$. The result is shown in Fig. 7 for a number of different and independent runs of the optimisation process. As one can see, multiple solutions exist that solve the constraints and minimise the cost function. The lowest cost function is reached when the estimated c parameters approach the correct solution (shown as red circles).

Finally, we consider once again the case in which the measurements are equal to the number of coefficients but are stochastic. In this case the results are represented

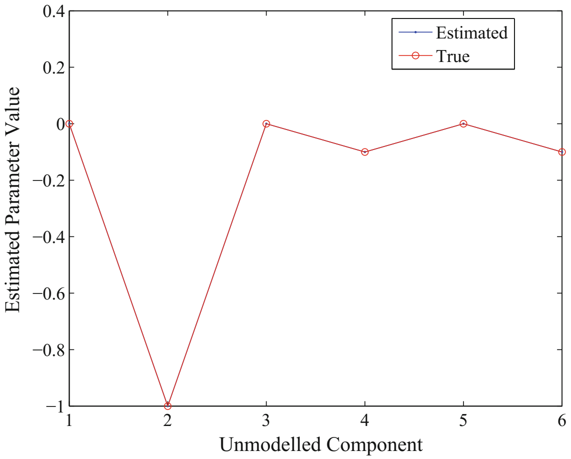


Fig. 6 Example of reconstructed Duffing's dynamics with exact observations and $N_o = 2N$

Fig. 7 Example of reconstructed Duffing's dynamics with exact observations and $N_o < 2N$

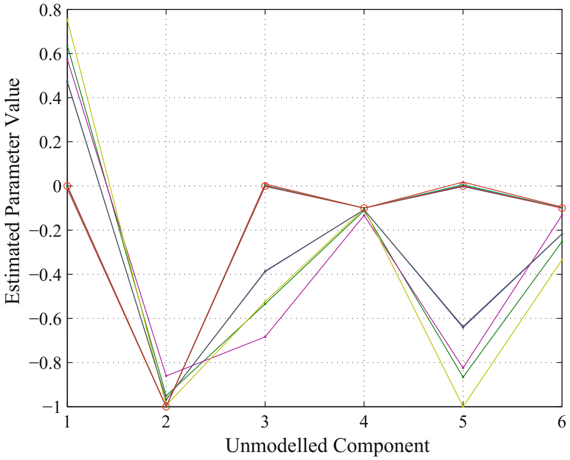
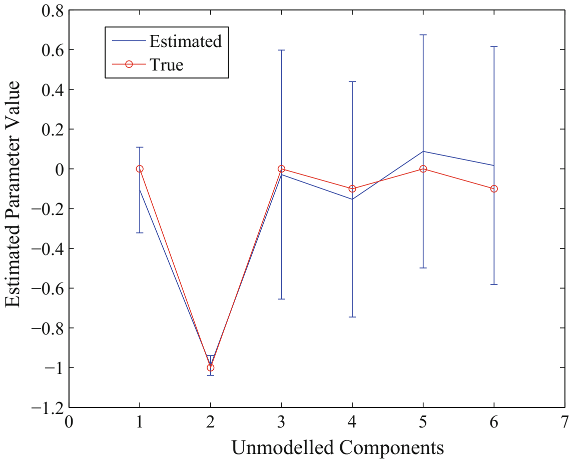


Fig. 8 Example of reconstructed Duffing's dynamics with confidence intervals



in Fig. 8. Even in this case the confidence intervals contain the correct solution, but due to the hypersensitivity of the system an exact reconstruction of the missing components appears to be more difficult as a small variation in the initial conditions can correspond to a significantly different set of parameters. The ϵ value for this case is 10^{-3} of the exact value of the states at the time of the measurement.

4 Reachability Under Model Uncertainty

Once the intervals for each of the parameters c is available, one can solve a reachability problem under uncertainty. One can consider Q to be a stochastic process comparable to a disturbance and the coefficients c to be stochastic variables with probability space $(\Omega, \Gamma, \mathscr{P})$. In this section the problem is limited to reaching the target set given by a sphere centred in $r(s(t_f)) = 0$ where $r(s)$ is the Euclidean distance. The reachability problem can be formulated as follows:

$$\begin{aligned}&\min_u \max_c r(t_f)\\ &s.t.\\ &\dot{s} = f(s) + Q(s, c) + u\\ &s_0 \in \Sigma_0\\ &u \in \mathscr{U}\\ &c \in \Omega \subseteq \mathscr{C}\end{aligned} \tag{23}$$

where $\mathscr{U}$ is the control space, Ω is a subspace of $\mathscr{C}$ defined by the confidence intervals for all the parameters c and Σ_0 is the set of initial conditions. As an example we can consider problem (15) where the confidence intervals for the parameters c are available from the previous identification process and one wants the optimal control u that satisfies:

$$\begin{aligned}&\min_u \max_c x^2(t_f)\\ &s.t.\\ &\ddot{x} = c_0 + c_1 x + c_2 x^2 + c_3 v^2 + u\\ &x_0 \in \Sigma_0\\ &u \in [-0.1, 0.1]\\ &c \in \Omega \subseteq \mathscr{C}\end{aligned} \tag{24}$$

The control is represented with cubic splines collocated at ten regular points in time in the interval $[0, t_f]$. Additionally, the control is bounded to be in the interval $[-0.1, 0.1]$. Problem (24) can be solved with the algorithm presented in [5] for the solution of *minmax* problems. The optimisation converges in a couple of iterations with the result in Fig. 9. Figure 9 show the min-max solution compared to the solution

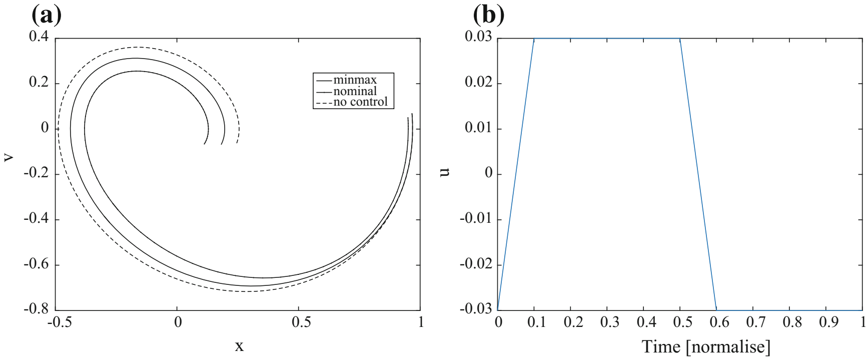


Fig. 9 **a** Phase space representation of the min-max solution; **b** Control profile for the min-max solution

with no control and worst case disturbance and the solution with no disturbance and the control action derived from the solution of problem (24).

5 Final Remarks

The chapter presented an approach to reconstruct unmodelled components in dynamical systems by using hierarchical polynomial expansions. It was shown that the expansion proposed in this paper can be derived from a scalar uncertainty function with the form of a pseudo-potential. Other choices are possible to include effects that can not be reduced to a scalar potential function.

In all the cases analysed in this chapter, the approach provided a good estimation of the missing components in the dynamical model. The next step is to iterate over longer arcs and to update the estimation when new measurements are available. The chapter proposed also the solution of a reachability problem where the unmodelled component are a stochastic disturbance that needs to be controlled by an optimal control action. The problem translates into a min-max optimisation that provides an optimal, yet robust, solution.

Acknowledgements This work is partially supported by the JSPS fellowship number L15548 and by the FP7 MCA Stardust ITN.

References

1. Alessi, M.E., Cicalo, S., Milani, A.: Accelerometer data hanfling for the bepicolombo orbit determination. In: Advances in the Astronautical Society 145 AAS 12-309 1st IAA Conference on Dynamics and Control of Space Systems (2012)
2. Montenbruck, O., Helleputte, T., Kroes, R., Gill, E.: Reduced dynamic orbit determination using gps code and carrier measurements. Aerosp. Sci. Technol. **9**, 261–271 (2005)
3. Vetrisano, M., Colombo, C., Vasile, M.: Asteroid rotation and orbit control via laser ablation. Adv. Space Res. **9** (2015)
4. Giza, D., Singla, P., Jah, M.: An approach for nonlinear uncertainty propagation: Application to orbital mechanics. In: Paper AIAA 2009-6082, AIAA Guidance, Navigation, and Control Conference, Chicago, Illinois, 10–13 Aug 2009
5. Vasile, M.: On the solution of min-max problems in robust optimization. In: Proceedings of EVOLVE 2014, Bejing (2014)

Part VII
Algorithms and Industrial Applications

Improved Archiving and Search Strategies for Multi Agent Collaborative Search

Lorenzo A. Ricciardi and Massimiliano Vasile

Abstract This paper presents a new archiving strategy and some modified search heuristics for the Multi Agent Collaborative Search algorithm (MACS). MACS is a memetic scheme for multi-objective optimisation that combines the local exploration of the neighbourhood of some virtual agents with social actions to advance towards the Pareto front. The new archiving strategy is based on the physical concept of minimising the potential energy of a cloud of points each of which repels the others. Social actions have been modified to better exploit the information in the archive and local actions dynamically adapt the maximum number of coordinates explored in the pattern search heuristic. The impact of these modifications is tested on a standard benchmark and the results are compared against MOEA/D and a previous version of MACS. Finally, a real space related problem is tackled.

1 Introduction

Multi Agent Collaborative Search is a memetic algorithm to solve multi-objective optimisation problems that was proposed some time ago to solve robust optimisation problems in space mission design [1, 2]. In MACS, a population of virtual agents is deployed at random locations in the search space. Each agent locally explores its neighbourhood performing a set of local search actions, also named individual actions. Then the population as a whole performs a set of social actions, to concurrently advance towards the front. An external archive is used to store the current best representation of the Pareto set. In a recent version of MACS, called MACS2, a combination of Pareto dominance and Tchebycheff scalarisation was introduced to select potential improvements towards the Pareto front. Previous studies by Vasile

L. A. Ricciardi · M. Vasile (✉)
Aerospace Centre of Excellence, Advanced Space Concepts Laboratory, University of Strathclyde, Glasgow, UK
e-mail: massimiliano.vasile@strath.ac.uk

L. A. Ricciardi
e-mail: lorenzo.ricciardi@strath.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_26

and Zuiani [3–6] showed the effectiveness of this approach on different benchmark and challenging real problems, testing numerous strategies both for the individual and the social actions. Since then MACS2 was successfully used for the design of space missions for the removal of space debris by means of low-thrust, many revolutions orbits, and for the design of the initial, low-thrust rising phase for the technology demonstrator mission DESTINY. Both are real engineering multi-objective optimisation problems for which no previous solution was known, and involved the concurrent minimisation of fuel consumption, mission time, and, for the DESTINY mission, radiation exposition time.

A thorough analysis of the behaviour of the algorithm, however, has revealed that the archiving procedure was suboptimal as it was not retaining some good isolated non-dominated solutions while keeping solutions in densely populated regions of the Pareto front. The final archive was therefore giving a rather uneven representation of the Pareto front, with poorly distributed solutions. This poor distribution had also an indirect impact on the effectiveness of the search itself, since social actions were exploiting the information in the archive.

In order to address this issue, a new archiving strategy is proposed in this paper. The new strategy, which from now on will be called Energy Based Archiving (EBA), is physically based on the simple idea of minimising the energy of a cloud of points which exert repulsion on each other. Given an initial set with $r + q$ elements, the new archiving procedure selects the subset with r elements with the lowest possible energy. The energy is simply defined as the inverse of the sum of the normalised squared distances of the points in criteria space. Thus, the lowest energy state is associated to the most evenly spread distribution of the points. Note that, the archiving strategy is not specific to MACS, but can be applied to any multi-objective optimisation algorithm, and, as will be shown, can also improve the results obtained by other algorithms.

The paper presents also a set of modified search heuristics that takes advantage of the higher quality of the front stored by the new archiver. One modification improves the exploitation of the information in the archive for the generation of social actions. The other is a dynamic adjustment of the maximum number of coordinates that are explored when local actions are implemented.

The paper is structured as follows: after a preliminary description of multi-objective optimisation in general, the discussion will focus on implementation details and the algorithms will be presented in pseudo-code. Tests will be carried out on the standard benchmark of UF functions from the CEC 2009 competition [7], which are known to have complex Pareto sets. MACS with the new archiving and search heuristics (which for clarity will be called MACS2.1) will be compared with MOEA/D [8], the winner of the CEC 2009 competition, and with MACS2 [5]. It will also be proved that the new archiving algorithm improves the IGD and averaged Hausdorff distance of the solutions given by MOEA/D. The impact of the modified search heuristics will also be investigated, and MACS2.1 and MACS2 will finally be compared on a real challenging space related problem.

2 Problem Formulation

This paper is focused on finding the feasible set of solutions that solves the following problem:

$$\min_{\mathbf{x} \in D} \mathbf{f}(\mathbf{x}) \tag{1}$$

where D is a hyperrectangle defined as $D = \{x_j | x_j \in [b_j^l \; b_j^u] \subseteq \Re, \;\; j = 1, \ldots, n\}$ and $\mathbf{f}$ is the vector function:

$$\mathbf{f} : D \rightarrow \Re^m, \qquad \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})]^T \tag{2}$$

The optimality of a particular solution is defined through the concept of dominance: with reference to problem (1), a vector $\mathbf{y} \in D$ is dominated by a vector $\mathbf{x} \in D$ if $f_l(\mathbf{x}) \leq f_l(\mathbf{y})$ for all $l = 1, \ldots, m$ and there exists k so that $f_k(\mathbf{x}) \neq f_k(\mathbf{y})$. The relation $\mathbf{x} \prec \mathbf{y}$ states that $\mathbf{x}$ dominates $\mathbf{y}$. A decision vector in D that is not dominated by any other vector in D is said to be Pareto optimal. All non-dominated decision vectors in D form the Pareto set D_P and the corresponding image in criteria space is the Pareto front.

Starting from the concept of dominance, it is possible to associate, to each solution in a finite set of solutions, the scalar dominance index:

$$I_d(\mathbf{x}_i) = |\{i^* \mid i, i^* \in N_p \wedge \mathbf{x}_{i^*} \prec \mathbf{x}_i\}| \tag{3}$$

where the symbol $|.|$ is used to denote the cardinality of a set and N_p is the set of the indices of all the solutions. All non-dominated and feasible solutions $\mathbf{x}_i \in D$ with $i \in N_p$ form the set:

$$X = \{\mathbf{x}_i \in D \mid I_d(\mathbf{x}_i) = 0\} \tag{4}$$

The set X is a subset of D_P, therefore, the solution of problem (1) translates into finding the elements of X. If D_P is made of a collection of compact sets of finite measure in $\Re^n$, then once an element of X is identified it makes sense to explore its neighbourhood to look for other elements of X. On the other hand, the set of non dominated solutions can be disconnected and its elements can form islands in D. Hence, multiple parallel exploration can increase the collection of elements of X.

2.1 Tchebycheff Scalarisation

In Tchebycheff approach to the solution of problem (1), a number of scalar optimization problems is solved in the form:

$$\min_{\mathbf{x} \in D} g(\mathbf{f}(\mathbf{x}), \lambda, \mathbf{z}) = \max_{l=1,\ldots,m} \{\lambda_l | f_l(\mathbf{x}) - z_l |\} \tag{5}$$

where $\mathbf{z} = [z_1, \dots, z_m]^T$ is the reference objective vector whose components are $z_l = \min_{\mathbf{x} \in D} f_l(\mathbf{x})$, for $l = 1, \dots, m$, and λ_l is the l-th component of a weight vector λ. By solving a number of problems (5), with different weight vectors, one can obtain different Pareto optimal solutions. Although the final goal is always to find the set X_g, using the solution of problem (5) or index (3) has substantially different consequences in the way samples are generated and selected. In the following, the solution to problem (5) will be used as selection criterion in combination with index (3).

3 Implementation

This section briefly summarises the main features of MACS2, the new archiving procedure and the modified heuristics.

With reference to Algorithm 1, MACS2 starts by initialising a population of n_{pop} agents at random locations within the search domain D with a Latin Hypercube Sampling. Non-dominated agents are copied in the archive A to form the first approximation of the Pareto set. The archive A has specified maximum size max_{arch}. A set of n_λ m-dimensional unit vectors λ_k (defining m directions in criteria space) is then generated, sampled uniformly from a quarter of circle or eighth of a sphere for bi and tri-objective problems or through Latin Hypercube Sampling for higher dimensional problems. The first m λ_k vectors form a base in $\Re^m$, so that the solution vectors that best optimise each individual objective function are always in the final approximation of the Pareto set. Line 8 of Algorithm 1 initialises a utility function (see Zhang et al. [8]) that is used to monitor the progress of each agent. A user-defined fraction p_{social} of agents is specified, and each social agent is then randomly associated to a particular scalarised sub-problem and to the corresponding weight vector λ_k (see Line 9). After initialising the velocity $\mathbf{V}_i$ of each agent (Line 10) the main loop starts (Line 12). Until a maximum number of function evaluations is reached, the agents perform first local search actions, described in the next section, to move towards the Pareto set. A local action is considered successful when it generates a dominating solution or a solution that satisfies the Tchebycheff scalarisation criterion corresponding to a particular sub-problem. After all local actions have been completed, the archive A is updated with all non-dominated solutions. A total of n_{social} agents then perform the social actions described in Sect. 2 and Algorithm 3.

In the new implementation of MACS2 proposed in this paper, when a finite size archive is assumed, the new archiving strategy, described in Sect. 3, is employed to choose which candidates are added to A. Furthermore, as the archive fills up, the number of coordinates scanned by the pattern search local action (see Algorithm 2) is gradually reduced until only one direction at a time is considered. At this point the archive is completely full. This heuristic is motivated by the fact that a well populated and distributed archive contains a lot of information that can be used to generate new samples at a lower cost than a full pattern search. Every n_λ iterations, the utility function will be updated and sub-problems with the lowest utility function will be changed (see Line 18–20 of Algorithm 1).

3.1 Individualistic Actions

In the following we describe the individualistic search actions. Each agent has a repertoire of three different actions, namely: *inertia, pattern search and differential evolution*. Each agent performs each action sequentially until an improvement is registered (i.e. the algorithm generates either a dominant solution or a solution that satisfies Tchebycheff criterion, if the agent is associated to a λ_k). The pseudo-code is given in Algorithm 2.

Inertia
If the previous moves defined a search direction $\mathbf{V}_i$ in parameter space, inertia generates a new sample in the same direction (lines 3–8). The trial position for the $i - th$ agent, $\mathbf{x}_{trial}$ is, defined as:

$$\mathbf{x}_{trial} = \mathbf{x}_i + \alpha \mathbf{V}_i \tag{6}$$

where α is a random number between 0 and 1. If $\mathbf{x}_{trial}$ is outside the admissible domain D, α is contracted with a simple backtracking procedure so that $\mathbf{x}_{trial}$ falls on the boundary of D. In the case a number of components of $\mathbf{x}_i$ lower than n is already equal to either their lower or upper limit and $\mathbf{x}_i + \alpha \mathbf{V}_i$ is outside D, then the corresponding components of $\mathbf{V}_i$ are set to zero before the backtracking procedure is applied. This heuristic is introduced to improve the exploration of the boundary of the search space.

Pattern Search
If inertia gives no improvement or is not performed ($\mathbf{V}_i = 0$), a simple pattern search strategy is implemented. This heuristic changes only one randomly chosen component j of $\mathbf{x}_i$ at a time (lines 14-20). The trial position $\mathbf{x}_{trial}$ is thus equal to $\mathbf{x}_i$, except for the $j - th$ component, which is:

$$x_{trial,j} = x_{ij} + \alpha \Delta_j \rho_i \tag{7}$$

where this time α is a random number between -1 and 1, Δ_j is the difference between the upper and lower boundaries for variable j and ρ_i defines the size of a hyperrectangle centred in $\mathbf{x}_i$. If direction $\alpha \Delta_j \rho_i$ is not successful, the opposite direction $-sign(\alpha)\beta \Delta_j \rho_i$ is attempted with β a random number between 0 and 1. If also this move fails, a new random direction (different from the previous ones) is chosen.

This strategy is repeated until either an improvement is found (i.e. a dominant solution is generated or Tchebycheff criterion is satisfied), or a specified maximum number of directions has been explored. In this version of MACS, the maximum number of directions is dynamically adjusted as

$$max\_dirs = round\left(n - (n-1)\frac{curr\_arch\_size}{max\_arch\_size}\right) \tag{8}$$

where max_dirs is the maximum number of dimensions to scan, n is the number of coordinates, $curr\_arch\_size$ is the current size of the archive and max_arch_size is the specified maximum size of the archive (lines 11–21). If a good sample is found, it is used to compute vector $\mathbf{V}_i$.

Differential Evolution
If pattern search did not lead to an improvement, a differential evolution step is taken, by combining vector $\mathbf{x}_i$ with 3 randomly chosen agents $\mathbf{x_{i_1}}$, $\mathbf{x_{i_2}}$ and $\mathbf{x_{i_3}}$ (lines 24-28). The displacement vector is then given by:

$$\mathbf{dx}_i = \alpha \mathbf{e} \left(\left(\mathbf{x}_i - \mathbf{x}_{i_1} \right) + F \left(\mathbf{x}_{i_2} - \mathbf{x}_{i_3} \right) \right) \tag{9}$$

where α is a random number between 0 and 1, F is a user specified constant and $\mathbf{e}$ is a mask vector whose elements are either 0 or 1 as follows:

$$e_j = \begin{cases} 1, & \text{if } \alpha_2 < CR \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where α_2 is a random number between 0 and 1, and CR is another user specified constant. The trial position for the differential evolution move finally reads:

$$\mathbf{x}_{trial} = \mathbf{x}_i + \mathbf{dx}_i \tag{11}$$

The feasibility check for this new position is performed exactly as for the inertia case: reducing α or suppressing some components of $\mathbf{dx}_i$.

Local Neighbourhood Size Management
If all local actions have failed, the local neighbourhood size ρ_i is reduced by a user defined factor ρ_{contr}. After a user defined maximum number of contractions $\rho_{max,contr}$, ρ_i is reset to ρ_{ini}. Conversely, if one action is successful, ρ_i is increased by a factor ρ_{contr}, up to the maximum value ρ_{ini} (lines 30–37).

3.2 Social Actions

Social actions are implemented following the same principle as in Zuiani and Vasile [5]: a fraction p_{social} of the total population of the agents implements a DE type of heuristic by picking agents either from the population or from the archive (lines 4 or 6). The probability of picking agents from the archive or from the population is determined by:

$$p_{arch\_vs\_pop} = 1 - e^{-\frac{current\_size\_archive}{num\_agents}} \tag{12}$$

In MACS2 each social agent was immediately moved to the trial position if the trial position was satisfying Tchebycheff criterion. The modified heuristic proposed in this paper, instead, updates the archive with all trial vectors that are non-dominated by any other element of the archive. After the archive is updated, each agent performing social actions is then moved to the location of the element of the archive that best improves the corresponding Tchebycheff sub-problem, unless that location is already occupied by another agent (lines 15–22). This new heuristic better exploits the information in the archive and at the same time does not exclude non-dominated trial vectors that do not satisfy the Tchebycheff condition before checking the content of the archive. The pseudo-code for social actions is given in Algorithm 3.

3.3 The New Archiving Strategy

All non-dominated solutions found by the agents are stored in an external archive A. The archiving process is a fundamental part of the optimisation process. Not only does the archive store an approximation of the Pareto set, but it also represents a source of information for the implementation of social actions. Thus, not only is a well distributed archive desirable but it is also a necessity to improve exploration. The algorithm described in this paper attempts to generate the most evenly distributed Pareto front possible with the available set of solution vectors. The heuristic is based on the physical concept of minimisation of an energy. It draws inspiration from the fact that a set of equally charged particles in a sphere will move towards its surface and spread uniformly. In this case, however, the particles are not free to move, but can only occupy specified positions.

Suppose that at iteration k the archive is full and is composed of r elements. Let $\mathbf{y}_i$ and $\mathbf{y}_j$ be the position of element i and j in objective space, then one can define the generalised energy of the archive as:

$$E = \sum_{i=1}^{r} \sum_{j=i+1}^{r} \frac{1}{\left(\mathbf{y}_i - \mathbf{y}_j\right)^T \left(\mathbf{y}_i - \mathbf{y}_j\right)} \tag{13}$$

This energy is simply the inverse of the sum of the squared distances of the points of the archive in the criteria space. Suppose now that there are q non-dominated candidate solutions which also do not dominate any of the elements in the archive. The problem of choosing which candidate substitutes which element of the archive is reformulated as finding the subset of r elements from the set of $r + q$ elements that minimises the energy E.

Note that a direct update of the archive using E is not feasible if r and q are big, because the total number of possible combinations is $\binom{r+q}{r}$. As an alternative, the following procedure is proposed.

Let the archive A be not full. If there is enough space in A to add all the candidates, the archive is simply updated adding those elements (lines 1-2). A symmetric matrix M, containing the reciprocals of the squared distances of all the elements in the archive, is updated (line 4). From M, the total energy of the archive E and a helper vector $\mathbf{E_2}$, are computed (lines 6–7). $\mathbf{E_2}$ is needed to simplify and speed up some computations, as will be explained later. If the archive is full, instead, the following procedure is applied.

Given the elements in the archive A and a set of candidate elements C, for each element in A the energy E is recalculated assuming that that element was replaced by an element in C (line 23). If the lowest variation of E is negative, the element of C that gives that variation and the element in the archive are swapped. If there has been at least one replacement, the whole process is repeated until no more improvements can be detected or a maximum specified number of iterations is exceeded (lines 19-29). In this study, we specified a maximum of 100 iterations. In case the archive is not full but there is not enough space to add all the candidates, the above mentioned algorithm adds, sequentially, the candidates which give the least increase of the total energy of the archive (lines 8–12). No swapping between candidates and agents in the archive is performed in this case, only addition of candidates until the archive is full, and the corresponding update of M and $\mathbf{E_2}$ (lines 13–14). The pseudo-code for the archiving procedure is given in Algorithm 4.

The actual Matlab implementation stores in a symmetric matrix the inverse of all pairwise squared distances between the elements currently in the archive. Deletion, addition and substitution of elements are performed as block matrix operations to save time. In the i-th entry of the $\mathbf{E_2}$ vector is stored the energy the archive would have if element i were removed. This way, the computation of the energy with a substitution of one element is linear in the number of candidates, because the baseline value (i.e. the energy of the archive without replacement) is already stored, and only the contribution of the new candidate needs to be computed. Finally, in order to avoid scaling problems when objectives have very different length scales, a normalisation of the elements in both the candidate set and the archive in criteria space is performed and repeated whenever one of the elements in the archive that optimise each individual objective function is replaced. The overall algorithm is called Energy Based Archiving (EBA).

As an example of the results provided by this archiving strategy, we considered a hypothetical Pareto front with 100 elements and tried to extract the q elements with the EBA algorithm, with the archiving algorithm employed in MACS2 and the one implemented in NSGA-II [9]. Figure 1 shows the results provided by the three archiving strategies for $q = 10$ and $q = 25$. The EBA strategy gives a good spreading of the extracted elements, slightly better than the one obtained with the strategy in MACS2 and much better than that obtained by the strategy implemented in NSGA-II. The hypothetical Pareto front is composed of a set of random samples taken from the ZDT4 Pareto front. All the algorithms extract the same number of elements from this set. The IGD produced by each algorithm is shown in Table 1.

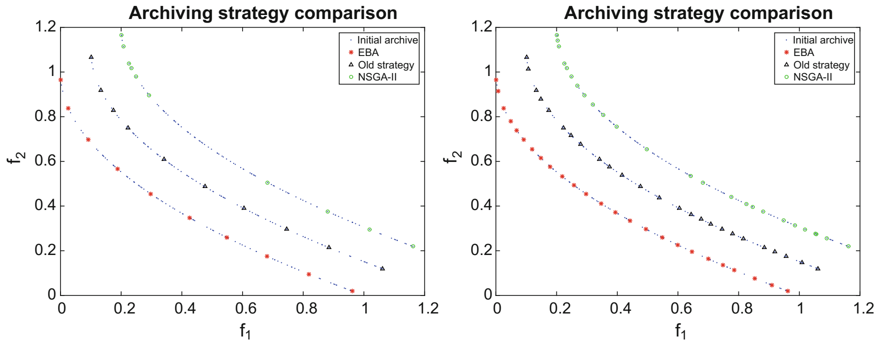
The EBA strategy requires two sets of operations corresponding to two steps: the fill-in of the archive and computation of the energy E, and its minimisation. The two steps never occur at the same time when the archive is updated. The computation of each reciprocal of squared distance, in (13), for an m dimensional space, requires $3m$ operations: m differences of homologous coordinates, m squares of differences, $m-1$ sums of squares and 1 reciprocal of sums. If the reciprocal of pairwise squared

Table 1 IGD of the different archiving strategies

	MACS 2.1	MACS2	NSGA-II
	Archiver	Archiver	Archiver
10 points	3.68e−2	3.90e−2	9.44e−2
25 points	1.52e−2	1.55e−2	2.44e−2

distances of the r elements of the archive is stored in an r by r symmetric matrix M with zero diagonal elements, the number of reciprocal of squared distances to be computed is $\frac{r(r-1)}{2}$, for a total of $\frac{3mr(r-1)}{2}$ operations. With this matrix, the computation of the total energy E of the archive requires the sum of the elements of the upper (or lower) triangular part of the matrix, for a total of $\frac{r(r-1)}{2}$ sums. Starting from the energy E and matrix M the already mentioned r components vector $\mathbf{E_2}$ is computed. $\mathbf{E_2}$ contains in its i-th entry the energy the archive would have if element i were excluded from the archive. The computation of the elements of this vector is conveniently performed by subtracting the sum of all elements of the i-th row of M from the energy E. Thus, a total of r^2 operations is required: 1 subtraction and $r-1$ sums for each component of $\mathbf{E_2}$. This completes the fill-in step of the archive, for a total of $\frac{3mr(r-1)}{2} + \frac{r(r-1)}{2} + r^2 \sim \mathscr{O}(mr^2)$ operations. Note that during a run of MACS2.1 the archive grows gradually, so the construction of the matrix M, energy E and vector $\mathbf{E_2}$ is performed incrementally, rather than all at once.

The energy minimisation step requires the computation of the reciprocal of square distances from each element of A to each candidate in C, for a total of rq combinations or $3mrq$ operations. At this point, the energy the archive would have if element i were substituted with the candidate j is computed: this is conveniently performed by summing $\mathbf{E_2}(i)$ to the reciprocal of the square distances from the j-th candidate to each other element in the archive, for a total of r^2 operations: r sums for each of element of the archive. Now, the minimum energy over all possible combinations of candidates E_{new} is compared against the energy of the archive E. Suppose all the tentative energies are stored in an r by q matrix and its lower value entry E_{new} is located in the position (i^*, j^*). If E_{new} is lower than E, then a new minimum is found, and the candidate in position j^* substitutes the element in position i^*. At this point, it is required to update all elements of the i-th row (or column) of the matrix M with the new reciprocal squared distances. This requires again $3m(r-1)$ operations, but can be avoided with proper bookkeeping (i.e. if the matrix containing the reciprocal of square distances from each candidate to each element of the archive is stored). Finally, the update of the vector $\mathbf{E_2}$ is performed exactly as before. If the energy minimisation step is performed n_{it} times, the total number of operations is $3mrq + n_{it}\left(r^2 + r^2\right)$. Thus, the total cost of the EBA archiving algorithm is $\mathscr{O}\left(3mr^2/2\right)$ for the fill-in step, and $\mathscr{O}(n_{it}r^2)$ for the minimisation step (assuming $q \leq r$). In order to verify the computational complexity of the EBA strategy, we performed a statistical analysis on the dependence of n_{it} on q by sampling the Pareto front of the ZDT4 benchmark function with 100 points plus an increasing number of additional candidates. 200



(a) 10 points selected

(b) 25 points selected

Fig. 1 Outcomes of different archiving strategies from the same initial archive. The fronts have been shifted to enhance the comparison

independent runs were made for each number of additional candidates, and an $\mathcal{O}(q^{\frac{1}{2}})$ relation was discovered. Thus, the average cost of the minimisation step was found to be $\mathcal{O}(r^2 q^{\frac{1}{2}})$. It is also important to note that the computational complexity is linear in the number of dimensions of the Pareto front, making EBA a promising method for many-objective optimisation problems.

Algorithm 1 MACS2

```
Set $n_{feval,max}$, $max_{arch}$, $n_{pop}$, $p_{social}$, $F$, $CR$, $\rho_{ini}$, $\rho_{contr}$, $\rho_{max,contr}$
Set $n_{social} = n_{pop} p_{social}$
Set iteration counter $h = 0$
Initialise population $P_h, n_{feval} = 0$
Initialise neighbourhood size $\rho_i = \rho_{ini}$ $\forall i \in \{1, .., n_{pop}\}$
Insert the non-dominated elements of $P_0$ in the archive $A$
Initialise $n_\lambda$ vectors $\lambda_k$ for $k \in \{1, .., n_\lambda\}$ such that $||\lambda_k|| = 1$
Initialise utility function $U_k = 1$ $\forall k \in \{1, .., n_\lambda\}$
Select $n_{social}$ active sub-problems to follow
Initialise the vector of agents' velocities $\mathbf{V}_i = 0$ $\forall i \in \{1, .., n_{pop}\}$
**while** $n_{feval} < n_{feval,max}$ **do**
  h=h+1
  update number of directions to scan in pattern search
  Perform local actions through Algorithm 2
  Update archive $A$ with non dominated elements through Algorithm 4
  Perform social actions through Algorithm 3
  Update archive $A$ with non dominated elements through Algorithm 4
  **if** $mod(h, n_{social}) == 0$ **then**
    Update utility function $U_k$ and the $n_{social}$ active sub-problems
  **end if**
**end while**
```

Algorithm 2 Individualistic actions

```
for $i = 1 : n_{pop}$ do
    Set improved=FALSE
    if $||\mathbf{V}_i|| \neq 0$ then
        Perform Inertia move
        Evaluate move
        if successful then
            set improved=TRUE
        end if
    end if
    if not improved then
        counter=0
        while counter$\leq max\_pat\_search\_dirs$ & not improved do
            counter=counter+1
            Pick random direction
            Perform Pattern Search
            Evaluate move
            if successful then
                set improved=TRUE
                set $\mathbf{V}_i = x_{i,old} - x_i$
            end if
        end while
    end if
    if not improved then
        Perform Differential Evolution
        Evaluate move
        if successful then
            set improved=TRUE
        end if
    end if
    if not improved then
        Contract $\rho_i$
        if $\rho_i$ has contracted more than $\rho_{max,contr}$ times then
            $\rho_i = \rho_{ini}$
        end if
    else
        De-contract $\rho_i$ unless this would cause $\rho_i$ to be greater than $\rho_{ini}$
    end if
end for
```

Algorithm 3 Social actions

```
choose random number $r$ between 0 and 1
compute $p = 1 - e^{-\frac{curr\_arch\_size}{num\_agents}}$
if $r \leq p$ then
    perform DE between social agents and random points from archive
else
    perform DE between social agents and random points from current population
end if
add candidate solutions in archive through Algorithm 4
if there are at least as many agents in the archive as objective functions then
    if there's exactly as many agents in the archive as objective functions then
        $n_{move}$ = num objective functions
    else
        $n_{move}$ = min(num agents in archive, num agents performing social actions)
    end if
    create pool of $n_{move}$ agents to be moved. Agents following exclusively one of the objectives are always chosen
    for all agents in pool do
        find the agent in archive better solving current agent's sub-problem
        if archive position is better than current position then
            move current agent to that position
            hide that position in archive for current run of social actions, to prevent multiple agents moving in the same position
        end if
    end for
end if
```

Algorithm 4 Energy Based Archiving

```
if there's room for all candidates in archive then
    Add them to the archive
    for all candidates do
        Update the symmetric matrix $M$ containing the reciprocal of the squared distance of each pair of elements
    end for
    Update the total energy $E$ of the archive
    Update the vector $\mathbf{E_2}$
else
    if only some candidates can be added then
        while archive is not full do
            Choose the candidate which gives the least possible addition of energy to the archive and add it
            Update $M$, $E$ and $\mathbf{E_2}$
        end while
    else if the archive is full then
        Set improved=TRUE
        iterations=0
        while improved and $iterations < max_{it}$ do
            improved=FALSE
            iterations=iterations+1
            Create a matrix containing the energy that the archive would have if each element of the archive were substituted with each candidate
            Locate the minimum entry $E_{new}$ of this matrix
            if $E_{new} < E$ then
                $E_{new}$ is at position $(i^*, j^*)$
                Swap candidate $j^*$ with element $i^*$
                Set improved=TRUE
                Update $E$, $M$ and $\mathbf{E_2}$
            end if
        end while
    end if
end if
```

4 Test Cases

The test set used in this paper is a mix of the first seven UF functions proposed in the CEC2009 competition on multi-objective optimisation, the function ZDT4 proposed by Zitzler et al. [10] and a real case of space mission design.

4.1 CEC 2009 UF Functions

The UF functions have a complex Pareto set and are a good benchmark to test the archiving procedure. The version of MACS2 with the new archiving procedure and the modified heuristics, called MACS2.1 from now on, was tested and compared against the version of MOEA/D that won the CEC2009 competition [8] and against MACS2.

On the UF test set, each algorithm was run 200 times for each of the functions UF1-7 on a Linux workstation with 8 GB of RAM and an Intel i7-4790 cpu. The settings for MACS2.1 are reported in Table 2 while for MOEA/D the parameters suggested by its authors in [8] were used. The algorithms are compared against the Inverse Generational Distance (IGD) metric, which was used to rank the solutions in the CEC2009 competition, and against the Averaged Hausdorff distance. Both metrics are described and extensively analysed in [11]. As pointed out by Schutze

Table 2 Settings for MACS, CEC problems

$n_{feval,max}$	n_{pop}	ρ_{ini}	F	CR	p_{social}	ρ_{contr}	$\rho_{max,contr}$
300000	150	1	0.9	0.9	0.2	0.5	5

Table 3 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MACS2.1 and MOEA/D, CEC2009 problems. Also reported the unsigned Wilcoxon test results

Problem	MACS2.1	MOEA/D	Wilcoxon	MACS2.1	MOEA/D	Wilcoxon
	IGD	IGD	Test IGD	Hausdorff	Hausdorff	Test hausdorff
UF1	**4.09e−3** (9.58e−9)	4.41e−3 (1.69e−8)	3.70e−59	**1.65e−2** (5.53e−5)	1.12e−1 (6.16e−2)	3.14e−11
UF2	**4.43e−3** (1.23e−7)	6.24e−3 (1.57e−6)	3.70e−59	**2.09e−2** (4.41e−5)	7.48e−2 (9.46e−3)	4.52e−53
UF3	1.84e−2 (1.09e−5)	**7.16e−3** (2.47e−5)	3.70e−59	1.46e−1 (2.61e−2)	**6.38e−2** (1.83e−2)	5.08e−34
UF4	**2.93e−2** (7.50e−7)	6.14e−2 (2.50e−5)	3.70e−59	**4.99e−2** (2.74e−5)	1.11e−1 (3.17e−4)	4.83e−67
UF5	**5.80e−2** (5.58e−5)	2.98e−1 (7.45e−3)	3.70e−59	**1.32e−1** (9.56e−4)	7.96e−1 (2.20e−1)	4.83e−67
UF6	**2.74e−2** (6.10e−5)	2.68e−1 (4.34e−2)	3.70e−59	**8.86e−2** (2.06e−3)	6.27e−1 (1.14e−1)	7.03e−67
UF7	**4.15e−3** (5.61e−8)	4.77e−3 (3.17e−6)	1.46e−34	**2.96e−2** (9.40e−4)	1.67e−1 (6.83e−2)	1.26e−08

et al. [11], the IGD metric is sensistive to the number of elements in the reference Pareto front and in the computed one. Hence the inclusion of the Averaged Hausdorff distance in this comparison. Mean and variance of the IGD and Averaged Hausdorff distance for each problem and algorithm are reported in Table 3, together with result of the Wilcoxon hypothesis test. In 6 of the 7 cases analysed in this paper, the results obtained by MACS2.1 have lower mean IGD and mean Averaged Hausdorff distance, and the variances of those metrics are 1 to 3 orders of magnitude lower for MACS2.1, meaning that the results or MACS2.1 are much more repeatable. The low values of the Wilcoxon test confirm that the underlying distributions of the metrics are indeed different.

To better appreciate the effect of the archiver, MACS2.1 was then run with the same settings but with the archiving strategy employed by MACS2 (as described in [5]), while the final results of MOEA/D where filtered with the EBA algorithm instead of using the filter employed by MOEA/D. Tables 4 and 5 summarise the results of this test. The EBA strategy improved the quality of the Pareto front found by MOEA/D in 4 cases without worsening the others, and improved the results of MACS2.1 in 3 cases with no significant variation in the other cases. The amount of the improvement depends on the quality and size of the the archive: a closer examination of the UF5 and UF6 cases showed that none of the 200 archives had

Table 4 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MOEA/D with EBA archiving and standard MOEA/D, CEC2009 problems. Also reported the Wilcoxon test result

Problem	MOEA/D+EBA	MOEA/D	Wilcoxon	MOEA/D+EBA	MOEA/D	Wilcoxon
	IGD	IGD	Test IGD	Hausdorff	Hausdorff	Test hausdorff
UF1	**4.11e−3** (1.71e−8)	4.41e−3 (1.69e−8)	8.16e−54	1.11e−1 (6.16e−2)	1.12e−1 (6.16e−2)	3.67e−1
UF2	**6.00e−3** (1.58e−6)	6.24e−3 (1.57e−6)	2.53e−04	7.48e−2 (9.46e−3)	7.48e−2 (9.46e−3)	9.94e−1
UF3	**6.88e−3** (2.54e−5)	7.16e−3 (2.47e−5)	2.54e−07	6.30e−2 (1.83e−2)	6.38e−2 (1.83e−2)	1.19e−1
UF4	6.13e−2 (2.49e−5)	6.14e−2 (2.50e−5)	7.92e−01	1.11e−1 (3.12e−4)	1.11-e1 (3.17e−4)	8.45e−1
UF5	2.98e−1 (7.45e−3)	2.98e−1 (7.45e−3)	9.97e−01	7.96e−1 (2.20e−1)	7.96e−1 (2.20e−1)	9.98e−1
UF6	2.68e−1 (4.34e−2)	2.68e−1 (4.34e−2)	9.95e−01	6.27e−1 (1.14e−1)	6.27e−1 (1.14e−1)	9.96e−1
UF7	**4.48e−3** (3.19e−6)	4.77e−3 (3.17e−6)	1.24e−32	1.67e−1(6.83e−2)	1.67e−1 (6.83e−2)	9.08e−1

Table 5 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MACS2.1 with EBA archiving versus MACS2.1 without EBA archiving on the CEC2009 problems. Also reported the Wilcoxon test result

Problem	MACS2.1	MACS2.1 NO EBA	Wilcoxon	MACS2.1	MACS2.1 NO EBA	Wilcoxon
	IGD	IGD	Test IGD	Hausdorff	Hausdorff	Test hausdorff
UF1	**4.09e−3** (9.58e−9)	4.32e−3 (1.04e−8)	1.88e−54	1.65e−2 (5.53e−5)	1.61e−2 (1.49e−5)	9.88e−1
UF2	**4.43e−3** (1.23e−7)	4.70e−3 (6.56e−8)	3.84e−25	2.09e−2 (4.41e−5)	2.09e−2 (4.13e−5)	8.49e−1
UF3	1.84e−2 (1.09e−5)	1.85e−2 (9.72e−6)	6.56e−01	1.46e−1 (2.61e−2)	1.41e−1 (2.25e−2)	6.45e−1
UF4	2.93e−2 (7.50e−7)	2.92e−2 (1.03e−6)	5.29e−01	4.99e−2 (2.74e−5)	5.04e−2 (3.13e−5)	2.46e−1
UF5	5.80e−2 (5.58e−5)	5.84e−2 (5.63e−5)	7.84e−01	1.32e−1 (9.56e−4)	1.35e−1 (9.28e−4)	2.19e−1
UF6	2.74e−2 (6.10e−5)	2.66e−2 (3.71e−5)	2.58e−01	8.86e−2 (2.06e−3)	9.61e−2 (2.64e−3)	8.12e−2
UF7	**4.15e−3** (5.61e−8)	4.49e−3 (6.01e−8)	1.40e−35	2.96e−2 (9.40e−4)	2.79e−2 (5.80e−5)	1.52e−1

Table 6 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MACS2.1 versus MACS2 on the CEC2009 problems. Also reported the Wilcoxon test result

Problem	MACS2.1	MACS2	Wilcoxon	MACS2.1	MACS2	Wilcoxon
	IGD	IGD	Test IGD	Hausdorff	Hausdorff	Test hausdorff
UF1	**4.09e-3** (9.58e−9)	4.39e−3 (2.48e−8)	1.50e−59	**1.65e−2** (5.53e−5)	2.80e−2 (3.14e−4)	9.26e−35
UF2	**4.43e−3** (1.23e−7)	4.49e−3 (1.32e−8)	6.33e−09	2.09e−2 (4.41e−5)	**1.57e−2** (7.61e−6)	5.08e−24
UF3	**1.84e−2** (1.09e−5)	2.41e−2 (4.98e−6)	8.25e−50	1.46e−1 (2.61e−2)	**6.75e−2** (3.60e−4)	1.07e−29
UF4	2.93e−2 (7.50e−7)	**2.63e−2** (2.96e−7)	2.15e−66	4.99e−2 (2.74e−5)	**4.43e−2** (2.01e−5)	1.21e−26
UF5	5.80e−2 (5.58e−5)	**5.29e−2** (4.81e−5)	2.81e−11	1.32e−1 (9.56e−4)	**1.22e−1** (1.09e−3)	2.09e−05
UF6	**2.74e−2** (6.10e−5)	3.41e−2 (1.06e−4)	7.30e−17	**8.86e−2** (2.06e−3)	1.02e−1 (2.60e−3)	2.03e−04
UF7	**4.15e−3** (5.61e−8)	6.54e−3 (4.96e−6)	3.84e−66	**2.96e−2** (9.40e−4)	4.93e−2 (4.37e−4)	5.68e−39

100 non-dominated elements, hence EBA simply gave the same result as the strategy implemented in MACS2.

For UF4 the high IGDs are caused by a relatively high distance between the computed front and the true one, more than by a poor distribution of the points, while for UF3 there is a relative lack of points in the upper left region of the Pareto front. For this comparison, the Averaged Hausdorff distance does not show any relevant improvement. This is due to the fact the Averaged Hausdorff distance penalizes the outliers (as clearly stated in [11]), and as such is a worst case measure. The EBA algorithm was conceived to maximise the spreading of the overall solution, but cannot guarantee the worst case distance from each point of the reference front to each point on the computed one, so the observed metrics are not surprising.

We then compared MACS2.1 versus MACS2 [5]. As it can be seen from Table 6, MACS2.1 improves over MACS2 in 5 out of 7 cases for the IGD, although it produces worse results on UF4 and UF5. The Averaged Hausdorff distance for this case favour MACS 2.1 only in 3 over 7 cases.

To better understand which heuristic is contributing to give the different results of MACS2.1 with respect to MACS2, we compared the results obtained by MACS2.1 with dynamic adjustment of the maximum number of directions in the pattern search, against MACS2.1 with a fixed maximum number of direction equal to $2n$. This because in MACS2 the number of directions scanned by pattern search is fixed and equal to $2n$. Results in Table 7 show that in all cases except for UF4 and UF5, the dynamic adjustment of the maximum number of directions scanned by pattern search has a positive effect on the IGD. The comparison of the Averaged Hausdorff distance shows a substantial parity between the two approaches, meaning that the dynamic

Table 7 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MACS2.1 with EBA archiving and dynamic setting of maximum number of coordinates for pattern search versus MACS2.1 with EBA archiving and static setting of maximum number of coordinates for pattern search on the CEC 2009 problems. Also reported the Wilcoxon test result

Problem	MACS2.1	MACS2.1 static	Wilcoxon	MACS2.1	MACS2.1 static	Wilcoxon
	IGD	IGD	Test IGD	Hausdorff	Hausdorff	Test hausdorff
UF1	**4.09e−3** (9.58e−9)	4.40e−3 (1.98e−8)	5.46e−61	**1.65e−2** (5.53e−5)	2.54e−2 (1.29e−4)	4.74e−39
UF2	**4.43e−3** (1.23e−7)	4.47e−3 (2.14e−8)	2.48e−06	2.09e−2 (4.41e−5)	**1.60e−2** (1.24e−5)	2.64e−21
UF3	**1.84e−2** (1.09e−5)	2.51e−2 (1.09e−5)	4.05e−09	1.46e−1 (2.61e−2)	1.36e−1 (1.43e−2)	4.67e−01
UF4	2.93e−2 (7.50e−7)	**2.66e−2** (3.42e−7)	2.20e−65	4.99e−2 (2.74e−5)	**4.54e−2** (2.26e−5)	2.70e−20
UF5	5.80e−2 (5.58e−5)	**5.47e−2** (4.98e−5)	9.51e−06	1.32e−1 (9.56e−4)	**1.24e−1** (7.51e−4)	1.21e−03
UF6	**2.74e−2** (6.10e−5)	3.04e−2 (7.43e−5)	1.27e−05	**8.86e−2** (2.06e−3)	9.78e−2 (2.13e−3)	3.48e−03
UF7	**4.15e−3** (5.61e−8)	5.08e−3 (1.59e−6)	9.58e−66	**2.96e−2** (9.40e−4)	4.76e−2 (2.33e−4)	9.84e−48

strategy does not improve the position of the outliers. It can be also be appreciated that in the static case, both the IGD and the Averaged Hausdorff distance associated to the UF1 to UF5 cases are close to the values obtained by MACS2. This is not surprising since in MACS2.1 DE is performed after pattern search, so if pattern search scans all possible coordinates it will most probably find an improvement and thus DE will not be performed. This also means that in the UF6 and UF7 cases some other heuristic of MACS2.1 is instead contributing.

With the same rationale as the previous analysis, we compared MACS2.1 with the new implementation of social moves against MACS2.1 with the old implementation of the social moves. Table 8 shows that the IGD of the new version is better than the old version in 4 over 7 cases, statistically the same in 1 case and worse in 2 cases, while the Averaged Hausdorff distance of the new social moves is better in 3 cases, statistically the same in 1 case and worse in 3 cases. Thus, this modification does not seem to give a clear contribution. However, further studies are required to assess the interaction of all the combinations of the proposed modifications, especially between the update of the sub-problems through the utility function and the social actions.

As a final rigorous performance test for the CEC cases, we compared the success rate of each algorithm on each of the UF functions. The success rate is defined as the number of runs in which the IGD falls below a given threshold over the total number of runs. Thresholds were chosen to differentiate the results as much as possible but using rather simple values. Table 9 summarises the results. As it is evident, MACS2.1

Table 8 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MACS2.1 versus MACS2.1 with old social moves on the CEC2009 problems. Also reported the Wilcoxon test result

Problem	MACS2.1	MACS2.1 old social	Wilcoxon	MACS2.1	MACS2.1 old social	Wilcoxon
	IGD	IGD	Test IGD	Hausdorff	Hausdorff	Test Hausdorff
UF1	**4.09e−3** (9.58e−9)	4.14e−3 (2.63-e8)	1.96e−04	**1.65e−2** (5.53e−5)	2.53e−2 (1.54e−3)	2.71e−06
UF2	4.43e−3 (1.23e−7)	**4.11e−3** (1.83e−8)	8.44e−36	2.09e−2 (4.41e−5)	**1.54e−2** (9.18e−6)	2.36e−27
UF3	**1.84e−2** (1.09e−5)	1.95e−2 (4.03e−6)	2.95e−04	1.46e−1 (2.61e−2)	**8.41e−2** (1.24e−2)	5.16e−31
UF4	2.93e−2 (7.50e−7)	**2.76e−2** (7.04e−7)	5.87e−48	4.99e−2 (2.74e−5)	**4.64e−2** (2.39e−5)	2.22e−15
UF5	5.80e−2 (5.58e−5)	5.75e−2 (5.01e−5)	5.33e−01	1.32e−1 (9.56e−4)	1.30e−1 (8.26e−4)	4.49e−01
UF6	**2.74e−2** (6.10e−5)	3.05e−2 (6.63e−5)	2.47e−05	**8.86e−2** (2.06e−3)	1.01e−1 (2.67e−3)	5.35e−05
UF7	**4.15e−3** (5.61e−7)	4.60e−3	4.13e−39	**2.96e−2** (9.40e−4)	3.31e−2 (7.54e−5)	3.39e−13

Table 9 Success rates for the IGD of the CEC2009 functions for all the tested algorithms and their variants

%IGD <	MACS2.1	MACS2.1	MOEA/D	MOEA/D	MACS2.1	MACS2.1	MACS2
τ		NO EBA		+ EBA	Static pat	Old social	
UF1 ($\tau =$ 4.5e−3)	100	94.5	85.0	98.0	76.0	99.0	79.5
UF2 ($\tau =$ 5.0e−3)	93.5	88.5	0.5	13.0	99.5	100	100
UF3 ($\tau =$ 2.0e−2)	68.5	67.5	95.0	95.0	9.0	61.0	3.5
UF4 ($\tau =$ 3.0e−2)	78.5	78.5	0	0	100	100	100
UF5 ($\tau =$ 5.0e−2)	16.0	12.0	0	0	25.0	15.0	36.5
UF6 ($\tau =$ 3.0e−2)	70.0	74.5	0	0	56.0	53.5	36.5
UF7 ($\tau =$ 4.5e−3)	92.0	60.0	64.5	92.0	1.5	40.5	1.5

has overall good performance, outperforming MOEA/D in all cases except for UF3. The introduction of EBA in MOEA/D can improve its results by 10–30% on some problems. MACS2.1 is also generally better than MACS2: although for UF2, UF4 and UF5 the latter has a success rate 20% higher than the former, MACS2.1 is more than 20% better in the other problems, up to 90% better for UF7. In MACS2.1, an overall 5 to 30% improvement is given by the EBA archiving strategy, while the dynamic setting of the maximum number of coordinates can improve results up to 90% or worsen them up to 15%. Similarly, the new implementation of the social moves can improve results up to 50% or worsen them up to 20%. Overall, the proposed version of MACS2.1 seems to have more consistent results on the entire set of problems, never falling behind by more than 25% over any other algorithm on any problem.

4.2 ZDT4 and 3 Impulse Problem

To further test the capabilities of MACS2.1 we run it 200 times on the ZDT4 test function and on a real space trajectory optimisation problem. In the space trajectory design problem the goal is to optimise three impulsive manoeuvres to transfer a spacecraft from a circular Low Earth Orbit, with a radius of 7000 km, to a circular Geostationary orbit, with a radius of 42000 km (for further details on the problem the interested reader can refer to [3]). The motivation behind the choice of the ZDT4 and the 3 impulse test case is that both of them are characterised by many local Pareto fronts. The settings for MACS2.1 are reported in Table 10, 200 solutions per run were maintained in the archive. The performance of MACS2.1 was compared against the perfomance of MACS2, whose settings were specified as in [5].

Note that, for the 3 impulse case the true Pareto front is unknown, thus for self consistence a global Pareto front was extracted from all the 400 runs and used as a reference Pareto front for the calculation of all the metrics.

Tables 11 and 12 report the metrics computed for both the ZDT4 and 3 impulse problem and the corresponding success rates. On the ZDT4 case MACS2.1 outperforms MACS2.

The 3 impulse case gives less clear results instead. Although many interesting areas of the global Pareto front are due to MACS2.1, the mean IGD associated to its solutions is higher than the mean IGD of the solutions computed by MACS2. This is due to the fact (see Fig. 2) that MACS2 is contributing to the global Pareto front with twice as many points than MACS2.1, and all those points are concentrated in a central area, while the contribution from MACS2.1 is as expected more widespread and surprisingly a bit scarce in the central area. Thus, the typical run of MACS2 will generate many points close to the over represented region, while the typical run of MACS2.1 will generate less points in that area but more widely spread points, and this results in the averaged IGD of MACS2.1 to be higher than that of MACS2. The Averaged Hausdorff distance instead does not suffer from this kind of bias, thus the lower value of this metric is associated to the fronts computed by MACS2.1.

Table 10 Settings of MACS2.1 on the 3 impulse and ZDT4 problems (in brackets)

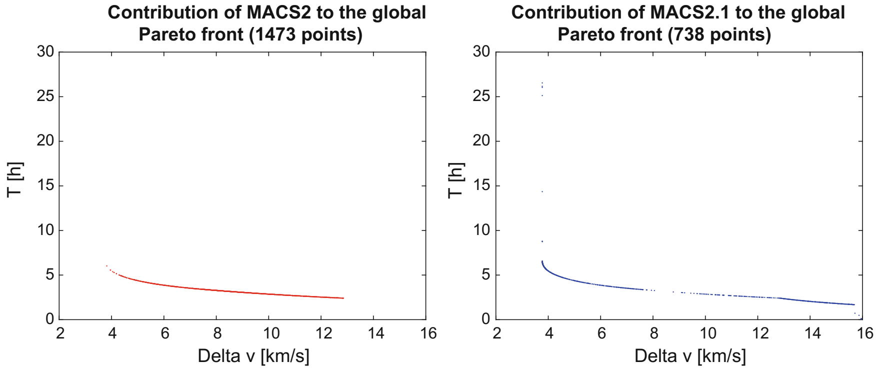
$n_{feval,max}$	n_{pop}	ρ_{ini}	F	CR	p_{social}	ρ_{contr}	$\rho_{max,contr}$
30000	10	1	0.9	0.9	1	0.5	5
(15000)	(10)	(1)	(0.9)	(0.9)	(1)	(0.5)	5

Table 11 Mean (variance in brackets) for the IGD and averaged Hausdorff distances for MACS2.1 versus MACS2 on zdt4 and triple impulse problems. Also reported the Wilcoxon test result

Problem	MACS2.1	MACS2	Wilcoxon	MACS2.1	MACS2	Wilcoxon
	IGD	IGD	Test	Hausdorff	Hausdorff	Test
zdt4	**7.6e4-3** (1.05e−4)	8.01e−1 (2.48e−1)	3.00e−64	**2.52e−2** (7.59e−4)	2.62e+0 (3.60e+0)	1.20e−62
3 imp	2.78e−1 (2.81e−2)	**1.17e−2** (1.07e−3)	1.32e−43	**2.89e+2** (7.11e+3)	3.45e+2 (6.05e+3)	7.41e−31

Table 12 ZDT4 and triple impulse success rates for MACS2 and MACS2.1 and the various metrics

% IGD <	MACS2	MACS2.1	% Hausdorff <	MACS2	MACS2.1
τ			τ		
zdt4 ($\tau =$ 1e−2)	1.5	83.5	zdt4 ($\tau =$ 5e−2)	3.0	97.5
3imp ($\tau =$1e−1)	29.5	3.0	3imp ($\tau =$ 3e+2)	5.0	82.5



(a) Contribution of MACS2

(b) Contribution of MACS2.1

Fig. 2 Contribution of the different algorithms to the global Pareto front

5 Conclusions

In this paper we presented a new archiving strategy and some modified search heuristics for MACS2. The results computed by the new algorithm, called MACS2.1, are overall better than those computed by MOEA/D on the UF test set. MACS2.1 outperformed also MACS2 in 5 over 7 of the UF functions and on the ZDT4 test case. In the 3 impulse case, MACS2.1 contributes with a wide spread of points to the Pareto front not concentrated in the central area densely covered by MACS2. The better IGD scored by MACS2 in this case can be explained by an uneven distribution of points on the global Pareto front. This is confirmed by the fact that MACS2.1 shows a better Averaged Hausdorff distance, a metric which does not resent from this kind of bias.

The effectiveness of the archiving strategy at extracting well spread Pareto fronts was shown both by comparing the results obtained by MACS2.1 with and without EBA, and by comparing the results obtained by MOEA/D with and without EBA.

The effect of the modified strategies employed in the current version of MACS was also investigated. From the results of our tests, we can conclude that the strategy to dynamically change the number of directions scanned by the pattern search algorithm can have a deep positive impact on some problems and a slightly negative impact on others. The new implementation of the social actions gives less clear results instead: further studies are required to assess how it is influenced by the choice of the sub problems, and how do these new strategies interact in general. Interesting future research options involve the use of more sophisticated local search heuristics in combination with a Monotonic Basin Hopping strategy already successfully tested in [6].

References

1. Zuiani, F., Vasile, M.: Preliminary design of debris removal missions by means of simplified models for low-thrust, many-revolution transfers. Int. J. Aerosp. Eng. (2012)
2. Zuiani, F., Kawakatsu, Y., Vasile, M.: Multi-objective optimisation of many-revolution, low-thrust orbit raising for destiny mission. In: 23rd AAS/AIAA Space Flight Mechanics Conference (2013)
3. Vasile, M., Zuiani, F.: Multi-agent collaborative search: an agent-based memetic multi-objective optimization algorithm applied to space trajectory design. Proc. Inst. Mech. Eng. Part G: J. Aerosp. Eng. **225**(11), 1211–1227 (2011)
4. Zuiani, F., Vasile, M.: Improved individualistic actions for multi-agent collaborative search. In: EVOLVE (2013)
5. Zuiani, F., Vasile, M.: Multi agent collaborative search based on tchebycheff decomposition. Comput. Optim. Appl. **56**(1), 189–208 (2013)
6. Zuiani, F, Vasile, M.: Multi agent collaborative search with thecycheff decomposition and monotonic basin hopping. In: BIOMA, May 2012
7. Zhang, Q., Zhou, A., Zhao, S., Suganthan, P.N., Liu, W., Tiwari, S.: Multiobjective optimization test instances for the cec 2009 special session and competition. University of Essex, Colchester, UK and Nanyang technological University, Singapore, special session on performance assessment of multi-objective optimization algorithms, technical report, pp. 1–30 (2008)

8. Zhang, Q., Liu, W., Li, H.: The performance of a new version of moea/d on cec09 unconstrained mop test instances. IEEE Congr. Evolut. Comput. **1**, 203–208 (2009)
9. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.A.M.T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
10. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. Evol. Comput. **8**(2), 173–195 (2000)
11. Schütze, O., Esquivel, X., Lara, A., Coello, C.A.C.: Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization. IEEE Trans. Evol. Comput. **16(**4), 504–522 (2012)

Comparison of Multi-objective Approaches to the Real-World Production Scheduling

Gregor Papa and Peter Korošec

Abstract The multi-objective optimization approach has a large influence in the industrial production scheduling. The goal of such optimization is to find a production schedule that satisfies different, usually contradictory, production and business constraints. In the paper, memetic versions of three multi-objective algorithms with different approaches to problem solving are implemented. The customized reproduction operators and local search procedures are also used. These memetic algorithms are applied to real order-lists from a production company. It is shown that the multi-objective approaches are able to find high-quality solutions, also when quick respond is required to adapt to dynamic business conditions. According to the results it is concluded that for the two tested real-world problems the IBEA confirmed its superiority over the NSGA-II and SPEA2.

1 Introduction

In the past we have already successfully approached a production-scheduling problem with a single-objective optimization [13]. The optimization goal was to find a production schedule that satisfies the production time constraints and minimizes the production costs. This involved many specific constraints that had to be considered. Later, the problem evolved, which brought some new constraints and new deciding criteria. Since the single-objective approach proved to be inefficient, we

G. Papa (✉) · P. Korošec
Computer Systems Department, Jožef Stefan Institute, Ljubljana, Slovenia
e-mail: gregor.papa@ijs.si

P. Korošec
e-mail: peter.korosec@ijs.si

G. Papa
Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

P. Korošec
Faculty of Mathematics, Natural Sciences and Information Technologies,
University of Primorska, Koper, Slovenia

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_27

had to consider a multi-objective approach [9].

There was some initial investigation performed on the usage of multi-objective approaches. In the previous work [9] we used the Indicator-Based Evolutionary Algorithm (IBEA) [16], since it nicely upgrades on our initial work when solving the single-objective scheduling problem [13]. In current work we further improve the findings presented previously [9] with the comparison of the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [5], Strength Pareto Evolutionary Algorithm 2 (SPEA2) [17], and IBEA. Following the IBEA's proven performance for more than three objectives [14], and the findings that for four contradictory objectives many classic multi-objective approaches are inappropriate [7], we decided to check the performance of those three multi-objective algorithms with the real-world production problem.

2 Related Work

The growing complexity of the real-world scheduling problems forced significant work to be devoted to the automation of scheduling and planning processes. Here, we often have to deal with very large search spaces, real-time performance demands, and dynamic environments [11]. Effective production scheduling solutions can result in reduction of personnel and production costs by minimizing machine idle time and increasing the number of on-time job deliveries [2].

The multi-objective optimization [4] is very common within the world of engineering problems. As this approach deals with multiple objectives it is also recognized in solving of planning and scheduling problems.

The Memetic Algorithms (MAs) were developed to obtain even better results than the Genetic Algorithm (GA) for various scheduling applications, and with the use of local search techniques the results were further improved. This hybrid approach not only improves the quality of the solutions, but it also reduces the overall computational time [8].

In our initial work [13] a guided local search algorithm was tested on real-world test cases of a production-scheduling problem. Such a problem is a member of the family of job shop scheduling problems, which are known to be NP-hard. Due to the problem's complexity (many constraints) we developed and used specialized local searches. They were guided with the genetic algorithm, parameter-less evolutionary search [12], and random selection. It was shown that the use of stochastic approaches greatly improved the quality of the production schedules with respect to the expert's manual solution. Furthermore, the evolutionary approach proved to be notably superior to the random search approach. On the other hand, the random-guided, local search approach was able to come impressively close to the results of the evolutionary approaches. It was obvious that its success was due to the quality of the local searches. Namely, to get good results in a relatively short time, a very powerful set of local searches had to be implemented. This led to good performances for all the guided approaches; the genetic algorithm being the most stable while producing the best results.

In the previous work [9] we have shown that the use of the memetic, multi-objective approach, based on the IBEA, does not reduce the quality of any objective with regard to the lexicographic evaluation of a single-objective approach, when used on the same production-scheduling problem. The only major downside of such an approach is in the increased time that is needed for a good Pareto front of solutions to be constructed. While in [13] we proved the suitability of the evolutionary approach to finding an optimum solution within a broad range of possible solutions, in [9] we presented some additional local search procedures, as well as we introduced the multi-objective approach, where the IBEA algorithm was used.

3 Implemented Multi-objective Algorithms

Based on the evolved production-schedule requirements we implemented and tested three memetic implementations based on different multi-objective algorithms: the NSGA-II, SPEA2, and IBEA. We adapted the basic implementation of these algorithms with our implementations of crossover and mutation operators in order to fully adapt to the specific problem of production scheduling.

3.1 Non-dominated Sorting Genetic Algorithm-II

The NSGA-II [5] is the second version of the Non-dominated Sorting Genetic Algorithm for solving non-convex and non-smooth single and multi-objective optimization problems. Its main features are: A non-dominated sorting procedure where all individuals are sorted according to the level of non-domination; It implements elitism which stores all non-dominated solutions, and enhances convergence properties; It adapts a suitable automatic mechanics based on the crowding distance in order to guarantee diversity and spread of solutions; Constraints are implemented using a modified definition of dominance without the use of penalty functions.

The NSGA-II orders the population into a hierarchy of non-dominated Pareto fronts. It calculates the crowding distance between members of each front on the front itself. The crossover and mutation are performed as classical operators of the GA. The members of the population are discriminated according to the rank of the front and distance within the front.

3.2 Strength Pareto Evolutionary Algorithm 2

The SPEA2 [17] is one of the multi-objective evolutionary algorithms that use elitism approach. Each individual is assigned a raw fitness calculated on the basis of the strength value of solutions who dominate it. To discriminate between individuals having identical raw fitness values additional density information is calculated.

The SPEA2 calculates the raw fitness as the sum of the strength values of the solutions that dominate a given candidate, where strength is the number of solutions that a given solution dominates. The density of an area of the Pareto front is estimated upon the Euclidean distance of the objective values between a given solution and the nearest neighbors of the solution. It iteratively fills the archive population with the candidate solutions in order of their fitness. The most similar solutions are truncated from the archive population. For selection of parents some classical GA selection method, such as binary tournament selection or random selection, is used. The crossover and mutation are performed as classical operators of the GA.

3.3 Indicator-Based Evolutionary Algorithm

The IBEA [16] is a multi-objective version of the GA, where the selection process is based on quality indicators. An indicator function assigns each pareto-set approximation a real value that reflects its quality, and the optimization goal is the identification of a pareto-set that minimizes an indicator function. Using the indicator concept no additional diversity-preservation mechanisms are required. It was demonstrated [16] that an indicator-based search can yield results that are superior to some other widely-used algorithms such as the improved SPEA2 and NSGA-II.

In a basic version of the IBEA, binary tournaments are used for the selection of individuals to undergo recombination. Next, it iteratively removes the worst individual from the population and updates the fitness values of the remaining individuals.

4 Production Scheduling Problem

The production scheduling problem was introduced in the company Eta Cerkno d.o.o., which produces components for domestic appliances [9, 13]. The most demanding production stage is the production of cooking hot plates. The fabrication process for various components used in different types of plates is similar, however due to clients' different demands the models differ in size (i.e., height, diameter), connector type, and power characteristics (i.e., wattage). For their logistic reasons the clients group different models of plates within the same order, implying the same due-dates for different products. Therefore, the production of these order groups must be scheduled very carefully to fulfil all the demands (i.e., quantities and due-dates), to maintain the specified amounts of different models in stock, to optimally occupy their workers, and to make efficient use of all the production lines. Although the assignment of due-dates is usually performed separately, and before the production scheduling, there are strong interactions between the two tasks. Each order placed by the customer somehow defines a batch of jobs, and their completion times should be as close as possible in order to reduce the waiting time and cost [15]. Furthermore, not all the production lines are equal, since each of them can produce only a few different models. A detailed formulation of the production-scheduling problem is presented in our initial work [13].

4.1 Production Schedule Encoding

The production schedule is encoded into a chromosome with tuples of values. Each tuple (gene) consists of the index of the enumerated order and the assigned production line. A chromosome with production schedule of n orders, is presented in Eq. 1.

$$C = g_{1o}g_{1l} \quad g_{2o}g_{2l} \quad \cdots g_{ko}g_{kl} \quad \cdots \quad g_{no}g_{nl}, \tag{1}$$

where n is the number of product orders, g_{ko} is an index of order $o_k \in O$ and g_{kl} is the production line used to produce the order o_k, for every $k \in \{1, 2, \ldots, n\}$.

4.2 Population Initialization

All input orders that have to be processed are firstly sorted within the initial order list, according to their due-dates. Next, different chromosomes are constructed as variations of the initial list, where each variation of the indexes of orders is encoded as a separate chromosome. In each chromosome the orders are randomly distributed, and the assigned production line is chosen randomly from among the possible lines for each order. The created initial population P consists of N chromosomes.

As the numbers that are encoded in the chromosome represent the indexes of orders, their values cannot be duplicated and also all indexes must be included. Also the assigned values for the production line depend on the possible production lines for particular order. These conditions have to be considered during the initialization as well as during all the subsequent phases.

4.3 Reproduction Operators

An order-based crossover operator interchanges positions that store the ordered numbers within some range. It takes the random part of two parents, and with a probability

p_c swaps the genes of the parents in this part and orders the remaining genes in the first parent in accordance with its order in the second parent. In our implementation four types of order-based crossover operators are used: order (OX) [10], cycle (CX) [10], partially-mapped (PMX) [10] and PTL [3] crossover. They are switched every 10 generations of the optimization process.

During the mutation process each value of the chromosome mutates with a mutation probability p_m. Five different types of mutation, which are described with more details in [9], are applied: Changing of the production line; Switching of two genes in the chromosome; Shifting of a gene into some new position; Replacing similar products; Merging of similar products. The first mutation type influences the second part of the gene (i.e., g_{kl}); the second mutation type influences the whole gene ($g_{ko}g_{kl}$); the remaining three mutation types influence only the first part of the gene (i.e., g_{ko}).

To limit a possible disruptive effect of mutation during the later stages of the optimization and to speed up the convergence to the optimum solution in the final optimization stages, the crossover and mutation probabilities are decreased during the algorithm execution.

4.4 Fitness Evaluation

The solutions $p \in P$ of each generation are evaluated after the reproduction operators and local search procedures modify them. Each solution p defines its set of objective values n_{obj}, where objective values are defined as: the number of delayed orders (n_{orders}); the sum of delayed days of all the delayed orders (n_{days}); the required number of workers (n_{workers}); and the sum of the change-over downtime in minutes (t_{change}). The objective values are calculated by the objective functions f_k, $k \in \{1, ..., n_{\text{obj}}\}$.

4.5 Ending Condition

In general the algorithm is run until the user stops the optimization process. To mimic overnight running, as it is used in real setting to form new production schedules, we decided to limit the number of evaluations to 300 million.

5 Memetic Algorithms

There are several approaches for implementing local search procedures. In our case we merged the presented NSGA-II, SPEA2 and IBEA algorithms to guide the local search procedures. The basic algorithms are implemented with the use of appropriate Java classes of the jMetal framework [6]. Since we are dealing with a combinatorial problem, we implemented our problem-specific versions of the crossover and mutation operators. Next, we added the local search procedures to enhance the efficiency of the algorithm.

Algorithm 1 Generic multi-objective memetic algorithm

```
SetInitialPopulation($P$)
Evaluate($P$)
while not EndingCondition() do
    $P'$ = Selection($P$)
    Crossover($P'$, $p_c$)
    Mutation($P'$, $p_m$)
    Evaluate($P'$)
    LocalSearch($P'$)
    $P$ = PopulationManagement($P \cup P'$)
end while
```

As presented in Algorithm 1 the pseudocode of a generic multi-objective memetic algorithm with different base algorithms is very similar. The main difference is in the Evaluate() function, which implements various fitness calculations (like raw fitness, density information, quality indicator...), and in PopulationManagement() function, which implements various algorithm-specific procedures (like sorting, crowding distance, and truncation procedure).

6 Performance Evaluation

6.1 Experimental Environment

The experiments were performed on the computer platform that is based on an AMD Opteron ™ 2.2-GHz processor, with 16 GB of RAM, and the Microsoft® Windows® 8.1 operating system. The algorithms are implemented in Sun Java 1.7.

6.2 Test Cases

For the fair comparison with the results from [9] the algorithms were tested on the same two real order lists from the production company. Task 1 consisted of $n = 470$ orders for 189 different products and Task 2 consisted of $n = 393$ orders for 175 different products. The number of orders n represents the problem dimension. The number of available production lines is $m = 5$.

As a comparison also a single objective result is presented. It was obtained as described in [9], where we used a lexicographic evaluation—the number of delayed orders (n_{orders}) was set as the most important objective, followed by the required number of workers (n_{workers}), the sum of delayed days for all the delayed orders (n_{days}), and the sum of the change-over downtime in minutes (t_{change}).

6.3 Control Parameter Settings

The control parameters were based on the previous setting as used in [9], to achieve as equal as possible conditions for all compared algorithms:

- the population size $N = 500$;
- the crossover probability $p_{\text{c}} = 0.5$;
- the mutation probabilities $p_{\text{m}_{\text{change}}} = 0.01$, $p_{\text{m}_{\text{switch}}} = 0.01$, $p_{\text{m}_{\text{shift}}} = 0.01$, $p_{\text{m}_{\text{randomize}}} = 0.05$ and $p_{\text{m}_{\text{merge}}} = 0.5$;
- the number of evaluations was 300 million.

The implementation of the crossover and mutation was the same for all algorithms. The algorithms differ in their specific implementations of fitness calculation, selection procedures and in progress of the solutions into the next generations (i.e., how the offspring/archive population is managed).

6.4 Results

Results of comparison of all three algorithms for both tasks are presented in Figs. 1, 2, 3 and 4 as well as in Tables 1 and 2. In all the figures the X axis represents the n_{orders} objective, the Y axis represents the n_{workers} objective, the Z axis represents the t_{change} objective and the color scheme represents the n_{days} objective.

Similarly, as presented in [9], Figs. 1 and 2 show the pareto front for Task 1 in 4D space from different perspectives for all three compared algorithms.

In Fig. 1 we can see that for the Task 1 the “nicest” pareto front in regard to the XY plane is returned by the IBEA algorithm, while both the SPEA2 and NSGA-II produce more wide spreaded pareto front. Also the minimum acquired values are much lower at the IBEA algorithm, followed by the NSGA-II and lastly the SPEA2 algorithm.

In Fig. 2 we can see that for the Task 1 the pareto front in regard to the XZ an YZ planes is pronouncedly divided into two parts by the IBEA algorithm, while both the SPEA2 and NSGA-II produce more even pareto front with some dislocated solutions. The main reason for this is that the IBEA was able to generate “nicer” pareto front on XY plane, with solution with lower Y values required much higher X, Z, and colored values. This indicates that the n_{workers} objective invertly influences other objectives. For X values we see that IBEA was able to find lower solutions, while for all other objectives the quality of solution is much closer.

In Table 1 we can see the performance of all three multi-objective algorithms for Task 1, and also the solution obtained by the single objective approach is presented

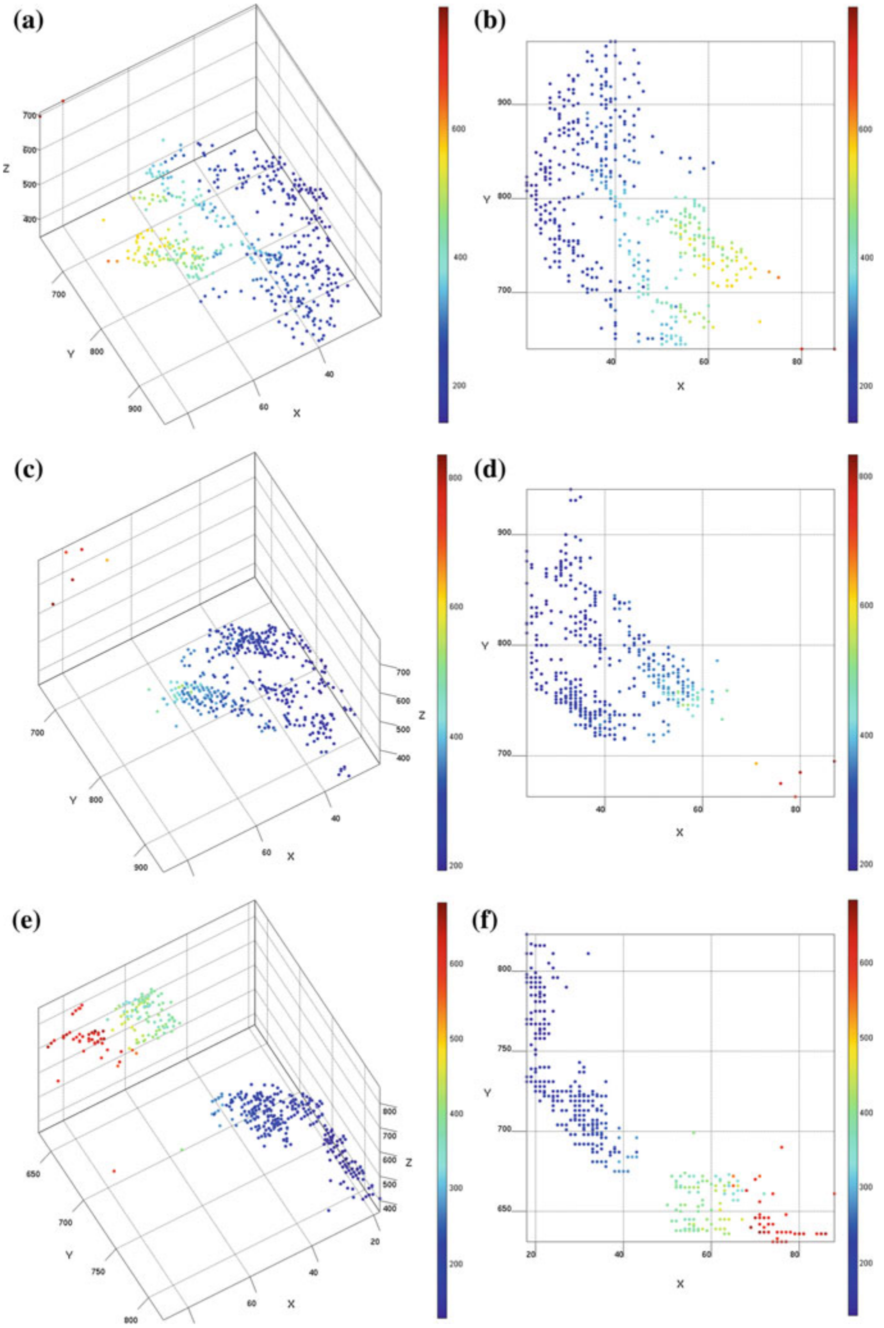


Fig. 1 Pareto front for Task 1 in the 4D space and XY plane: **a** NSGA-II 4D, **b** NSGA-II XY, **c** SPEA2 4D, **d** SPEA2 XY, **e** IBEA 4D, and **f** IBEA XY

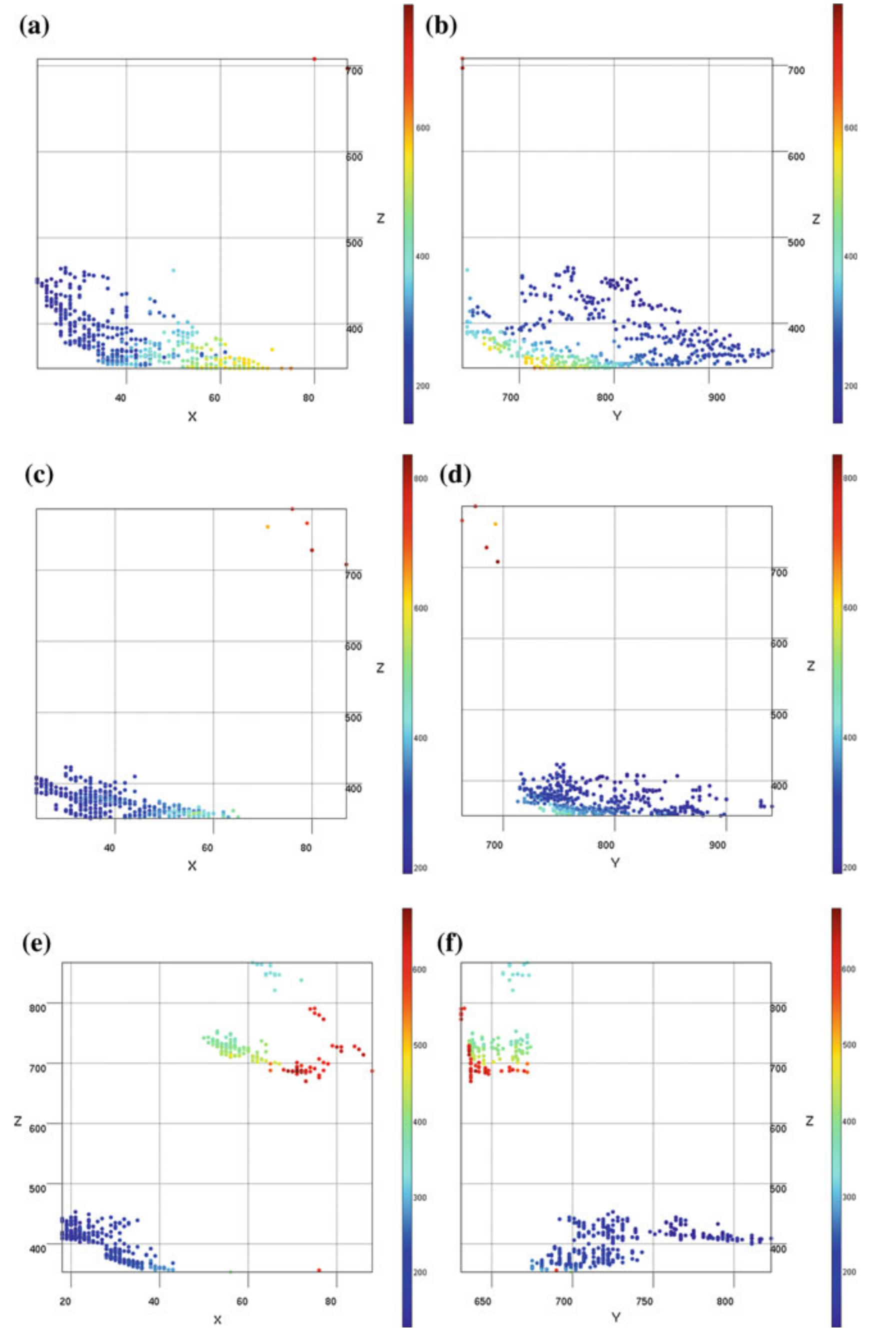


Fig. 2 Pareto front for Task 1 in the XZ and YZ plane: **a** NSGA-II XZ, **b** NSGA-II YZ, **c** SPEA2 XZ, **d** SPEA2 YZ, **e** IBEA XZ, and **f** IBEA YZ

Table 1 Results of optimization for task 1

Algorithm	Statistics	n_{orders}	n_{workers}	t_{change}	n_{days}
NSGA-II	pareto min	21	640	348	141
	Pareto max	87	967	708	790
	Pareto median	40	778	370	281
SPEA2	Pareto min	24	663	351	191
	Pareto max	87	941	786	836
	Pareto median	38	769	370	279
IBEA	Pareto min	18	631	353	127
	Pareto max	88	823	867	681
	Pareto median	32	704	414	206
Single-objective		18	767	714	156

Table 2 Results of optimization for Task 2

Algorithm	Statistics	n_{orders}	n_{workers}	t_{change}	n_{days}
NSGA-II	Pareto min	17	567	336	63
	Pareto max	56	902	655	621
	Pareto median	27	681	369	186
SPEA2	Pareto min	31	695	368	178
	Pareto max	82	999	451	1095
	Pareto median	39	779	385	284
IBEA	pareto min	16	538	355	59
	Pareto max	50	778	433	330
	Pareto median	26	601	371	101
Single-objective		15	702	443	155

as a comparison. The Table presents algorithms' pareto min, max, and median values for all objectives. The median value shows where the focus of search is. Since we are dealing with minimisation problem on all objectives, these are the values most interesting and indicating for us. When comparing the objective n_{orders} we can see that the range of values is more or less the same for all three algorithms, while the median value is the lowest at the IBEA. For the objective n_{workers} the IBEA has the smallest range and the lowest median value. For the objective t_{change} the IBEA has the largest range and a little bit higher median value than the other two algorithms. For the objective n_{days} the range of values is a little larger for the NSGA-II and SPEA2, while the median value of the IBEA is again the lowest one.

In Fig. 3 we can see that for the Task 2 the pareto fronts in regard to the XY plane are quite similar for the IBEA and NSGA-II , while SPEA2 produced much "weaker" pareto front (containing less solutions). Similarly to Task 1 the minimum acquired

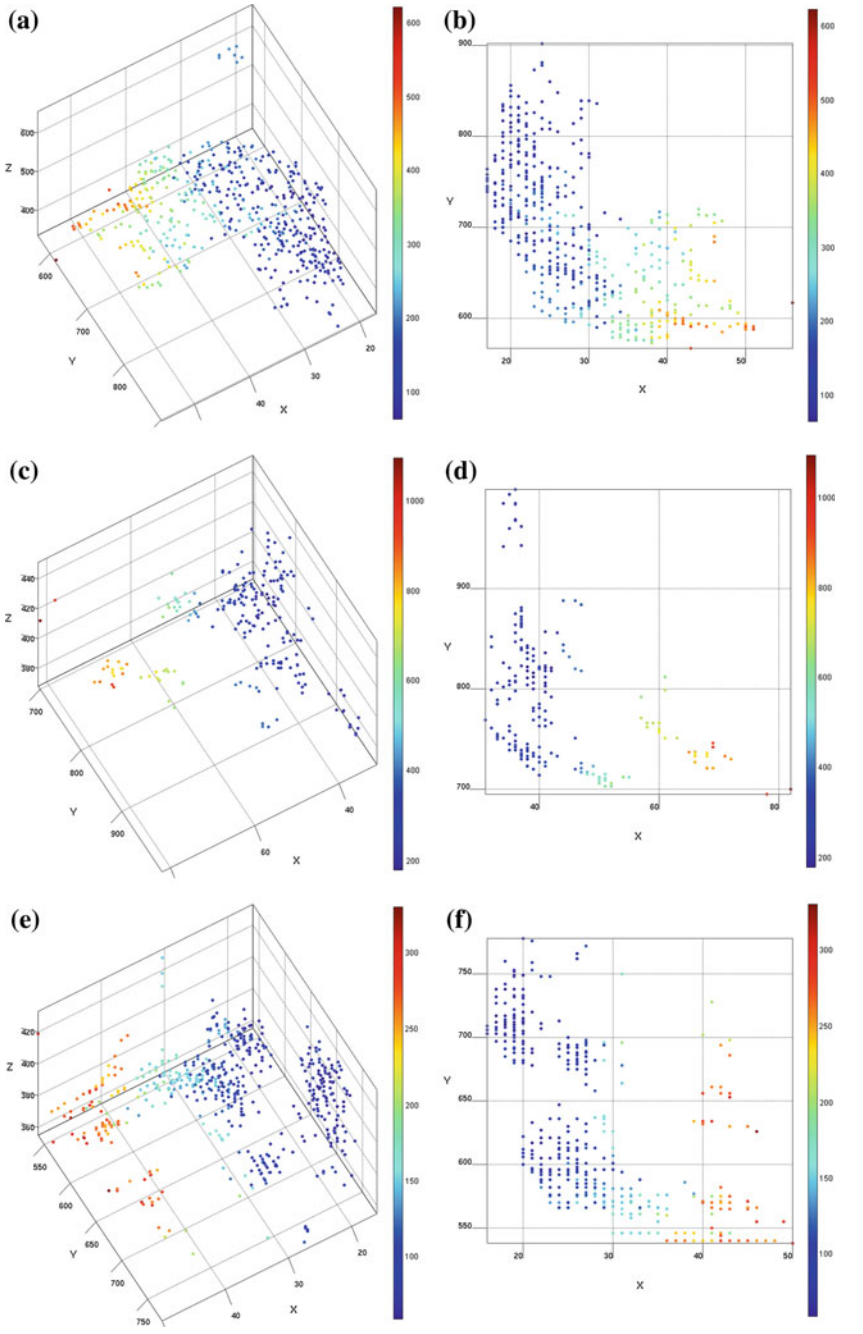


Fig. 3 Pareto front for Task 2 in the 4D space and XY plane: **a** NSGA-II 4D, **b** NSGA-II XY, **c** SPEA2 4D, **d** SPEA2 XY, **e** IBEA 4D, and **f** IBEA XY

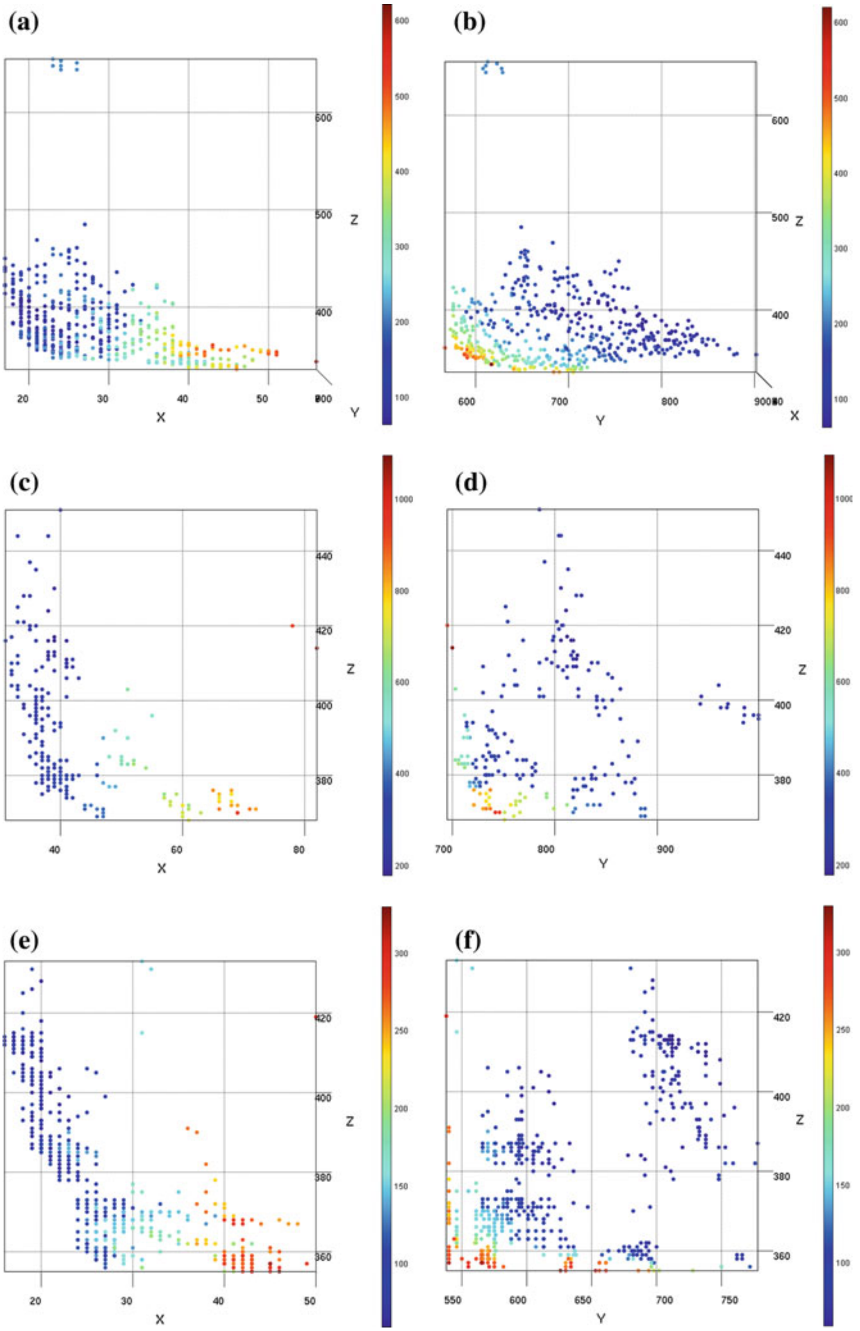


Fig. 4 Pareto front for Task 2 in the XZ and YZ plane: **a** NSGA-II XZ, **b** NSGA-II YZ, **c** SPEA2 XZ, **d** SPEA2 YZ, **e** IBEA XZ, and **f** IBEA YZ

values are much lower at the IBEA algorithm, followed by the NSGA-II and lastly the SPEA2 algorithm.

In Fig. 4 we can see that for the Task 2 the pareto front in regard to the XZ an YZ planes is not pronouncedly divided in two parts as noticed with Task 1. This indicates that the n_{workers} objective does not always "negatively" influences other objectives.

In Table 2 we can see the performance of all algorithms in Task 2, where their pareto min, max, and median values for all objectives are shown. When comparing the objective n_{orders} we can see that the range of values is more or less the same for the IBEA and NSGA-II, while the range is a bit larger for SPEA2; also the median values are lower for the IBEA and NSGA-II. For the objective n_{workers} the IBEA has a little bit smaller and lower range than the NSGA-II and SPEA2, and also the IBEA has the lowest median value. For the objective t_{change} the IBEA and SPEA2 have smaller range of values, but a little bit higher median value than the NSGA-II. For the objective n_{days} the range of values is the smallest and lowest for the IBEA, and also the median value of the IBEA is the lowest one. Similarly as with Task 1, the IBEA was able to find the lowest solution values for objectives except t_{change}, where the NSGA-II was able to find the lowest value. The median value, which shows where the focus of search is, follows the same pattern.

Considering all the information provided by the figures and tables one can conclude that the two tested tasks of presented real-world problem confirm the superiority of the IBEA over the NSGA-II and SPEA2, as shown by using benchmark functions in [16]. So, in this case test benchmark functions proved as a good indication, which is the most suitable algorithm for the job.

7 Conclusion

The multi-objective optimization approach has become important part in the industrial production scheduling, where its goal is to find a production schedule that satisfies different, usually contradictory, production and business constraints. We implemented memetic versions of the NSGA-II, SPEA2 and IBEA multi-objective algorithms, with different approaches to problem solving. These memetic algorithms were applied to real order-lists from a production company. We have shown that for the tested real-world problem the IBEA confirmed its superiority over the NSGA-II and SPEA2, as already indicated by the synthetic test benchmark functions.

Acknowledgements The authors acknowledge the financial support from the Slovenian Research Agency (research core funding No. P2-0098). This work is also part of a project SYNERGY that has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 692286. This work is also part of a project MANTIS that has received funding from the ECSEL Joint Undertaking under grant agreement No 662189. This Joint Undertaking receives support from the European Unions Horizon 2020 research and innovation programme and Spain, Finland, Denmark, Belgium, Netherlands, Portugal, Italy, Austria, United Kingdom, Hungary, Slovenia, Germany.

References

1. Bäck, T., Fogel, D., Michalewicz, Z.: Evolutionary Computation 1: Basic Algorithms and Operators, 2nd edn. Taylor & Francis Group, Heidelberg (2000)
2. Chan, F.T., Au, K., Chan, P.: A decision support system for production scheduling in an ion plating cell. Expert Syst. Appl. **30**(4), 727–738 (2006)
3. Czogalla, J., Fink, A.: On the effectiveness of particle swarm optimization and variable neighborhood descent for the continuous flow-shop scheduling problem. In: Xhafa, F., Abraham, A. (eds) Metaheuristics for Scheduling in Industrial and Manufacturing Applications, volume 128 of Studies in Computational Intelligence, pp. 61–89. Springer, Berlin, Heidelberg (2008)
4. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley-Interscience Series in Systems and Optimization. Wiley, Chichester (2001)
5. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J., Schwefel, H.-P. (eds.) Parallel Problem Solving from Nature PPSN VI, volume 1917 of Lecture Notes in Computer Science, pp. 849–858. Springer, Berlin. Heidelberg (2000)
6. Durillo, J.J., Nebro, A.J.: jmetal: a java framework for multi-objective optimization. Adv. Eng. Softw. **42**(10), 760–771 (2011)
7. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary many-objective optimization: a short review. IEEE Congr. Evol. Comput. CEC **2008**, 2424–2431 (2008)
8. Kamrul Hasan, S.M., Sarker, R., Essam, D., Cornforth, D.: Memetic algorithms for solving job-shop scheduling problems. Memet. Comput. **1**(1), 69–83 (2009)
9. Korošec, P., Bole, U., Papa, G.: A multi-objective approach to the application of real-world production scheduling. Expert Syst. Appl. **40**(15), 5839–5853 (2013)
10. Michalewicz, Z., Fogel, D.: How to Solve It: Modern Heuristics, 2nd edn. Springer, Berlin, Heidelberg (2004)
11. Nareyek, A. (ed.): Local Search for Planning and Scheduling, ECAI 2000, Berlin, Germany, volume 2148 of Lecture Notes in Computer Science. Springer (2000)
12. Papa, G.: Parameter-less algorithm for evolutionary-based optimization. Comput. Optim. Appl. **56**(1), 209–229 (2013)
13. Papa, G., Vukašinović, V., Korošec, P.: Guided restarting local search for production planning. Eng. Appl. Artif. Intell. **25**(2), 242–253 (2012)
14. Wagner, T., Beume, N., Naujoks, B.: Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) Evolutionary Multi-Criterion Optimization, volume 4403 of Lecture Notes in Computer Science, pp. 742–756. Springer, Berlin, Heidelberg (2007)
15. Zhang, R., Wu, C.: A hybrid local search algorithm for scheduling real-world job shops with batch-wise pending due dates. Eng. Appl. Artif. Intell. **25**(2), 209–221 (2012)
16. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII), pp. 832–842. Springer (2004)
17. Zitzler, E., Laumanns, M., and Thiele, L.: SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In: Giannakoglou, K.C., Tsahalis, D.T., Périaux, J., Papailiou, K.D., Fogarty, T. (eds.) Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, pp. 95–100 (2001)
18. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, Grunert: V.: performance assessment of multiobjective optimizers: an analysis and review. IEEE Trans. Evol. Comput. **7**(2), 117–132 (2003)

Elucidation of Influence of Fuels on Hybrid Rocket Using Visualization of Design-Space Structure

Kazuhisa Chiba, Shin'ya Watanabe, Masahiro Kanazaki, Koki Kitagawa and Toru Shimada

Abstract The stratum-type association analysis as a new data mining technique has been applied to the conceptual design of a single-stage launch vehicle with hybrid rocket engine. The conceptual design was performed by using design informatics, which has three points of view, i.e., problem definition, optimization, and data mining. The primary objective of the present design is that the down range and the duration time in the lower thermosphere are sufficiently secured for the aurora scientific observation, whereas the initial gross weight is held down to the extent possible. The multidisciplinary design optimization was performed by using a hybrid evolutionary computation. Data mining was also implemented by using the stratum-type association analysis. Consequently, the design information regarding the tradeoffs has been revealed. The hierarchical dendrogram generated by using the stratum-type

K. Chiba (✉)
The University of Electro -Communications, 1-5-1, Chofugaoka,
Tokyo, Chofu 182-8585, Japan
e-mail: kazchiba@uec.ac.jp

S. Watanabe
Muroran Institute of Technology, 27-1, Mizumotocho, Hokkaido,
Muroran 050-8585, Japan
e-mail: sin@csse.muroran-it.ac.jp

M. Kanazaki
Tokyo Metropolitan University, 6-6, Asahigaoka, Hino, Tokyo 191-0065, Japan
e-mail: kana@tmu.ac.jp

K. Kitagawa · T. Shimada
Japan Aerospace Exploration Agency, 3-1-1, Yoshinodai, Chuo,
Sagamihara 252-5210, Japan
e-mail: kitagawa.koki@jaxa.jp

T. Shimada
e-mail: shimada.toru@jaxa.jp

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_28

association analysis indicates the structure of the design space in order to improve the objective functions. Furthermore, the assignments of the stratum-type association analysis have been obtained.

1 Introduction

Design informatics is essential for practical design problems. Although solving design optimization problems is important under the consideration of many disciplines of engineering [1], the most significant part of the process is the extraction of useful knowledge of the design space from results of optimization runs. The results produced by multiobjective optimization (MOO) are not an individual optimal solution but rather an entire set of optimal solutions due to tradeoffs. That is, the result of an MOO is not sufficient from the practical point of view as designers need a conclusive shape and not the entire selection of possible optimal shapes. On the other hand, this set of optimal solutions produced by an evolutionary MOO algorithm can be considered a hypothetical design database for design space. Then, data mining techniques can be applied to this hypothetical database in order to acquire not only useful design knowledge but also the structurization and visualization of design space for the conception support of basic design. This approach was suggested as design informatics [6]. The goal of this approach is the conception support for designers in order to materialize innovation. This methodology is constructed by the three essences as (1) problem definition, (2) efficient optimization, and (3) structurization and visualization of design space by data mining. A design problem including objective function, design variable, and constraint, is strictly defined in view of the background physics for several months (problem definition is the most important process for all designers because it directly gives effect on the quality of design space. Since the garrulous objective-function/design-variable space including physics and design information which is not inherently necessary to consider should be performed unnecessary evolutionary exploration and mining, it is conceived to be low-quality design space), then optimization is implemented in order to acquire nondominated solutions (quasi-Pareto solutions) as hypothetical database. Data mining is performed for this database in order to obtain design information. Mining has the role of a postprocess for optimization. Mining result is the significant observations for next design phase and also becomes the material to redefine a design problem.

In the present study, a single-stage launch vehicle with hybrid rocket engine using solid fuel and liquid oxidizer for the scientific observation of aurora will be conceptually designed by using design informatics approach. The objective of this study is to apply the stratum-type association analysis as a new data-mining approach to the optimization results [5] so that the design knowledge is accumulated for the development of actual vehicle.

2 Design Informatics

2.1 Optimization Method

Design informatics after the definition of detailed problem is constructed by two phases as optimization and data mining. Evolutionary computation is used for optimization. Although a surrogate model [14] like as the Kriging model [12], which is a response surface model developed in the field of spatial statistics and geostatistics, can be employed as optimization method, it will not be selected because it is difficult to deal with a large number of design variables. In addition, since the designers require to present many exact optimum solutions for the decision of a compromise one, an evolutionary-based Pareto approach as an efficient multi-thread algorithm, which the plural individuals are parallel conducted, is employed instead of gradient-based methods. The optimizer used in the present study is the hybrid evolutionary method between the differential evolution (DE) and the genetic algorithm (GA) [4]. Moreover, global design information is primarily essential in order to determine a compromise solution. The view of hybridization is inspired by the evolutionary developmental biology [2]. When there is the evolution which the Darwinism cannot explain in the identical species, each individual might have a different evolutionary methodology. When the practical evolution is imitated for the evolutionary computation, the different evolutionary algorithms might ultimately be applied to each individual in population. The making performance of next generation for each methodology depends on not only their algorithms but also the quality of candidate of parent in the archive of nondominated solutions. The present hybridization is intended to improve the quality of candidate of parent by sharing the nondominated solutions in the archive among each methodology. In the present study, the evolutionary hybrid optimization methodology between DE and GA is employed. It was confirmed that this methodology had the high performance regarding the convergence and diversity, as well as the strength for noise [4]. Note that noise imitates the error on computational analyses and experiments and is described as the perturbation on objective functions. It is an important factor when the optimization for practical engineering problem is considered.

First, multiple individuals are generated randomly as an initial population. Then, objective functions are evaluated for each individual. The population size is equally divided into sub-populations between DE and GA (although sub-population size can be changed at every generations on the optimizer, the determined initial sub-populations are fixed at all generations in the present study). New individuals generated by each operation are combined in next generation. The nondominated solutions in the combined population are archived in common. It is notable that only the archive data is in common between DE and GA. The respective optimization methods are independently performed in the present hybrid methodology.

The present optimization methodology is a real-coded optimizer [16]. Although GA is based on the real-coded NSGA-II (the elitist nondominated sorting genetic algorithm) [8], it is made several improvements in order to be progressed with the diversity of solutions. Fonseca's Pareto ranking [9] and the crowding distance [8] are used for the fitness value of each individual. The stochastic universal sampling [3] is employed for parents selection. The crossover rate is 100%. The principal component analysis blended crossover-α (PCABLX) [20] and the confidence interval based crossover using L_2 norm (CIX) [11] are used because of the high performance for the convergence and the diversity as well as the strength for noise [4]. The subpopulation size served by GA is equally divided for these two crossovers. The mutation rate is set to be constant as the reciprocal of the number of design variables. For alternation of generations, the Best-N selection [8] is used. DE is used as the revised scheme [17] for multiobjective optimization from DE/rand/1/bin scheme. The scaling factor F is set to be 0.5. The present optimizer has the function of range adaptation [18], which changes the search region according to the statistics of better solutions, for all design variables. In the present study, the range adaptation is implemented at every 20th generations.

2.2 Data-Mining Technique

The new data mining technique named as the stratum-type association analysis [21] has been applied to analyzing nondominated solutions. The previous study [7] employed the rough set theory in order to obtain the concrete rule regarding the design principles of design variables. Since the rough set theory gives individual rules based on the machine learning, it does not reveal the correlation knowledge among them. The present methodology systematizes individual rules and structurize design space using a hierarchical dendrogram so that methodology obtains a bird's-eye view of it in order to have useful knowledge. The feature of the present methodology is a recursive clustering using association rules and a multi-granular analysis. The results of the recursive clustering can be visualized as a hierarchical dendrogram in which each node is a sub cluster of nondominated solutions. The present system is expected to extract design information from microscopic to macroscopic view points due to the structurization of design space.

First, the present system discretizes continuous data for logical analysis. Second, association rules are derived from discretized data through logical analysis. Since designer would like to primarily acquire the design knowledge regarding objective function because it corresponds to design requirement. The association rules regarding objective function will be extracted at antecedent and consequent processes. The association rules are gradually integrated into subsets regarding the degree of coincidence at antecedent and consequent processes. Finally, the present methodology constructs a structured hierarchical dendrogram by clustering subsets based on synthetic correlations. It is essential that the methodology generates the subsets of nondominated solutions using not the similarity among them but the inherent characteristics in each nondominated solution.

A hierarchical dendrogram can be simply generated through clustering the association rules with similarity and a multi-granular analysis. Therefore, the present methodology can be optionally selected regarding clustering manner. The methodology integrates the generated rules into subsets using not similarity but the accordance with the coincidence of the rules generated by the antecedent and consequent operations. A generated hierarchical dendrogram is simply constructed by the inclusive correlation among the subsets. Indeed, all elements which contain all extracted association rules are selected. All combinations of these elements are generated without overlap. Then, the subsets are generated in order to concentrate the elements. Thereupon, all of the subsets have common characteristics. That is, all nodes of a hierarchical dendrogram have one common characteristic at least. The characteristic in a node is useful knowledge for designers.

3 Problem Definition

Single-stage rockets have been researched and developed for the scientific observations and the experiments of high-altitude zero-gravity condition, whereas multi-stage rockets have been also studied for the orbit injection of payload. The launch vehicle with hybrid rocket engine using solid fuel and liquid oxidizer has been researched and developed as an innovative technology in mainly Europe and United States [13, 19]. The present study will investigate the conceptual design in order to develop a next-generation single-stage launch vehicle with hybrid rocket engine. A hybrid rocket offers the several advantages as higher safety, lower cost, and pollution free flight. The multi-time ignition is the especial ascendancy of hybrid rocket engine. On the other hand, the disadvantage of a hybrid rocket engine is in its combustion. As a hybrid rocket engine has low regression rate due to turbulent boundary layer combustion, the thrust of hybrid rocket engine is less than that of pure solid and pure liquid engines which can obtain premixed combustion. In addition, as the mixture ratio between solid fuel and liquid oxidizer is temporally fluctuated, thrust changes with time.

The conceptual design for a single-stage hybrid rocket [15], simply composed of a payload chamber, an oxidizer tank, a combustion chamber, and a nozzle, is considered in the present study shown in Fig. 1. A single-stage hybrid rocket for

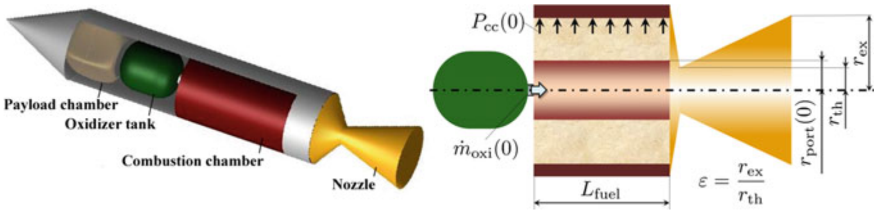


Fig. 1 Conceptual illustrations of hybrid rocket and its design variables regarding the geometry. Aperture ratio of nozzle ϵ is described by using the radius at nozzle exit r_{ex} and the radius at nozzle throat r_{th}

aurora scientific observation will be focused because the rocket for more efficient scientific observation is desired for successfully obtaining new scientific knowledge on the aurora observation by the Institute of Space and Astronautical Science (ISAS), Japan Aerospace Exploration Agency (JAXA) in 2009. In addition, a single-stage hybrid rocket problem fits for the resolution of the fundamental physics regarding hybrid rocket engine.

3.1 Objective Functions

Three objective functions are defined in the present study. First objective is the maximization of the down range in the lower thermosphere (altitude of 90–150 km) R_d (km) (obj1). Second is the maximization of the duration time in the lower thermosphere T_d (s) (obj2). It recently turns out that atmosphere has furious and intricate motion in the lower thermosphere due to the energy injection, which leads aurora, from high altitude. The view of these objective functions are to secure the horizontal distance and time for the competent observation of atmospheric temperature and the wind for the elucidation of atmospheric dynamics and the balance of thermal energy. Third objective is the minimization of the initial gross weight of launch vehicle $M_{\mathrm{tot}}(0)$ (kg) (obj3), which is generally the primary proposition for space transportation system.

3.2 Design Variables

Seven design variables are used as initial mass flow of oxidizer $\dot{m}_{\mathrm{oxi}}(0)$ (kg/s) (dv1), fuel length L_{fuel} (m) (dv2), initial radius of port $r_{\mathrm{port}}(0)$ (m) (dv3), combustion time t_{burn} (s) (dv4), initial pressure in combustion chamber $P_{\mathrm{cc}}(0)$ (MPa) (dv5), aperture ratio of nozzle ϵ [-] (dv6), and elevation at launch time ϕ (deg) (dv7). Note that there is no constraint except the limitations of upper/lower values of each design variable summarized in Table 1. These upper/lower values are exhaustively covering the region of design space which is physically admitted. When there is a sweet spot (the region that all objective functions proceed optimum directions) in the objective-function space, the exploration space would intentionally become narrow due to the operation of range adaptation on the evolutionary computation.

3.3 Evaluation Method

First of all, the mixture ratio between liquid oxidizer and solid fuel $O/F(t)$ is computed by the ratio between the mass flow of oxidizer $\dot{m}_{\mathrm{oxi}}(t)$ and that of fuel $\dot{m}_{\mathrm{fuel}}(t)$.

Table 1 Limitation of upper/lower values of each design variable

Serial number	Design variable	Design space		
dv1	Initial mass flow of oxidizer	1.0≤	$\dot{m}_{\text{oxi}}(0)$ (kg/s)	≤30.0
dv2	Fuel length	1.0≤	L_{fuel} (m)	≤10.0
dv3	Initial radius of port	0.01≤	$r_{\text{port}}(0)$ (m)	≤0.30
dv4	Combustion time	10.0≤	t_{burn} (s)	≤ 40.0
dv5	Initial pressure in combustion chamber	3.0≤	$P_{\text{cc}}(0)$ (MPa)	≤6.0
dv6	Aperture ratio of nozzle	5.0≤	ε [-]	≤8.0
dv7	Elevation at launch time	50.0≤	$\phi(0)(°)$	≤90.0

$$
\begin{aligned}
O/F(t) &= \frac{\dot{m}_{\text{oxi}}(t)}{\dot{m}_{\text{fuel}}(t)}. \\
\dot{m}_{\text{fuel}}(t) &= 2\pi r_{\text{port}}(t) L_{\text{fuel}} \rho_{\text{fuel}} \bar{\dot{r}}_{\text{port}}(t), \\
r_{\text{port}}(t) &= r_{\text{port}}(0) + \int \dot{r}_{\text{port}}(t) dt.
\end{aligned}
\tag{1}
$$

$\dot{m}_{\text{oxi}}(t)$ and $\dot{m}_{\text{fuel}}(t)$ are the mass flow of oxidizer (kg/s) and the mass flow of fuel (kg/s) at time t, respectively. $r_{\text{port}}(t)$ is the radius of port (m) at t, L_{fuel} describes fuel length, and ρ_{fuel} is the density of fuel (kg/m^3). $\dot{r}_{\text{port}}(t)$ describes the regression rate. After that, an analysis of chemical equilibrium is performed by using NASA-CEA (chemical equilibrium with applications) [10], then trajectory, thrust, aerodynamic, and structural analyses are respectively implemented. The present rocket is assumed as a point mass.

A combustion chamber is filled with solid fuel with a single port at the center to supply oxidizer. As the regression rate to the radial direction of the fuel $\dot{r}_{\text{port}}(t)$ (m/s) generally governs the thrust power of hybrid rocket engine, it is a significant parameter. The following experimental model [22] is used in the present study.

$$
\begin{aligned}
\dot{r}_{\text{port}}(t) &= a_{\text{fuel}} \times G_{\text{oxi}}^{n_{\text{fuel}}}(t) \\
&= a_{\text{fuel}} \times \left(\frac{\dot{m}_{\text{oxi}}(t)}{\pi r_{\text{port}}^2(t)} \right)^{n_{\text{fuel}}},
\end{aligned}
\tag{2}
$$

where, $G_{\text{oxi}}(t)$ is oxidizer mass flux (kg/m^2/s). a_{fuel} (m/s) and n_{fuel} [-] are the constant values experimentally determined by fuels. In the present study, liquid oxygen as liquid oxidizer and five solid fuels are used as thermoplastic resin polypropylene (PP), two WAX-type fuels (FT0070 and PW120), and two compounds between glycidyl

Table 2 Characteristic values of the fuels

Fuels	a_{fuel} (mm/s)	n_{fuel} [-]	Density (kg/m^3)
Polypropylene	0.0826	0.5500	910.0
WAX FT0070	0.1561	0.3905	926.6
WAX PW120	0.1677	0.4352	896.8
GAP50PEG50	0.3218	0.2937	1180.0
GAP60PEG40	0.4641	0.2864	1196.0

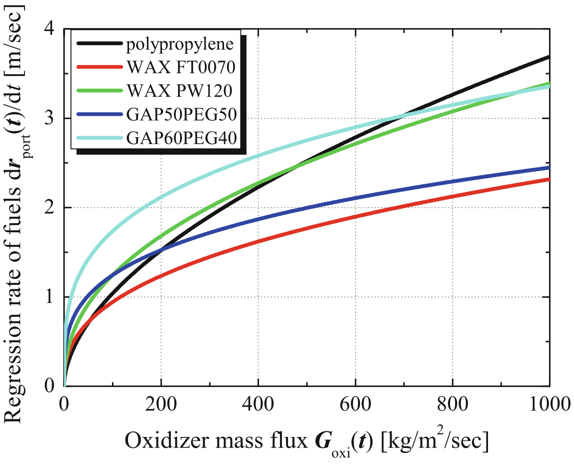


Fig. 2 Comparison of the regression rate $\dot{r}_{\mathrm{port}}(t)$ for the oxidizer mass flux $G_{\mathrm{oxi}}(t)$ per unit volume among the fuels

azide polymer and polyethylene glycol (GAP50PEG50 and GAP60PEG40, the number means the blend proportion) for solid fuel in order to compare the implications of fuels in the performance of hybrid rocket. Polypropylene has swirling flow for the supply mode of oxidizer and the other fuels have non-swirling flow. The characteristic values of the fuels are summarized in Table 2. The variation of the regression rate $\dot{r}_{\mathrm{port}}(t)$ for the oxidizer mass flux $G_{\mathrm{oxi}}(t)$ represented by Eq. (2) is shown in Fig. 2. The regression rate $\dot{r}_{\mathrm{port}}(t)$ of GAP60PEG40 is highest at low $G_{\mathrm{oxi}}(t)$, and $\dot{r}_{\mathrm{port}}(t)$ of WAX FT0070 and GAP50PEG50 is low on the whole.

4 Optimization Results

The present population size is set to be 18 and evolutionary computation is performed until 3,000 generations. The plots of acquired nondominated solutions are shown in Fig. 3. Figure 3a is the nondominated solutions in the three-dimensional objective-function space and Fig. 3b–d are projected onto two-dimensional surfaces in order to intuitively observe the tradeoffs. Figure 3 also reveals that there is no meaningful difference among the fuels for the tendency regarding the correlation among the

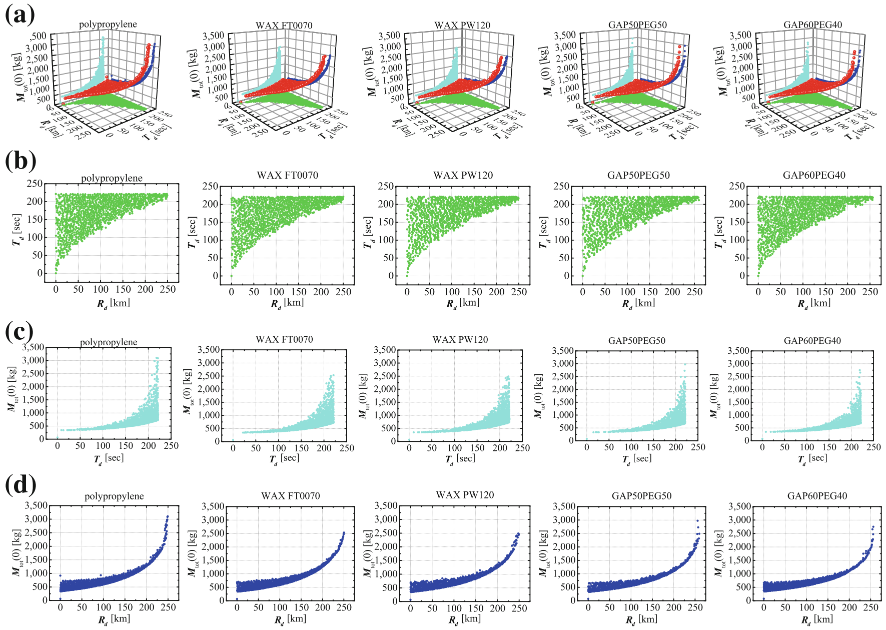


Fig. 3 Plots of nondominated solutions on the objective function space, **a** plots in three-dimensional objective function space, **b** two-dimensional plots between R_d (obj1) and T_d (obj2), **c** two-dimensional plots between T_d (obj2) and $M_{\text{tot}}(0)$ (obj3), and **d** two-dimensional plots between R_d (obj1) and $M_{\text{tot}}(0)$ (obj3). The column means the results for polypropylene, WAX FT0070, WAX PW120, GAP50PEG50, GAP60PEG40 from the left

objective functions. The considerations regarding the tradeoffs among the objective functions was performed [5]. Thereupon, the tradeoffs regarding polypropylene will be observed as a representation.

Figure 3a reveals that the connecting and convex nondominated surface except several isolated individuals is generated. There is no tradeoff between R_d and T_d in the lower thermosphere shown in Fig. 3b. This figure also shows that there are upper limitations of roughly 250 km for R_d and 220 s for T_d. Therefore, the projection plots onto two dimensions between R_d and T_d do not converge in one point. In this study, $\dot{m}_{\text{oxi}}(0)$ (dv1) has the limitation of upper/lower values. Since the regression rate to $\dot{r}_{\text{port}}(t)$ as an experimental model uses $\dot{m}_{\text{oxi}}(t)$, $\dot{r}_{\text{port}}(t)$ has constraint. As a result, the limitations are generated for R_d and T_d.

There is an incomplete tradeoff between T_d and $M_{\text{tot}}(0)$ shown in Fig. 3c. The convex nondominated surface to optimum direction with incompleteness is generated due to the limitation of T_d. As the inclination $\mathrm{d}M_{\text{tot}}(0)/\mathrm{d}T_d$ is small on the convex curve, T_d can be substantially improved when trifling $M_{\text{tot}}(0)$ would be sacrificed. In addition, Fig. 3c shows that the minimum initial gross weight to reach the limitation of the duration time (roughly 220 s) is approximately 700 kg. And also, the smallest initial gross weight to attain to the lower thermosphere (altitude of 90 km) is approximately 350 kg. As these values are better than those of the solid rockets which are operated at present for scientific observation, it suggests that hybrid rocket has an advantage even when hybrid rocket does not have a sequence of multi-time ignition.

There is a severe tradeoff between R_d and $M_{\mathrm{tot}}(0)$ shown in Fig. 3d (although the down range strictly has the upper limitation, it seems that the clean convex curve is generated because the limitation is on the edge of the nondominated surface). This figure shows that the maximum down range is roughly 130 km when the minimum initial gross weight to reach the limitation of T_d (approximately 700 kg) is adopted. $M_{\mathrm{tot}}(0)$ should be absolutely increased in order to have more R_d (greater than 130 km) despite no increase of T_d (remaining roughly 220 s). This fact suggests that the design strategies for the maximizations of R_d and T_d are different.

5 Data-Mining Results

A discretization is necessary for rule generation. The hybrid discretization manner between an equivalent distance and an equivalent frequency methods [21] is utilized. The number of discretization is set to be 10 for the objective functions and the design variables. The discretized regions are in common among the results for the five fuels.

The present algorithm has two significant parameters which should be artificially regulated. Those are the minimum confidence and the minimum support. The minimum confidence should be high value in order to retain the confidence of generated rules. It is set to be 90% in this study. On the other hand, value of minimum support dominates the number of nodes (which denotes the box in Figs. 4, 5, 6, 7 and 8). Present minimum support is artificially regulated from 7 to 10% in order to reduce the appropriate number of elements (which denotes each association rule) because many nodes with the small influence are generated.

The hierarchical dendrograms which the first stratum has the association rule regarding the objective functions are constructed for the minimization of all of the three objective functions. Note that there are severe tradeoffs between the initial gross weight and the other two objective functions, which were already revealed in the optimization results shown in Fig. 3. Moreover, the initial gross weight is the minimization function while the other two objective functions are maximization functions. Thereupon, all dendrograms for five fuels are shown regarding the minimization of the initial gross weight. The decoding manner of each node in the present hierarchical dendrograms is explained in the caption of Fig. 4.

Result for polypropylene: The first stratum has 456 nondominated solutions (because the total number of 999 is obtained for nondominated solutions by the optimization process, the proportion of application is roughly 45.6%) which have the attribute with the minimum-value node of $M_{\mathrm{tot}}(0)$. This result indicates that roughly half number of the nondominated solutions has the small value of $M_{\mathrm{tot}}(0)$.

The second stratum has five nodes with two attributes. Since one attribute is “2 = 0”, i.e., the minimum-value node of $M_{\mathrm{tot}}(0)$, the node indicates the tradeoff

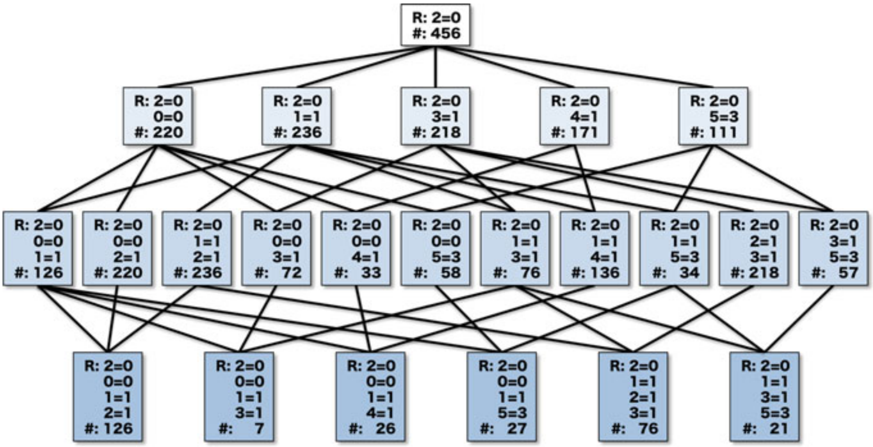


Fig. 4 Hierarchical dendrogram for polypropylene generated by the stratum-type association analysis for the minimization of the initial gross weight $M_{\mathrm{tot}}(0)$. The present dendrogram is constructed by 23 nodes. The two-line explanations show the rule (which is described as 'the variables' = 'its discretized region') from the top line using "R:" and the correspondent number of all nondominated solutions with the rule at the bottom using "#:". The variable numbers of 0–9 respectively denote obj1, obj2, obj3, dv1, dv2, ..., and dv7. Since all variables are discretized into 10 in the present study, the discretized region is described by using the number of 0–9. The top node is in the first stratum and the bottom nodes are in the fourth stratum

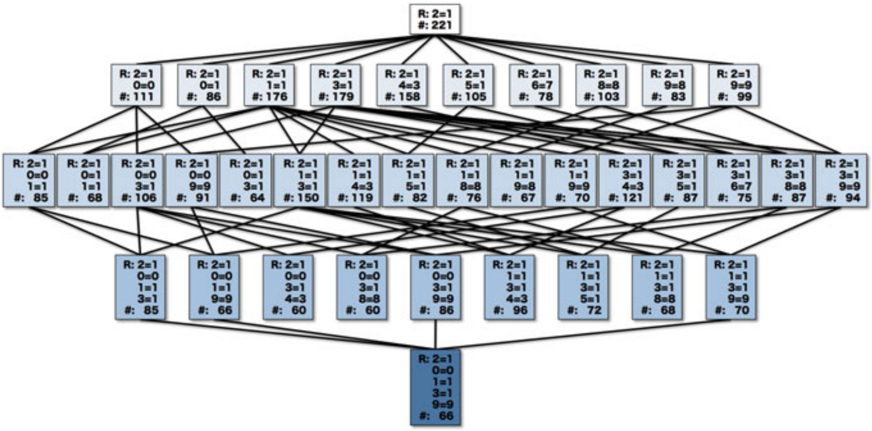


Fig. 5 Herarchical dendrogram for WAX FT0070

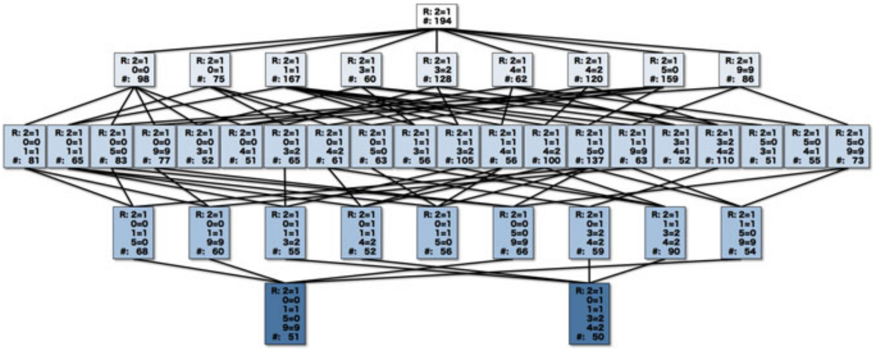


Fig. 6 Hierarchical dendrogram for WAX PW120

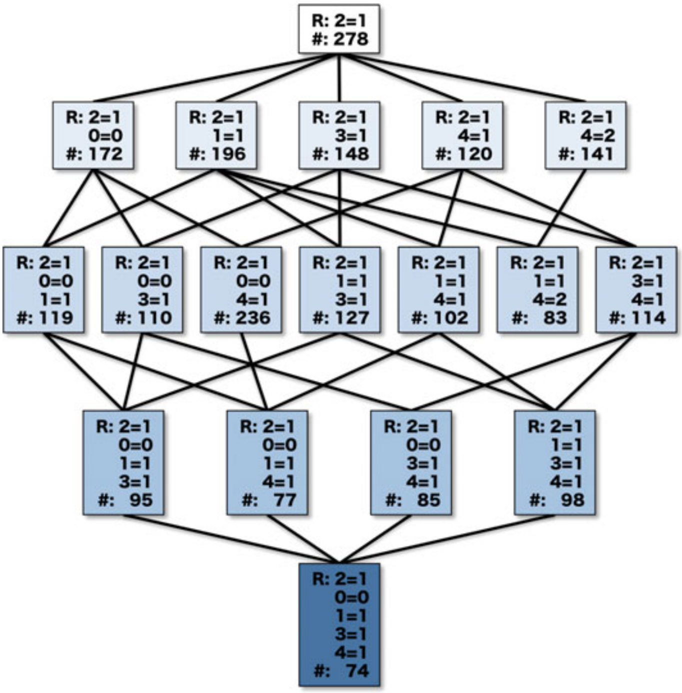


Fig. 7 Hierarchical dendrogram for GAP50PEG50

information when the other attribute is the rule regarding the other objective functions. In contrast, the node indicates the information regarding the effective design variable to minimize $M_{\mathrm{tot}}(0)$ when the other attribute is the rule regarding the design variables. The third node from the left has the rule for R_d with the lowest values (the number of nondominated solutions is 220). This node reveals the severe tradeoff because R_d for the 48% of the nondominated solutions is in the discretized region for minimum value when $M_{\mathrm{tot}}(0)$ is minimized. The second node from the left has the rule for

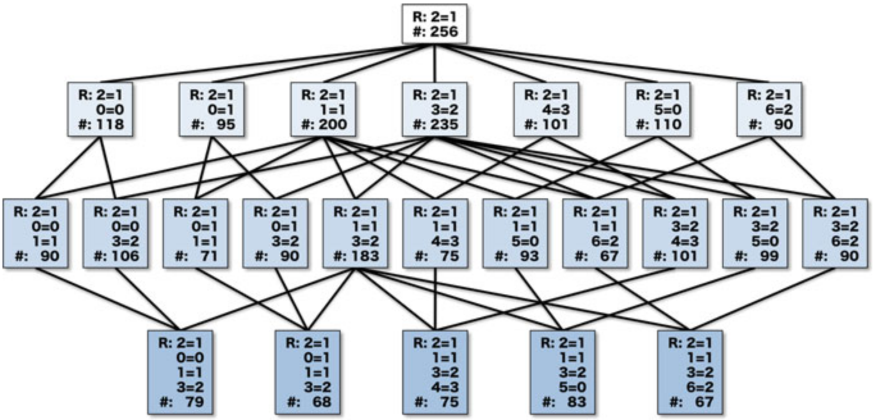


Fig. 8 Hierarchical dendrogram for GAP60PEG40

T_d with low values (the number of nondominated solutions is 236). This node also reveals the severe tradeoff because T_d for 52% of the nondominated solutions is in the discretized region for minimum value when $M_{\mathrm{tot}}(0)$ is minimized. Furthermore, the first node from the left has the rule for $\dot{m}_{\mathrm{oxi}}(0)$ (dv1) with low values (the number of nondominated solutions is 218). The fourth node from the left is the rule for L_{fuel} (dv2) with low values (the number of nondominated solutions is 171). These nodes reveal that dv1 and dv2 are essential to restrain $M_{\mathrm{tot}}(0)$. In addition, dv1 and dv2 are the significant design variables in order to also reduce the other two objective functions. The first node from the right is the rule for $r_{\mathrm{port}}(0)$ (dv3) with low values (the number of nondominated solutions is 11 as a small number). $r_{\mathrm{port}}(0)$ (dv3) indirectly gives an effect because small $r_{\mathrm{port}}(0)$ (dv3) gives an effect on restraining the mass flow of oxidizer $\dot{m}_{\mathrm{oxi}}(t)$ although $r_{\mathrm{port}}(0)$ (dv3) does not have a direct effect.

The third stratum has 11 nodes with three attributes. Seven nodes of those consist of two rules regarding the objective function and one rule regarding the design variable. Since the knowledge of the second stratum shows the effective design variable for the objective functions, the observation of these seven nodes can be omitted due to the comprehension into the design knowledge from the second stratum. The crucial point in the third stratum is that merely one node exists which has one rule regarding the objective function and two rules regarding the design variables. This result reveals that the correlation between the objective functions is strong and one design variable to depend on each objective function can be narrowed.

The fourth stratum has six nodes with four attributes. Since the knowledge of the third stratum reveals that the present problem has severe tradeoffs, four nodes have three rules regarding the objective function and one rule regarding the design variable. The fourth node of the left merely has two rules regarding the objective functions and two rules regarding the design variables. However, since the indicated design variables are $\dot{m}_{\mathrm{oxi}}(0)$ (dv1) and $r_{\mathrm{port}}(0)$ (dv3), the knowledge from the second stratum can be similarly interpreted.

Although the stratum-type association analysis cannot instruct the optimum and pessimum directions, the severeness of tradeoff can be quantitatively shown by using the number of application. In addition, the magnitude of the influence of the design variables on the objective functions can be quantitatively elucidated. Moreover, since the nondominated solutions to apply to a node are generated, the stratum-type association analysis is also useful for the selection of specific nondominated solutions in order to perform the data mining for the significant local region in the objective-function space.

Result for WAX FT0070: Since the rules regarding the objective functions shown in Fig. 5 are similar to those in Fig. 4, the difference of the rules regarding the design variables will be observed hereafter.

Figure 5 shows that the design variables with influence are increased compared with the result of polypropylene. Although the difference of the minimum support value is one of the reasons, it is considerable that there is a physical reason. Since WAX FT0070 entirely has low regression rate, only dv1, dv2, and dv3 which are indicated in Fig. 4 are insufficient in order to ensure the necessary thrust.

The newly indicated design variables are dv4, dv6, and dv7. The region of the rule for dv4 as combustion time has “7” shown in the fourth node from the right on the second stratum. Since WAX FT0070 entirely has low regression rate, the combustion time becomes large even when the fuel volume is small. That is, the rule regarding dv4 is caused by the low regression rate of WAX FT0070. As thrust of launch vehicle should be increased in order to reach the altitude of 90 km (which is the lower limit of the target altitude in the lower thermosphere) under the condition of small fuel volume, the aperture ratio of nozzle will be large. Therefore, the rule for dv6 as the aperture ratio of the nozzle is generated. When dv7 as the elevation at launch time comes closer to the angle of 90°, the flight pass will be shortened. Therefore, as fuel volume will become small, the initial gross weight can below.

In addition, the difference of the design region for dv2 and dv3 between polypropylene and WAX FT0070 will be considered. The hierarchical dendrograms for polypropylene and WAX FT0070 indicate that dv1, dv2, and dv3 are essential to reduce the initial gross weight. The design region of dv2 for polypropylene is “1” and that for WAX FT0070 is “3”. On the other hand, the design region of dv3 for polypropylene is “3” and that for WAX FT0070 is “1”. As dv2 (fuel length) is shorter, the fuel volume becomes small. And also, as dv3 (initial radius of port) is larger, the fuel volume becomes small. Therefore, Fig. 5 indicates that polypropylene designs dv2 and dv3 to make the initial gross weight smaller compared with WAX FT0070. This difference can be accomplished because the regression rate of polypropylene is greater than that of WAX FT0070.

Result for WAX PW120: Figure 6 shows that dv1, dv2, and dv3 are essential in order to reduce the initial gross weight. Design regions of each design variable depend on the regression rate of each fuel. Figure 6 also indicates that dv7 as the elevation at launch time is important. The physical mechanism that dv7 gives an effect on decreasing the initial gross weight is already mentioned in the previous subsection. Since the regression rate of WAX PW120 is close to that of polypropylene, it is considerable that the generation of this rule merely depends on the difference of the regulation for the minimum support.

Result for GAP50PEG50: Figure 7 shows that only dv1 and dv2 are essential in order to reduce the initial gross weight. Although the regression rate of GAP50PEG50 is small, dv2 as fuel length can be shortened compared with WAX FT0070 which has low regression rate and the rule for dv3 as initial radius of port is not generated. This difference may depend on the density of fuels. However, the knowledge regarding the density of fuels is insufficient. The influence regarding the density of fuels is a future assignment.

Result for GAP60PEG40: Figure 8 shows that the result of generated rules is similar to that for WAX FT0070 because dv1, dv2, dv3, and dv4 is essential in order to reduce the initial gross weight. However, the tendency of the behavior of those design variables is different between WAX FT0070 and GAP60PEG40. Despite GAP60PEG40 has high regression rate, dv2 (fuel length) is higher and dv3 (initial radius of port) is smaller. That is, the fuel volume becomes high. It is considerable that the appropriate fuel volume should be ensured because GAP60PEG40 has high regression rate. Actually, the rule that dv4 (combustion time) is low is generated in Fig. 8. This result indicates that the initial gross weight becomes high so that the large fuel volume suitable for high regression rate should be ensured.

6 Conclusions

The stratum-type association analysis has been applied to a single-stage launch vehicle with hybrid rocket engine constructed by solid fuel and liquid oxidizer in order to obtain design knowledge from the visualization of design-space structure. As a result, the design information has been revealed regarding the tradeoffs among the objective functions. Furthermore, the hierarchical dendrogram indicates the structure of the design space in order to improve the objective functions. The structurization implemented by the stratum-type association analysis indicates the concrete design strategy regarding the significant design variables for the objective functions. The results also reveal the difference of design strategy among the five fuels. In addition, the present application indicates the assignment how to effectively visualize the result of stratum-type association analysis.

References

1. Arias-Montano, A., Coello, C.A.C., Mezura-Montes, E.: Multiobjective evolutionary algorithms in aeronautical and aerospace engineering. IEEE Trans. Evol. Comput. **16**(5), 662–694 (2012)
2. Arthur, W.: The emerging conceptual framework of evolutionary developmental biology. Nature **415**(6873), 757–764 (2002)

3. Baker, J.E.: Adaptive selection methods for genetic algorithms. In: Proceedings of the International Conference on Genetic Algorithms and their Applications, pp. 101–111. Lawrence Erlbaum Associates (1985)
4. Chiba, K.: Evolutionary hybrid computation in view of design information by data mining. In: Proceedings on IEEE Congress on Evolutionary Computation, pp. 3387–3394. IEEE (2013)
5. Chiba, K., Kanazaki, M., Nakamiya, M., Kitagawa, K., Shimada, T.: Diversity of design knowledge for launch vehicle in view of fuels on hybrid rocket engine. J. Adv. Mech. Des. Syst. Manuf. **8**(3), JAMDSM0023, 1–14 (2014)
6. Chiba, K., Makino, Y., Takatoya, T.: Design-informatics approach for intimate configuration of silent supersonic technology demonstrator. J. Aircr. **49**(5), 1200–1211 (2012)
7. Chiba, K., Obayashi, S.: Knowledge discovery in aerodynamic design space for flyback-booster wing using data mining. J. Spacecr. Rockets **45**(5), 975–987 (2008)
8. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
9. Fonseca, C.M., Fleming, P.J.: Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In: Proceedings of the Fifth International Conference on Genetic Algorithms, pp. 416–423. Morgan Kaufmann (1993)
10. Gordon, S., McBride, B.J.: Computer program for calculation of complex chemical equilibrium compositions and applications I. analysis. In: NASA Reference Publication RP-1311. NASA (1994)
11. Hervas-Martinez, C., Ortiz-Bayer, D., Garcia-Pedrajas, N.: Theoretical analysis of the confidence interval based crossover for real-coded genetic algorithms. The 7th International Conference on Parallel Problem Solving from Nature. LNCS 2439, pp. 153–161. Springer, Berlin Heidelberg (2002)
12. Jeong, S., Murayama, M., Yamamoto, K.: Efficient optimization design method using kriging model. J. Aircr. **42**(2), 413–420 (2005)
13. Karabeyoglu, M.A.: Advanced hybrid rockets for future space launch. In: Proceedings on 5th European Conference for Aeronautics and Space Sciences. EUCASS (2013)
14. Keane, A.J.: Statistical improvement criteria for use in multiobjective design optimization. AIAA J. **44**(4), 879–891 (2006)
15. Kosugi, Y., Oyama, A., Fujii, K., Kanazaki, M.: Multidisciplinary and multi-objective design exploration methodology for conceptual design of a hybrid rocket. In: AIAA Paper 2011-1634. AIAA (2011)
16. Oyama, A., Obayashi, S., Nakamura, T.: Real-coded adaptive range genetic algorithm applied to transonic wing optimization. Appl. Soft Comput. **1**(3), 179–187 (2001)
17. Robic, T., Filipic, B.: DEMO: differential evolution for multiobjective optimization. In: The 3rd International Conference on Evolutionary Multi-Criterion Optimization. LNCS 3410, pp. 520–533. Splinger, Guanajuato, Mexico (2005)
18. Sasaki, D., Obayashi, S.: Efficient search for trade-offs by adaptive range multi-objective genetic algorithms. J. Aerosp. Comput. Inf. Commun. **2**(1), 44–64 (2005)
19. Simurda, L., Zilliac, G., Zaseck, C.: High performance hybrid propulsion system for small satellites. In: AIAA Paper 2013-3635. AIAA (2013)
20. Takahashi, M., Kita, H.: A crossover operator using independent component analysis for real-coded genetic algorithms. In: Proceedings of IEEE Congress on Evolutionary Computation 2001, pp. 643–649. IEEE (2001)
21. Watanabe, S., Chiba, Y., Kanazaki, M.: A proposal on analysis support system based on associate rule analysis for non-dominated solutions. In: Proceedings on IEEE World Congress on Computational Intelligence, pp. E–14,855. IEEE, Beijing, China (2014)
22. Yuasa, S., Shiraishi, N., Hirata, K.: Controlling parameters for fuel regression rate of swirling-oxidizer-flow-type hybrid rocket engine. In: AIAA Paper 2012-4106. AIAA (2012)

Creating Optimised Employee Travel Plans

Neil Urquhart and Emma Hart

Abstract The routing of employees who provide services such as home health or social care is a complex problem. When sending an employee between two addresses, there may exist more than one travel option, e.g. public transport or car. In this paper we examine the optimisation of travel plans, supporting the provision of health and social services, with respect to objectives of travel times and estimated CO_2 produced. We include modal choice, either car or public transport as a decision variable. Car travel is normally quicker, but has a higher CO_2 cost, whereas public transport may have longer journey times, but produces less CO_2. We examine a set of problems involving real city and transport network data based in the UK cities of Edinburgh and London. We show that a multi-objective Evolutionary Algorithm can produces Pareto sets of solutions that allow a trade off between CO_2 and travel time through the use of the decision variable.

1 Introduction and Motivation

Many organisations are currently faced with an increasing requirement to reduce their environmental footprint, either due to statutory requirements or to meet aspirational goals set by their employees or customers. For organisations with a mobile workforce this requirement can possibly be met, in part through the increased use of public transport links by the mobile workforce. Public transport, which can encompasses travel modes such as bus, tram and rail and may also include short amounts of walking, has the advantages of low cost and low environmental footprint when compared to car based travel. However it suffers from the disadvantage of potentially being slower for many journeys. Even within developed cities, public transport will have limited

N. Urquhart (✉) · E. Hart
Edinburgh Napier University, 10 Colinton Road Edinburgh,
Edinburgh Eh10 5DT, Scotland
e-mail: n.urquhart@napier.ac.uk

E. Hart
e-mail: e.hart@napier.ac.uk

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_29

coverage with some addresses being too far from the nearest access point or too many individual journey legs being required. In addition, a typical journey by public transport comprises a walking element to the nearest access point, one or more legs by public transport, then an additional walk to the final destination. There may also be short walks involved between individual legs of the journey (e.g. from bus to tram) with the result that some journeys may be considered infeasible by public transport. This paper builds upon the brief introduction made by the authors in [1] and investigates whether an Evolutionary Algorithm can incorporate a mixture of travel modes in order to increase the range of solutions created. Our aim is to explore how public transport links can be introduced within such a mobile workforce scenario and to assess the impact on the objectives of minimising the employee time required to make the visits and the environmental impact of the solution. For the employer, there are two major considerations when examining modal choice: environmental impact (CO_2 emissions) and travel times. Within the context of this paper we will examine the effect of making modal choice a decision variable within a mobile workforce problem to provide schedules that meet time window constraints but offer multiple solutions that trade-off time against carbon emissions.

The remainder of this paper is organised as follows, Sect. 2 presents a review of previous relevant work, Sect. 3 describes the problem instances being investigated, the methodology used (an Evolutionary Algorithm) is described in Sect. 4, results are given in Sect. 5 with conclusions in Sect. 6.

2 Previous Work

This paper brings together two strands of work—that of multi-modal Workforce Routing and Scheduling Problems (WRSP) and vehicle routing to optimise emissions. It extends previous work in bringing together the two issues, and treating the problem as multi-objective in terms of meeting constraints, minimising time, and minimising emissions. In addition, it addresses some issues prevalent in the literature with respect to exploiting realistic data by utilising a government sponsored journey planning service in conjunction with an emissions model.

The WRSP has been investigated by a number of researchers, for a full survey of work in this area the reader is referred to [2]. Typically the problem is formulated around mobile health workers, the aim being to find the optimum allocation of workers to jobs each which have a location and a time window associated with them. The resulting schedule has to respect constraints such as time windows, working hours and qualifications. Approaches include Linear programming, constraint programming and meta-heuristics [3], Markov chains [4] and clustering [5].

An investigation into the WRSP incorporating modal choice for transport was carried out in [6]. A two stage approach is used, the first stage uses constraint programming to produce an initial solution. The second stage attempts to improve the solutions by iteratively apply four meta heuristics—neighbourhood search, memetic algorithm, scatter search and simulated annealing. The modal choices are between cars and public transportation, it is not clear how the public transport data is gathered. The problem is formulated as a single objective problem, with no focus on carbon emissions.

The issues of taking into account environmental factors within vehicle routing is examined by the authors of [7] who examine the routing of light delivery vehicles within an urban setting. The authors consider a mixed fleet of EFV (Environmentally friendly vehicle) and EUF (Environmentally Unfriendly Vehicles). The novel approach taken utilises a neural network to calculate the likely environmental benefits (in terms of air and noise pollution) for each street section and subsequently the Clarke Wright savings algorithm [8] is utilised to find a set of routes that most effectively deploys the EFVs within the fleet to gain the maximum environmental benefits.

The problem under consideration in this paper, may be considered to be a multiplex network [9] in that there is an interaction between the related street network representing car journeys and the public transport network. Within such a multiplex network one network may influence the other, which in turn affects processes utilising the network. As the public transport network can include tram and rail links it is not simply a subgraph of the street graph. The author of [10] presents a useful study looking at the relation between the street graphs in greater London area and New York, to the graphs represented by their respective underground rail networks. They investigate the affect of the underground networks on commuting abilities and travel connectivity across the cities, noting that the underground networks mostly influence journeys to and from the city centres and within the suburbs. Within the problem under discussion, the public transport and car networks are both related, a switch in journey mode for an individual within the WRSP is in practice switching them between travel networks.

3 Problem Instances

The problem instances within this study are based on real-world road and public transport networks and associated emissions factors. The datasets have to encompass actual journey times for the given transport mode, and emissions produced (which requires additional data suc h as speed and distance). This has lead the author to utilse a number of data sources and models which must be combined to produce the datasets required. In particular each of the two cities has differing public transport providers, which leads to differing data sources being used for public transport data.

Eight problem instances are presented, based on two UK cities, Edinburgh and London. The problems are generic instances of routing problems with time windows (often referred to in literature as the Vehicle Routing Problem with Time Windows—VRPTW). Each problem instance comprises a number of client visits which, requiire a member of staff to attend, each visit lasting 30 min and commences within a specified time window which will always start and end between 09:00 and 17:00.

Table 1 Description of the problem instances used

Set	Visits	Window length	No windows
Edinburgh 1	96	8	1
Edinburgh 2	96	4	2
Edinburgh 3	96	2	4
Edinburgh 4	96	1	8
London 1	61	8	1
London 2	61	4	2
London 3	61	2	4
London 4	61	1	8

Four instances, with differing time window arrangements, were created for each city (see Table 1). Note that the Edinburgh1 and London1 instances, effectively have no time-window as visits can be placed anywhere in the 8 h day, whilst in set 4 all visits are allocated a 1 h time window within which the visit must commence.

As the aim of this paper is to investigate the incorporation of multiple travel modes into travel plans these generic instances allow any member of staff to make any visit, also the number of staff available is not limited, but in many real-life problems the use of agency staff allows extra staff be deployed at short notice.

3.1 The Edinburgh Datasets

The Edinburgh problem instances investigated are based upon a set of 96 random addresses within the City of Edinburgh, UK and surrounding district, y, each instance uses the same addresses, but with differing time windows. Public transport provision within the area covered by the problem is mostly provided by buses, but with a limited tram and rail service. The journey data for public transport is supplied by the Transport Direct bulk journey planning software [11]. Which was downloaded and stored in a local database the data and problem instances may be downloaded [12]. Values for emissions for journeys by public transport are supplied by Transport Direct as part of their journey planning data. Car journeys were modelled using street network data obtained from Open StreetMap [13], and the GraphHopper [14] library to compute journey durations. Estimated emissions were obtained by applying emissions factors obtained from the National Atmospheric Emissions Inventory (NAEI) [15] for a medium sized petrol engined motor car.

3.2 The London Datasets

The London datasets are based upon 61 randomly selected addresses in central London. As with the Edinburgh datasets car journey data is derived from GraphHopper and Open StreetMap with car emissions factors derrived using NAEI. Time windows are allocated in the same manner as the Edinburgh datasets. Transport for London (TfL) is the statutory corporation responsible for transportation within the greater London area, TfL coordinate bus services as well as operating the Underground and Overgound rail networks. Data is obtained via an API [16] provided by TfL, emissions values are taken from statistics provided by TfL [17].

4 Methodology

4.1 The Evolutionary Algorithm

The Evolutionary Algorithm used is the Non-dominated Sorting Genetic Algorithm II (NSGA-II) algorithm [18, 19], with the twin objectives of reducing the predicted emissions generated by the solution and reducing the time required to make the visits. NSGA-II produces a set of non-dominated solutions to the problem under consideration allowing the final choice of solution to be left to the end user, who may examine solutions and choose the one which best fits their current requirements. Two objectives are defined as follows:

- *Minimise total time* $T = \sum_{i=1}^{n} t_i$, where n is the number of employees, t_i is the total journey time for employee i, calculated as the time between leaving and returning to the office, and including travel time, time spent on visits and any waiting time incurred due to arriving at a visit prior to the time window.
- *Minimise total carbon emissions* $E = \sum_{i=1}^{n} e_i$, where e_i is the emissions associated with the journeys made by employee i and calculated according to the emissions values stored in the database as described in Sect. 3.

Each problem assumes an unlimited number of employees are available for allocating work too, therefore all solutions are *valid* in that they guarantee that all work items are scheduled.

The genotype representation used within the EA is that of a grand tour [20], comprising a permutation of visits, these permuatations are fed toa decoder (see Sect. 4.2), that builds a travel plan which can then be evaluated. Each visit comprises the visit identifier (which is linked to the location and time window data) and a travel mode identifier. The travel mode identifier assigns a travel mode for that visit, a value of 1 for public transport, a value of 0 for car transport. Hence, the EA must evolve must optimise the ordering of vsits fed to the decoder, and the mode of travel that should be assigned for that visit.

With the EA a fixed sized population of 300 solutions is used, the size being determined by empirical experimentation. Each new population is the same size as the previous population, the parents being selected by binary tournament from the previous population. Children are created using a permutation crossover operator that copies a complete route, selected at random, from one parent, remaining routes, less any duplicate genes, are copied from the other parent in the order they appear and are appended to the child. Each child has one of two mutation operators, selected at random, applied to it. The first operator selects an entry from the grand tour at random and moves it to a new randomly selected position within the grand tour. The second mutation operator selects a visit at random from the grand tour and 'flips' the travel mode.

4.2 Building Travel Plans

A solution comprises a travel plan, which consists of a number of individual employee tours which collectively cover all of the visits. A solution is constrcuted from the grand tour gentotype using a decoder which considers each travel mode in turn, firstly public transport and secondly car based visits in the order that they appear within the grand tour. Visits are initially added to the first employee, as each visit is added the arrival time is calculated based upon the journey times (see Sect. 3) for that mode of transport. If an arrival time is earlier than the commencement of the time window then the employee waits until the start of the time window in order to make the visit. If the arrival time is after the end of the time window then a new employee tour is added to the solution and the visit allocated to the new tour. This decoding process continues until of the visits for the current travel mode have been added. The decoder then passes through the grand tour considering visits with next travel mode. In this way each pass through the grand tour creates a set of employee routes for that travel mode. Note that when considering public transport, not all journeys are feasible, if a visit has been allocated to public transport, but travel from the previous visit is not possible, then the travel mode is changed to car and the decoder continues. This repair mechanism ensures that a feasible solution is always constructed, by using car transport as the default choice. A pseudo code implementation of the decoder is presented in Algorithm 1. After applying the decoder, the carbon emissions e_i and time t_i associated with each individual tour can be evaluated and therefore the values of the two objectives.

4.3 Experiments

For each dataset described in Table 1 two initial experiments were performed:

- Using only car based transport—labelled in the results as *car*
- Using a combination of car and public transport—labelled in the results as *pTrans*

The EA was limited to 1,000,000 evaluations on each run as empirical experimentation demonstrated that beyond that value improvements in the quality of solution

Algorithm 1 The decoder used to transform the grand tour genotype into the phenotype.

```
procedure DECODE(chromosome)
    solution = new Phenotype            ▷ The chromosome willl be expanded into a set of routes
    for travel-mode= 0 to MAX-MODE do                          ▷ Cycle through each travel mode
        current-route = new Route(travel-mode)                 ▷ Add a new route to the solution
        solution.add(current-route)
        previousLoc = OFFICE-BASE
        for each gene in chromosome do
            if gene.getMode() == travel-mode then
                if (feasibleTrip(previuousLoc,gene,current-mode) then   ▷ True if the travel mode feasible
                    if (feasibleVisit(gene) then        ▷ True if the visit can be made within the time window
                        current-route.add(gene)
                        previousLoc = gene
                    else                                       ▷ Add a new route to the solution
                        current-route = new Route
                        current-route.setMode(gene.travel-mode)
                        current-route.add(gene)
                        previousLoc = gene
                    end if
                else
                    gene.setMode(CAR)        ▷ If trip not feasible by current mode, default to car
                end if
            end if
        end for
    end for
    return solution
end procedure
```

were seldom noted. Each run of the solver was repeated 10 times in order to obtain a set of results. The Pareto-fronts obtained from the 10 runs were combined and plotted as described in the next section.

5 Results

Figure 2 shows the non-dominated fronts obtained when combining the output of 10 runs, using only car transport (in black) and using a combination of car and public transport. A summary of all results may be found in Table 2. Considering the car-only solutions as we would expect the output forms a curve ranging from high CO_2 solutions taking less time than the lower CO_2 solutions at the other end of the curve. Shown alongside this is the curve generated by combining the output from 10 runs where both public transport and car travel is used. As the *pt* runs control the decision variable we would expect them to be able to create car-only runs which would cover the same solution space as the *car* runs resulting in a pareto front that overlaps that

Table 2 Key: Time—minutes, emissions—CO_2 g. A summary of the best solutions found with and without using public transport links. The figures in brackets within the time columns represent the number of employees required within the solution

Instance	Criterion for best	Car only		Public Transport		Modified	
		Time	Emissions	Time	Emissions	Time	Emissions
Edin 1 window	Time	3369	119995.34	3474.00	118917.03	3358	119620.88
	Emissions	3432	94471.37	4778.01	73986.22	4725	79761.21
Edin 2 windows	Time	3576	138192.45	3835	114682.47	3585	130331.66
	Emissions	4091	104052.55	5627	74084.64	5347	76455.2
Edin 4 windows	Time	3779	123826.16	4154	161021.05	3768	140161.99
	Emissions	4532	108137.63	5936.01	85692.63	6752.01	92530.02
Edin 8 windows	Time	3932	181014.68	4277.01	195616.28	3998.01	179210.1
	Emissions	6618.01	114441.52	6719.01	82449.23	7062.01	94264.53
London 1 window	Time	2176	43456	2258	34931.14	2194	38685
	Emissions	2286	36252	3429	9292.98	3523	9104.5
London 2 windows	Time	2321	45322	2550	35318.07	2381	48022
	Emissions	3537	40226	4185	9621.33	4638	10071.96
London 4 windows	Time	2411	51454	2813	41631.31	2423	56755
	Emissions	3080	42195	4886	11296.9	4576	11966.33
London 8 windows	Time	2544	66842	2796	58451.19	2542	72612
	Emissions	3503	50985	4868	14019.83	5266	14148.74

produced from the *car* runs. Viewing Fig. 2 shows that this is not the case, none of the graphs show the expected overlap. We note that in no case were the public transport based solutions able to find a solution with the same as or less time than the car-only solution, although in some cases (graphs a, c and d) high CO_2 solutions were found, but none of them exhibited low time values. Comparing between the two cities we note that for the London based problems (graphs e–h) the solver rarely finds high CO_2 solutions unlike the Edinburgh based datasets (Fig. 1).

When considering the results to be returned to the user, it is important that the non-dominated set encompasses extreme solutions, for instance the all car high CI_2 low time solution and the all public transport low CO_2 high time value solution, clearly the results presented in Fig. 2 do not encompass those solutions. We may surmise that the fitness landscape is not conducive to NSGA-II exploring car-only solutions. We investigate two possible approaches to resolving this issue the results of which

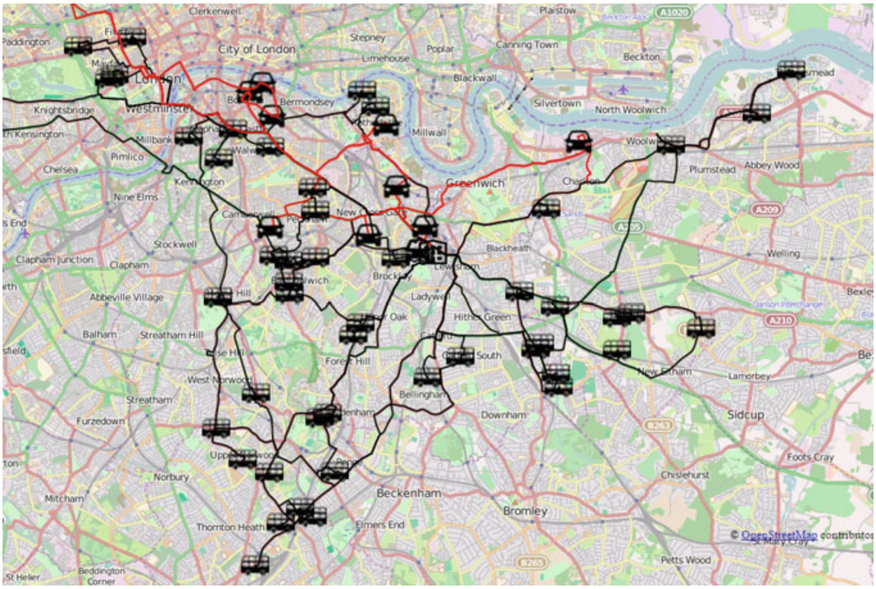
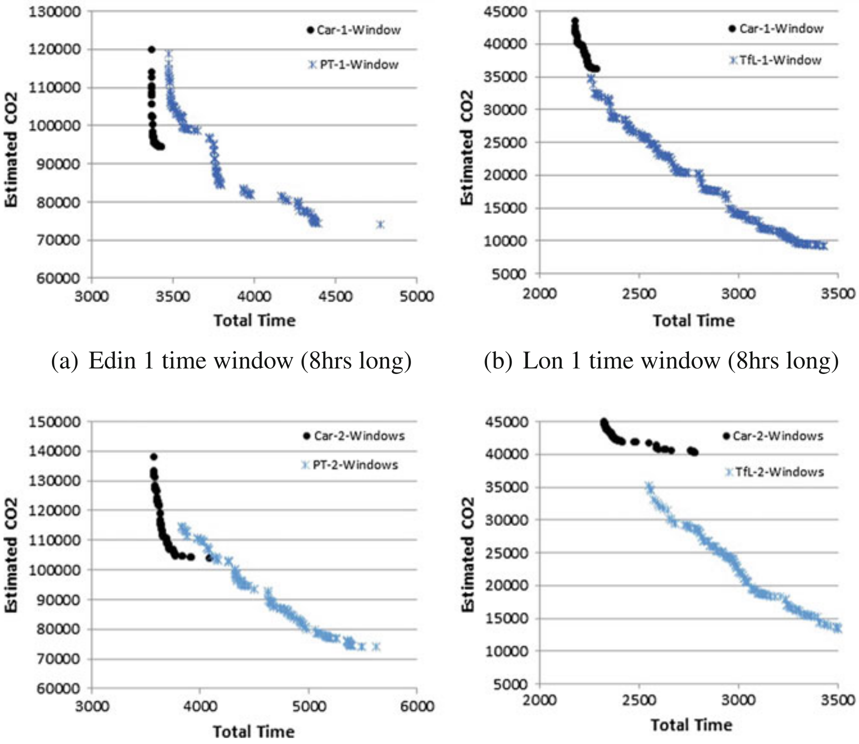


Fig. 1 An example of a solution constructed from the london data set. The start point is represented by the office building in the middle of the map, visits made by car are noted by the car icon and visits by public transport by the bus icon. Map data © OpenStreetMap contributors using the Open Database License. https://www.openstreetmap.org/copyright

are shown in Fig. 3. The initial approach taken was to combine the outputs of the *car* and *pt* runs to produce a combined set of non-dominated solutions, these are labelled *comb*. Within Fig. 3 we can see that combining the outputs produces a Pareto set that encompasses the desired range of solutions. In some cases (noticeably the Edinburgh datasets) we can clearly see a gap in the Pareto front which would appear to delineate the *car* and *pt* solution spaces. Whilst combining outputs produces a useful set of solutions, it does has the disadvantage that it requires the algorithm to be executed twice, thus potentially increasing the runtime by a factor of 2. A second approach towards finding low time, high CO_2 solutions utilises a second mutation operator. The new mutation operator sets all of the travel mode identifiers to 0 (car travel). Whilst this may seem extreme, experiments with only setting part of the chromosome to car did not produce any solutions within the desired area. The rate for this chromosome had to be set to 0.60 (i.e. applied to 60% of all child solutions) in order to produce solutions in the desired space. Figure 3 show the mutation operator achieving a rang of solutions comparable to those achieved by combining the separate results. Figure 3 graphs a-d show that when applied to the Edinburgh dataset the mutation operator results in the production of low time, high CO_2 solutions, but has a detrimental effect on the production of low CO_2 solutions.



(a) Edin 1 time window (8hrs long)

(b) Lon 1 time window (8hrs long)

(c) Edin 2 time windows (4 hrs long)

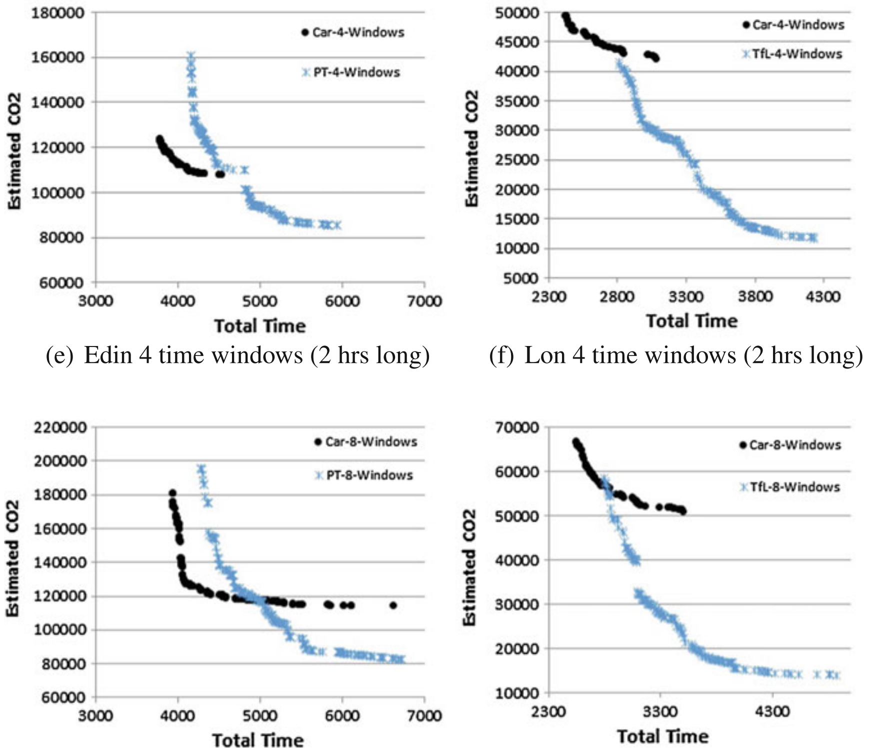
(d) Lon 2 time windows (4 hrs long)

Fig. 2 Total solutions produced for each problem instance using only car transport and combining car (shown as *car*) and public transport (shown as *PT*)

6 Conclusions

The paper set out to investigate the trade-offs that could be obtained in terms of time and emissions when scheduling a mobile workforce to complete a set of customer visits. Using data from a real public transport and road network, a multi-objective EA was used to evolve solutions under three scenarios: using a car only, using mixed modes of transport and using mixed modes with an added constraint that excluded some potential routes based on public transport due to excessive walking or change-overs. Experiments investigated the trade-offs between the two objectives in four different scenarios in which the time-window for a visit was varied between 1 and 8 h.

Based on the evidence presented in Sect. 5 the NSGA-II based solver is capable of producing solutions that make use of public transport links in order to reduce CO_2. The aim of using NSGA-II is to give the user a choice of non-dominated solutions to choose from, in this case to allow the user the option of trading off CO_2 against



(e) Edin 4 time windows (2 hrs long)

(f) Lon 4 time windows (2 hrs long)

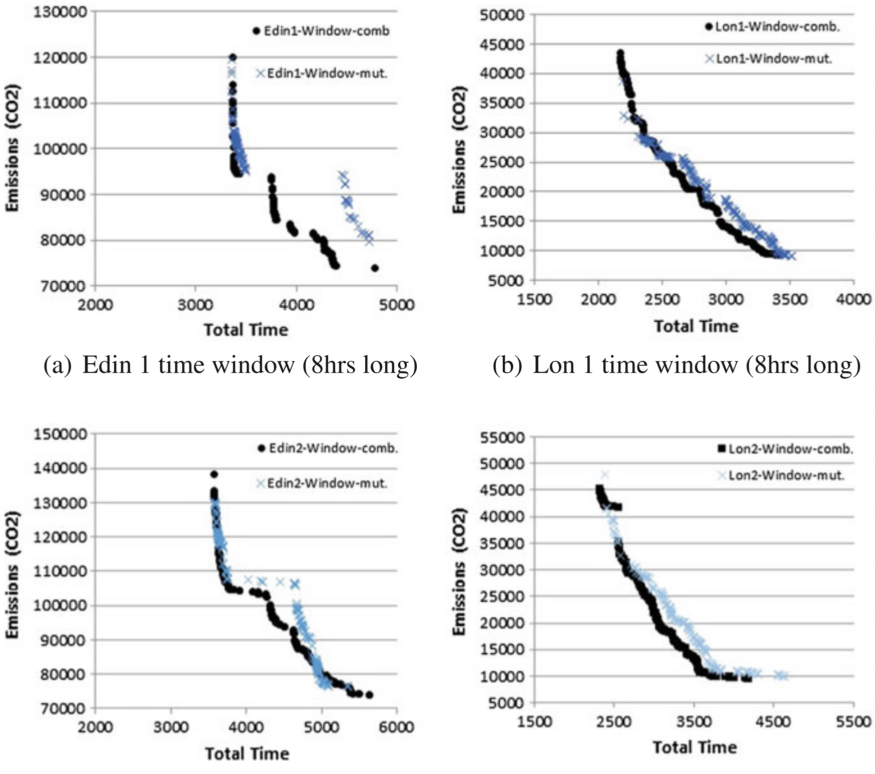
(g) Edin 8 time windows (1 hr long)

(h) Lon 8 time windows (1 hr long)

Fig. 2 (continued)

travel time by allowing public transport to be included within the solution. As initially presented (Fig. 2) the NSGA-II solver has difficulty in evolving a set of solutions that encompass low time values, despite the car only based solutions proving that low time solutions are feasible. This problem is overcome by the simplistic expedient of combining the Pareto fronts produced using car only solutions and public transport with car solutions, but this has a penalty in the form of increased run times.

Finally, we note that a number of commercial geographical information systems offer public transport data, but currently the costs involved in acquiring data for a problem such as this make them uneconomic. Storing the data in a local database as described not only reduces the requirement for repeated access to on line databases but also decreases the running time of the evolutionary algorithm. However, it is recognised that a future extension of the system would include frequent updating of the database in order to take account of changing travel conditions.



(a) Edin 1 time window (8hrs long)

(b) Lon 1 time window (8hrs long)

(c) Edin 2 time windows (4 hrs long)

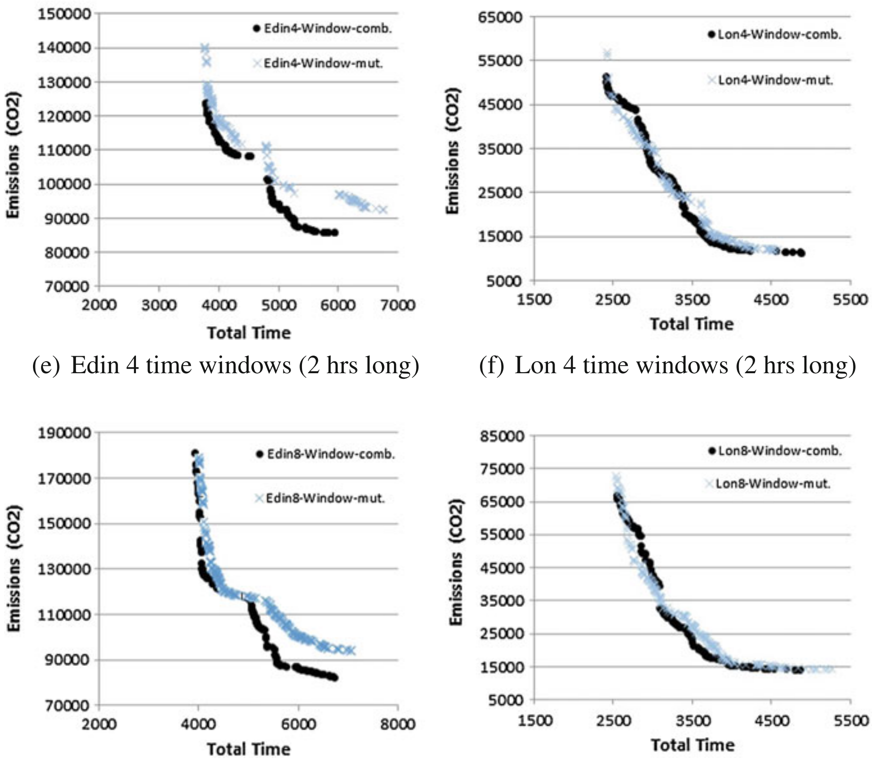
(d) Lon 2 time windows (4 hrs long)

Fig. 3 Total solutions produced for each problem instance by combing the outputs from Fig. 2 noted as *comb* and the modified mutation noted as *mut*

7 Future Work

In the immediate future further investigation of the issues involved in generating a reasonable spread of solutions are to be investigated, both of the solutions proposed in this paper have disadvantages (increased runtime and inconsistent performance). The nature of the non-dominated fronts is such that they may be combined, easily into a front that combines the non-dominated elements of its constituants. This property may be combined with algoritm portfolios to allow multiple algorithms to contribute solutions to a single non-dominated front.

The problem instances may be further developed in a number of ways, the binary model of transport modes (car or public transport) may be considered to be overlay simplistic, in that other variations and modes exist such It is hoped to undertake further studies based on real-world mobile work force problems, and to increase the constraints within the problem to match those typically found in such problems, such as some visits requiring specific workers or workers with a specific attribute or



(e) Edin 4 time windows (2 hrs long)

(f) Lon 4 time windows (2 hrs long)

(g) Edin 8 time windows (1 hr long)

(h) Lon 8 time windows (1 hr long)

Fig. 3 (continued)

multiple workers. It is also planned to expand the problem formulation to include more transport modes, such as car share, cycling and taxis, such a formulation could include transport costs as one of its optimisation criterion. Future work will also include scaling journey times a specific times in order to reflect rush hour congestion, such scaling would affect road based transport to a far greater degree than rail transport. The scenario investigated here is simplistic, future work will examine problems where there are fixed numbers of staff and constraints on travel modes (e.g. not all staff can drive). We recognise that the problem of finding and optimising transport routes through a city is not restricted to workers, but also occurs in domestic and social situations as well, in the longer term an investigation into optimising travel activities in these areas would be appropriate.

References

1. Urquhart, N., Hart, E., Judson, A.: Multi-modal employee routing with time windows in an urban environment. Genetic and Evolutionary Computation Conference (GECCO) Companion 2015, (2015)
2. Castillo-Salazar, J.A., Landa-Silva, D., Qu, R.: A survey on workforce scheduling and routing problems. In: Proceedings of the 9th international conference on the practice and theory of automated timetabling, pp. 283–302 (2012)
3. Bertels, S., Fafle, T.A.: A hybrid setup for a hybrid scenario: Combining heuristics for the home health care problem. Comput. Oper. Res. **33**(10) (2006)
4. Out, P.M., Bhulai, S., Van Meersbergen, M.: Optimal patient and personnel scheduling policies for care-at-home service facilities. Eur. J. Oper. Res. **219**(3) (2012)
5. Rasmussen, M.S., Justesen, T., Dohn, A., Larsen, J.: The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. Eur. J. Oper. Res. **219**, 598–610 (2012)
6. Hiermann, G., Prandtstetter, M., Rendl, A., Puchinger, J., Raidl, G.: Metaheuristics for solving a multimodal home-healthcare scheduling problem. Cent. Eur. J. Oper. Res. **23**(1) (2013)
7. Dragan Pamuar Goran irovi. Green logistic vehicle routing problem: Routing light delivery vehicles in urban areas using a neuro-fuzzy model. Expert Syst. Appl. **41**(9), 4245–4258 (2014)
8. Clarke, G., Wright, J.W.: Scheduling of vehicles from a central depot to a number of delivery points. Oper. Res. **12**, 568–581 (1964)
9. Gomez-Gardenes, J., Reinares, I., Arenas, A., Floria, L.M.: Evolution of cooperation in multiplex networks. Sci. Rep. **2** (August) (2012)
10. Dobson, S.: Multiplex cities: Interacting transport networks in metropolitan areas. University of St Andrews. https://www.simondobson.org/static/sd/softcopy/urban-networks-sicsa.pdf (2015)
11. Transport Direct.: http://www.transportdirect.info (2014)
12. Urquhart, N.: Public Transport Journey Data (2015)
13. Open StreetMap Foundation.: http://www.openstreetmap.org (2014)
14. Karich, P.G.: https://graphhopper.com/ (2014)
15. Department for the environment fodd and rural affairs.: National Atmospheric Imissions Inventory (NAEI) http://www.naei.org.uk (2009)
16. Transport For London.: Transport for London Public. http://api.tfl.gov.uk/k (2015)
17. Transport For London.: Travel in London, key trends and developments. report number 1. http://www.tfl.gov.uk/cdn/static/cms/documents/Travel-in-London-report-1.pdf (2015)
18. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast elitist multi-objective genetic algorithm: IEEE Trans. Evol. Comput. **6**, 182–197 (2000)
19. Seshadri, A.: Nsga-ii a multi-objective optimization algorithm (2006)
20. Beasley, J.: Route first-cluster second methods for vehicle routing. Omega **11**(4), 403–408 (1983)

A New Rich Vehicle Routing Problem Model and Benchmark Resource

Kevin Sim, Emma Hart, Neil Urquhart and Tim Pigden

Abstract We describe a new rich VRP model that captures many real-world constraints, following a recently proposed taxonomy that addresses both scenario and problem physical characteristics. The model is used to generate 4800 new instances of rich VRPs which is made freely available. To the best of our knowledge this represents the most comprehensive resource of rich VRP problems available, and provides a platform for researchers to conduct rigorous comparisons of new methods and solvers, moving academic research much closer to real practice in the future.

1 Introduction

More than half of century since Dantzig and Ramser [1] first introduced the Vehicle Routing Problem (VRP) in 1959, academics and practitioners continue to actively explore variants of this problem and introduce new methods to provide solutions. A survey in 2009 by Laporte [2] summarised the state-of-the-art in research into the classical VRP charting the developments in exact methods and heuristics, noting that heuristics now enable realistically sized instances of VRP to be solved adequately. However, the author points out the gap between classical VRP models and the stochastic and dynamic features of problems that are apparent in many real world problems. Caceres-Cruz et al. [3] go further in recognising that in addition to

K. Sim (✉) · E. Hart · N. Urquhart
Centre for Algorithms, Visualisation and Evolving Systems, Edinburgh Napier University,
Merchiston Campus, Edinburgh EH10 5DT, Scotland
e-mail: k.sim@napier.ac.uk

E. Hart
e-mail: e.hart@napier.ac.uk

N. Urquhart
e-mail: n.urquhart@napier.ac.uk

T. Pigden
Optrak Distribution Software Ltd., Ware, Hertfordshire, UK
e-mail: tim.pigden@optrak.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods
in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_30

stochastic and dynamic features, real VRP problems deal with multi-objective optimisation functions and a wide variety of constraints that cover factors such as heterogeneous fleet, time factors relating to congestion, the need to combine routing with driver scheduling and the increasing need to provide compliance with environmental regulations. In response to this, the term *rich VRP* has been coined to describe VRP instances that account for some or all of these factors.

Much research has been published under this heading: two recent surveys [3, 4] provide a detailed summary of the state-of-the art including problem combinations, constraints defined, and approaches found, demonstrating the rich variety of problems now being tackled. However, we note that unlike in the classical VRP literature, it is difficult to find *datasets* containing sufficient benchmark instances to fairly evaluate new approaches; many of the published works refer to real-world problems where only a few instances are described and often datasets are not published. To address this issue, we describe a new model of a rich VRP that encapsulates many real constraints, informed by input from practitioners.[1] The model is instantiated in an extensible software-framework that is straightforward to extend with additional constraints. Using the framework, we generate a significant resource of 4800 problems that are made freely available. Instances cover a wide range of sizes and are generated on a map of the UK; driving distances and times are generated from mapping software utilising actual road-networks and thus potentially includes asymmetric distances between points. The instances provide a foundation for rigorous comparisons of algorithms and methods, that it is hoped will drive forward academic research within this area in the future.

In the next section we provide some background to the RVRP, grounding our model in a recently proposed taxonomy for RVRP. The software framework and problem instances are then described. All instances are downloadable from [5].

2 Background

Motivated by the need to address the gap that is clearly apparent between the complex characteristics of real-life VRPs and much academic research, increasingly more complex variants of the classical VRP problem are tackled, often referred to as *Rich VRP* problems. However, Lahyani et al. recently observed that across the wide-range of research published under this heading, there was no precise definiton of what criteria led to a problem being classified as *rich*, and proposed a generic taxonomy for RVRP. The taxonomy differentiates between Scenario Characteristics and Problem Characteristics; within each class, subclasses describe characteristics at the *strategic*, *tactical* and *operational* levels within a hierarchical structure. The taxonomy is shown in Table 1 (note that no characteristics at the operational level are described for Scenarios).

[1] Optrak Vehicle Routing Software http://www.optrak.com

Table 1 Rich VRP Taxonomy from [4]: characteristics reflected in the proposed model are shown in *italics*

Scenario characteristics

Strategic	Tactical
Input data	*Static*
	Dynamic
	Deterministic
	Stochastic
Decision management options	Routing
	Inventory and routing
	Location and routing
	Routing and driver scheduling
	Production/distribution planning
Number of depots	*Single*
	Multiple
Operation Type	Pickup or *delivery*
	Pickup and delivery
	Backhauls
	Dial-a-ride
Load splitting constraints	*Splitting Allowed*
	Splitting not allowed
Planning period	*Single period*
	Multi-period
Multiple use of vehicles	Single trip
	Multi-trip

Physical characteristics

Strategic	Tactical	Operational
Vehicles	Type	Homogeneous *Heterogeneous*
	Number	*Fixed* Unlimited
	Structure	*Compartmentalised* Not compartmentalised
	Loading Policy	chronological *no order*
	Capacity constraints	
Time constraints	*Restriction on customer*	
	Restriction on road access	

(continued)

Table 1 (continued)

Scenario characteristics

Strategic	Tactical	
	Restriction on depot	
	Service time	
	Waiting time	
	Driver regulations	
Time window structure	*Single time window* Multiple time windows	
Incompatibility constraints	*Equipment* *Compartment restrictions*	
Specific Constraints		
Objective Function	Single Objective *Multi-objective*	

Based on this taxonomy and an extensive analysis of literature, Lahaynai et al. propose a precise definition of a VRP, given below:

Definition 1 RVRP extends the academic variants of the VRP in the different decision levels by considering at least four strategic and tactical aspects in the distribution system and including at least six different daily restrictions related to the physical characteristics. When a VRP is mainly defined through strategic and tactical aspects, at least five of them are present in a RVRP. When a VRP is mainly defined through physical characteristics, at least nine of them are present in a RVRP.

The authors note that under this definition, many papers previously considered as 'rich' may not comply. Caceres-Crux et al. note that in fact, the RVRP definition evolves continuously and conclude that "*the RVRP reflects, as a model, most of the relevant attributes of a real-life vehicle-routing distribution system*", and therefore should be seem as an accurate representation of a real-life distribution system, and hence methods for solving instances characterised as RVRP should be directly applicable to a real-life scenario.

In order for research in developing new methods that deal with the complex set of constraints associated with RVRP to progress, it is beneficial for the community to have access to a resource of relevant benchmark problems on which approaches can be rigorously compared. However, unlike in classical VRP in which many datasets are well known [6], an equivalent resource of RVRP problems is not readily available. For example, Lahyani et al. [4] survey 16 recent papers relating to pure vehicle routing that describe rich problems. Of these 15 of them focus on real world problems, each with very specific constraints, and hence do not provide a wide ranging suite of benchmarks that enable algorithms to be easily and fairly compared. On the other hand, the non-rich VRP literature abounds with benchmarks datasets. We briefly examine these before proposing a new rich VRP model that is easily extensible, and a freely available problem generator that can be used to generate multiple instances

2.1 Benchmark Data for VRP

A number of web-based resources provided aggregated collections of VRP datasets. The VRP-REP project [6] aims to provide the VRP community with a collaborative open data platform, enabling users to share instance files, check solutions and track solutions. It currently provides 34 datasets, the largest of which contains 180 instances. The majority of these datasets do not meet the definition of RVRP, although the newer additions move towards this (for example Mendoza et al. [7] provide a dataset that models stochastic demands and duration constraints). The Networking and Emerging Optimization group [8] provide datasets for capacitated VRP, including those with time-windows and pickups and deliveries, as well as multi-depot and periodic problems. More recently, Goeke and Schneider [9] provide instances for a variant of VRP that includes electric vehicles, time-windows and recharging constraints.

Although many distinct data-sets are available, the relatively small size of each set limits the extent to which rigorous algorithms comparisons can be made, particularly in research using approximation methods that can deal with large instances and find solutions in relatively short amounts of time. A specific case of this is recent research in *hyper-heuristics* which deals with methods that find cheap but acceptable solutions that generalise over very large problem sets and are often tested with problem databases containing 1000 s of instances.

Recognising the dual need for new benchmarks that capture rich vehicle routing problems that reflect real-world constraints and diverse collections of benchmarks to encourage comparative research, we propose a new RVRP model. The model is translated to an object-oriented software framework that is easily extensible, therefore new scenario and problem physical constraints can be added in future. We provide 4800 problem instances of varying size and complexity for future research. The model is described in detail in the next section.

3 Model

The proposed model aligns with the taxonomy proposed by Lahyani [4] and shown in Table 1. In this table, features of the new model are shown in italics within the taxonomy. Note that the model contains one new characteristic at the *strategic* level, in the use of real road-network data to calculate distances and times.

3.1 Scenario Characteristics

We describe a *static* problem p given by the tuple $p = \{J, V, d, M, s, E\}$ where J is a set of jobs to be delivered, V is a set of vehicles, d defines a *single* depot location and M is a set of 2 matrices giving the asymmetrical distances and times between all locations. Only *Routing* is considered, i.e. it is assumed that a driver will be available for a particular vehicle at all times. *Planning* is over a single continuous time period given by s. *Multi-trips* by the same vehicle are allowed in a day. The remaining term E defines the time constraints imposed on the driver and are described in Sect. 3.2.2.

We consider *delivery* problems only. Each job $j_i \in J$ is described by $j_i = \{l_i, U_i, C_i\}$. l_i gives the location of the customer the job is to be delivered to, U_i is the set of pallets of a product to be delivered. $C_i = \{w_i, t_i, q_i, \alpha_i, \beta_i, \gamma_i, \delta_i\}$ describe the set of constraints associated with the physical characteristics of a job and are described in Sects. 3.2.2 and 3.2.3. *Splitting* is allowed—the pallets associated with a job j_i can be delivered by multiple vehicles.

3.2 Scenario Characteristics

3.2.1 Vehicles

A *fixed* and *heterogeneous* vehicle fleet V is defined for each problem.

Each *compartmentalised* vehicle $v_j \in V$ is described by $v_j = \{T_j, e_j, q_j\}$ where T_j describes the set of temperature controlled compartments on the vehicle, e_j defines the vehicle's engine type and q_j indicates the availability of any additional equipment (tailgate) for the vehicle. Each compartment $t_k \in T_j$ is described by $t_k = \{r_k, n_k\}$ where r_k defines the compartment type and n_k indicates the capacity of the compartment. The values of r_k, n_k, e_j, q_j considered are listed below:

- $r_k \in \{FROZEN, REFRIGERATED, AMBIENT\}$
- $n_k \in \{6, 12, 24\}$
- $e_j \in \{DIESEL, HYBRID, GAS\}$
- $q_j \in \{TAILGATE, NONE\}$

Vehicles do not have a loading policy, that is, compartments can loaded and unloaded in any order.

3.2.2 Time Constraints and Time-Windows

Each problem has employee constraints $E = \{x, b\}$ which define the maximum driving time allowed x before a compulsory break of length b is imposed. The set of physical constraints associated with a job j_i, given by $C_i = \{w_i, t_i, q_i, \alpha_i, \beta_i, \gamma_i, \delta_i\}$, are described in this and the following section. A single time-window for delivery w_i is assumed for each job (*customer restrictions*) during which the job should *arrive* at the customer location. In addition, each job has 4 terms defining *service times* $(\alpha_i, \beta_i, \gamma_i, \delta_i)$. α_i and β_i give the fixed loading time per job and loading time per pallet respectively at the depot. These terms are not applied for the first route assigned to a vehicle as this is generally completed prior to the service period. Similarly γ_i and δ_i define the fixed unloading time per job and unloading time per pallet at the customer's location. All durations are defined in minutes.

It is assumed that all vehicles are available for the duration of the planning period during which all routes must start and finish at the depot. When a vehicle (and its driver) return the depot, it immediately becomes available for reuse (the driver's cumulative driving time is reset).

3.2.3 Incompatibility Constraints

All pallets from a job have the same type $t_i \in \{FROZEN, REFRIGERATED, AMBIENT\}$ and must be placed in a compartment of the corresponding type. Some customers require that the vehicle that a load is delivered on must have particular equipment given by $q_i \in \{TAILGATE, NONE\}$.

3.2.4 Driving Times/Distances

The manner in which driving times are computed is not included in the RVRP taxonomy. We consider routing performed on a real road-network. Distances and times between any two locations are obtained from the open-source mapping product GraphHopper [10]. This returns *asymmetric* distances (and times) between any two points (e.g. accounting for one-way streets). Travel times are stored in the matrix m_t and distances in m_d. In the model supplied, the time between any two points is fixed and therefore does not account for rush-hour driving. The model could be extended to account for this using stepwise linear functions that then have to be combined in order to calculate (and minimise) travel times across the day.

3.2.5 Customer Locations

The depot and the customers are located at geographical locations derived from actual postcodes within the UK. In the model supplied, customer locations are selected at random from all locations listed in the UK postcode database that can be serviced within a maximum time-scale from the depot. This process ensures a realistic distribution of customer locations based on the actual distribution of residential and retail dwellings in the geographic area surrounding the depot.

3.2.6 Cost Model

Vehicles incur different costs depending on their size and engine type. Three costs are considered; fixed, running and environmental costs. The problem model does not contain actual costs but contains 3 multipliers for each vehicle $v \in V$ which show the relative costs compared to a *standard* vehicle which we define as a *large* vehicle fitted with a *diesel* engine having one large ambient container of capacity *24* product units. Tables 2 and 3 show the values used to calculate the cost multipliers based on the vehicle's engine type and total capacity respectively. The 3 cost multipliers supplied with each vehicle are calculated by multiplying the values obtained from each of the two tables corresponding to the vehicles size and engine type.

The actual costs associated with any vehicle can be calculated by scaling the costs incurred by a standard vehicle. Suggested costs for the standard vehicle type, based on current market prices, are given in Table 4. The actual costs associated with any vehicle are described and calculated as follows.

- **Fixed Costs** fc_j is the fixed cost associated with vehicle j and is calculated by multiplying the fixed cost multiplier of that vehicle by the standard vehicle cost per day. The standard vehicle fixed cost per day typically includes driver costs / depreciation, servicing, taxes and other costs that are independent of the distance covered by the vehicle.
- **Running Costs** dc_r is the running cost associated with route r which varies depending on the distance covered and the vehicle used. This is calculated as the standard vehicle running cost per km * distance covered by route * vehicle running cost multiplier. The standard vehicle cost per KM typically includes fuel, tyre wear and other distance dependent costs.
- **Environmental Costs** ec_r is the environmental cost associated with route r which varies depending on the distance covered and the vehicle used. This is calculated

Table 2 Cost multipliers corresponding to engine type

	Cost Multiplier		
Engine Type	Fixed	Running	CO$_2$
Diesel	1.00	1.00	1.00
Hybrid	1.07	0.80	0.80
Gas	1.04	0.70	0.95

Table 3 Cost multipliers corresponding to vehicle size

Vehicle size (capacity)	Cost multiplier
Small (6)	0.35
Medium (12)	0.60
Large (24)	1.00

Table 4 Costs associated with the standard vehicle type

Cost	0.015 Value
Fixed cost	242/day [11]
Running cost	0.36/km [11]
Environmental cost	0.015/km [12]

as the standard vehicle environmental cost per km * distance covered by route * vehicle environmental cost multiplier.

3.2.7 Objectives

Any single objective function can be defined as a sum over the costs described; alternatively, the problem can be treated as a multi-objective problem. An example single objective function that takes into consideration all the costs described as well as penalties incurred for breaking soft constraints is given by:

$$f = \sum_{j=1}^{n} fc_j + \sum_{r \in R} dc_r + \sum_{r \in R} ec_r + Z_l + Z_o$$

The fixed cost fc_j is applied once for each vehicle used (any unused vehicle does not incur a cost). The total running cost $\sum_{r \in R} dc_r$ and environmental costs $\sum_{r \in R} ec_r$ are summed over all routes R conducted by all vehicles. $Z_l + Z_o$ are financial penalties imposed as described below.

- **Time Penalties** Z_l is the late/early penalty which is incurred for all pallets arriving at a customer's location outside of the specified time window. Each pallet arriving outside the time window incurs a fixed penalty of (£1 $\times$ $time$)^2 where $time$ is the the number of minutes that the delivery is either early or late. Research on real world problems [] has shown that using a quadratic function to penalise broken time windows minimises the number of potential solutions that incur equal penalties.
- **Outsourcing Penalty** Z_o is the outsourced pallet penalty per pallet per KM. This is calculated using $arg_{max}\{£5/pallet/KM, £50/pallet\}$ where the distance is measured from the depot to the associated customers location. The second term defines a minimum cost for outsourcing a pallet.

Solutions can be evaluated using the suggested metric or may be evaluated using any user specified objective function. The model allows for relaxation of both hard and soft constraints as required. For example the end user could ignore suggested penalties for breaking soft constraints and disregard all costs focussing only on total distance, effectively turning the RVRP model into a conventional VRP. Hard constraints could also be ignored. For example jobs marked as having to be transported on an ambient container could be allowed to be delivered on any container type. This abstraction of the objective function allows the RVRP model implemented to be easily transformed using different scenarios.

4 Problem Generation

The section describes the process of generating the 4800 problem instances that are made available in XML format. For each problem $p = \{J, V, d, M, s, E\}$, the service period s during which all customers should be serviced is fixed to between 9am and 5pm on the 1st January 2015. The driver constraints $E = \{x, b\}$ representing the maximum continuous working time and break length are fixed for all problems at 270 min before which a 45 min break is enforced such that the driver never works more than 270 min continuously. The break is applied either before (where unloading takes the cumulative working time over the threshold) or after a job is unloaded (where the next period of driving breaks the maximum working time). A driver returning to the depot has their cumulative working time reset. All instances generated have valid solutions that meet all hard and soft constraints without requiring any pallets to be outsourced. The fixed and variable loading and unloading times for jobs and pallets, given by the terms $\alpha_i, \beta_i, \gamma_i, \delta_i\}$, may vary and are defined in each problem instance. Note that loading times are not applied for jobs that are delivered on the first route by each vehicle.

Each problem instance is generated using a specific *vehicle fleet* and a specific *customer data set*. The procedure is described in the following sections.

4.1 Vehicle Fleet

Two parameters control the fleet of vehicles used—the size of the fleet vf_s and the vehicle class. The vehicle class describes a set of vehicles defined by the parameters $(containers, engine - type, equipment)$. Four classes of vehicle are considered:

- Vehicle class 1: Medium sized vehicle fitted with a diesel engine, no tailgate and a single ambient container. (1 possible vehicle type)
- Vehicle class 2. Medium or large vehicle with a single ambient container, no tailgate and any engine type. (6 possible combinations)
- Vehicle class 3: Medium or large vehicles of any engine type, with or without a tailgate and with either 1 ambient container or 1 refrigerated and 1 frozen container. (24 possible combinations)
- Vehicle class 4: All possible vehicle sizes, container combinations, engine types and equipment types (90 different combinations).

4.2 Customer Databases

Five sets of customers are used to generate problems. Each customer set has an associated distance and time matrix which are supplied separately to minimise file sizes. A single depot for all problem instances is used that is located in the city of

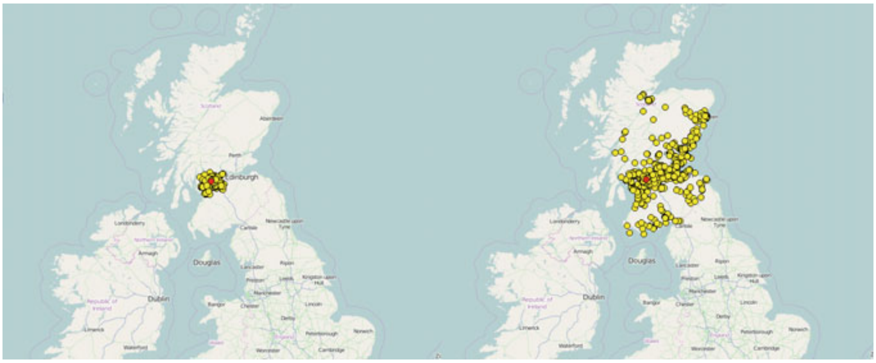


Fig. 1 Geographical distribution of customers that can be serviced within 30 and 480 min from the depot

Glasgow at a location roughly 7 KM from the city centre defined by the geographic coordinates (*latitude 55.853 longitude −4.309*).

A class of problems is generated by considering customers that can be serviced within a service-time rt_{max} minutes from the depot. We generate five classes of problems for $rt_{max} \in \{30, 60, 120, 240, 480\}$. Within each class, 1000 customers locations are randomly selected from the subset of postcodes that lie within rt_{max}; this ensures a realistic distribution of customer locations based on the actual distribution of residential and retail dwellings in the geographic area surrounding the depot.

Figure 1 shows the distribution of customers from 2 of the 5 customer data sets generated. The depot is highlighted in red.

Distances (km) and travel times (rounded to the nearest minute) between all pairs of locations are then calculated and stored in matrices m_d, m_t respectively. These are derived using the Graphhopper library as described in Sect. 3.2.4. The library makes use of *Dijkstra*'s algorithm to determine the shortest path between two locations, using a road network modelled as a directed graph. The time taken to travel between locations is correlated with the road type used with shorter travel times returned for higher capacity roads. A minimum drivin2 min 2 min is used where the software returns a shorter duration. The resulting data sets require *cleaning* due to inaccuracies present in the open source software which on occasion returns erroneous distance or time measurements. The resulting data sets are consequently reduced in size from 1000 to between 524 and 734 customers.

4.3 Generating Problem Instances

For each of the five customer sets, 10 problem instances are generated using all 96 combinations of the following parameters, resulting in 4800 problem instances in total:

- *Minimum round trip*: $rt_{min} \in \{0, rt_{max}/2\}$, where rt_{max} is the maximum round trip time used to generate the corresponding customer data set.
- *Vehicle Class* Four distinct classes (described in Sect. 4.1).
- *Vehicle fleet size* $|V| \in \{4, 8, 16, 32\}$
- *Time Window* $\in$ 60, 120, 240 min

For each instance, a fleet type is defined, and then v_{fs} vehicle types are randomly selected from the class. Having defined the customer set and a fleet, problem instances are generated by assigning route(s) to each vehicle using the process described by Algorithm 1.

Algorithm 1 Problem Generation

```
for all $v \in V$ do
    Time Vehicle Utilised = 0
    repeat
        Add a Route for the current Vehicle
        repeat
            Select a customer at random that can be serviced within the remaining time (allowing time to return to the depot and allowing for any required driver breaks)
            Create a job for the customer with a random number of pallets between the limits (or the available capacity) of type (temperature) that corresponds to a randomly selected container with available capacity available to the vehicle
            Assign the time window that corresponds with the time that the vehicle will arrive at that location
        until Vehicle Capacity Reached OR Time Vehicle Utilised $\geq$ 450
    until Time Vehicle Utilised $\geq$ 450
end for
```

5 Benchmark Datasets

5.1 XML Description

The problems are released as XML files that reflect the model described in Sect. 3. The XML files comply to the Class Diagrams depicted in Figs. 2 and 3. In order to reduce file sizes, each problem instance is supplied without the required matrices that specify travel times and distances between all pairs of locations. The matrices associated with each company data set are supplied as *CompanyData* objects, also in XML format. Each of the 5 files is formatted as described by Fig. 4 and contain the time and distance matrices that are required by a problem instance.

All problem instances have a unique integer ID ranging from 1,000,000 to 1,004,799. For each problem instance all jobs, vehicles, containers and pallets have a unique ID (ranging from $1, \ldots, n$). Customers can be identified by their unique ID or their location which remain the same where a customer appears in more than 1

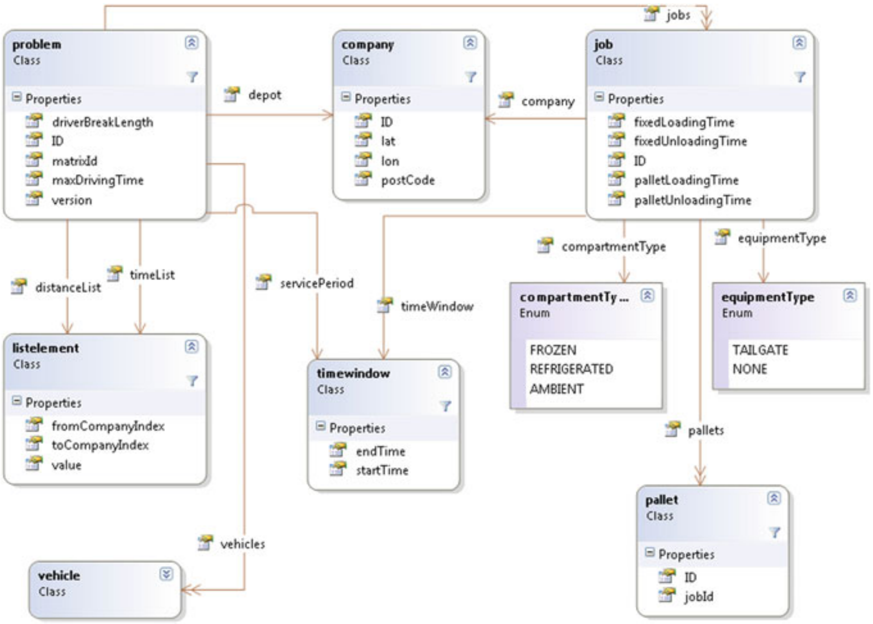


Fig. 2 The problem model

of the data sets. A customer may require to have more than one job delivered for a single problem instance and therefore multiple jobs may exist where the associated customers have the same ID.

Problems are stored on the distribution site in a hierarchical directory structure that corresponds to the parameters that the problems were generated from. Each set of 10 problem instances that were generated using the same parameter combination are zipped in their own low level directory. The corresponding matrices can be identified using the $matrixID$ given in each problem instance and can be located in the corresponding high level folder. The directory structure consists of 5 nested layers listed below from highest to lowest.

- **Company Data Set A**: $A \in \{1 \dots 5\}$ corresponding to $rt_{max} \in \{30, 60, 120, 240, 480\}$.
- **Minimum Trip Time B**: $B \in 0, 1$ corresponding to $rt_{min} \in \{0, rt_{max}/2\}$minimum round trip time
- **Vehicle Set Type C**: $C \in \{1 \dots 4\}$
- **Fleet Size D**: $D \in \{4, 8, 16, 32\}$
- **Time Window E**: $E \in \{60, 120, 240\}$

Within each of the lowest level folders is a single zip file containing the 10 problem instances generated using the same parameter settings. The associated distance and time matrices are accessible from the zipped XML document to be found in the corresponding high level folder.

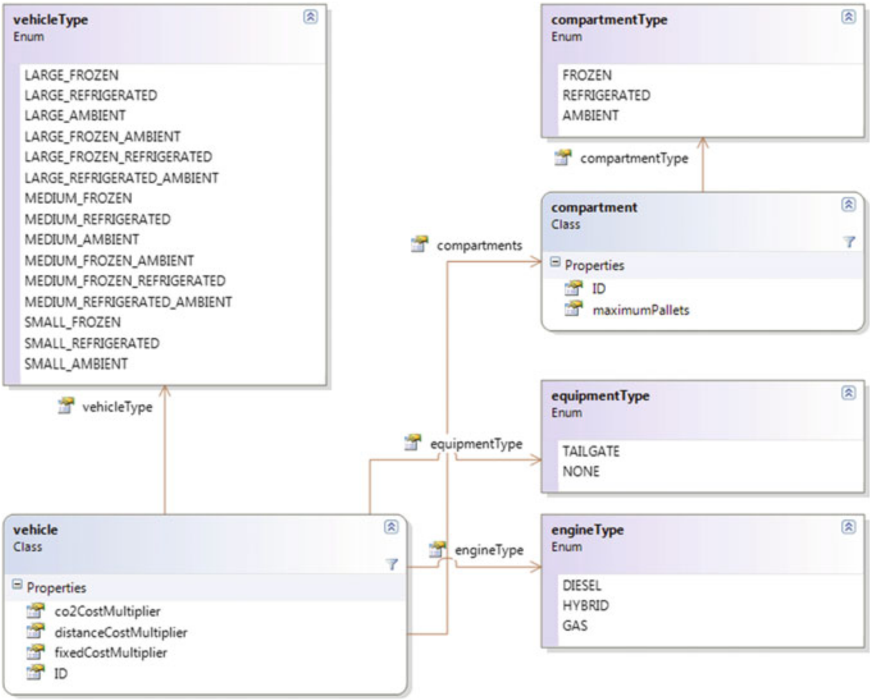


Fig. 3 The vehicle model

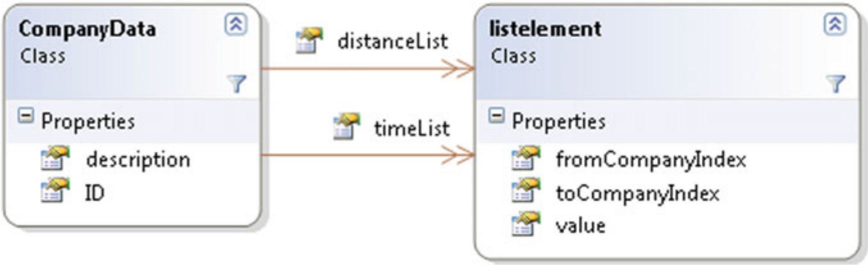


Fig. 4 The company data model

5.2 Parsing Problems Using the Supplied Java Parser

The distribution web site includes an XML parser, supplied as a jar file, that includes the problem model and a XMLParser Class containing a single method with the signature *getProblem(String instance, String companyDataSet)*. The method takes two strings as parameters representing the file names of a problem instance and the corresponding CompanyData object and returns a Problem object with the associated distance and time matrices added.

5.3 Other Formats or Programming Languages

For users wishing to use platforms other than Java, two XML schema definition files are supplied that allow the Problem and CompanyData structures to be recreated. The matrices required for a problem instance can be identified by the matrixId field of the Problem Class and by the ID field of the CompanyData Class.

6 Conclusion

It is clear that if research within vehicle routing is going to address the concerns of the industry, then large-scale instances and variants of the VRP that meet the definiton of *rich* problems are urgently required. We have proposed a new rich VRP model that incorporates load-splitting and multi-trip scenarios, combined with physical constraints that include heterogeneous fleets, multi-compartments, equipment and compartment constraints, time-window and multiple objectives. The model utilises time and distance data from a model of a real road-network, in which times are modified according to road capacity. The model is described using an XML format that can be read by any parser. 4800 instances are supplied, providing the biggest resource of rich VRP problems available to date. The framework can easily be extended to include further constraints to capture new characteristics of problems emerging in rapidly changing industry. From an academic perspective, the resource provides a means of rigorously evaluating new algorithms to assess performance across a diverse range of problems. This provides an essential mechanism for highlighting strengths and weaknesses of algorithms in relation to problem characteristics, rather than assuming a 'one-size-fits-all' approach, that will drive new innovation in the field.

References

1. Dantzig, G.B., Ramser, J.H.: The truck dispatching problem. Manag. Sci. **6**(1), 80–91 (1959)
2. Laporte, G.: Fifty years of vehicle routing. Transp. Sci. **4**, 408–416 (2009)
3. Caceres-Cruz, J., Arias, P., Guimarans, D., Riera, D., Juan, A.A.: Rich vehicle routing problem: survey. ACM Comput. Surv. **47**(2), 32:1–32:28 (2014)
4. Lahyani, R., Khemakhem, M., Semet, F.: Rich vehicle routing problems: from a taxonomy to a definition. Eur. J. Oper. Res. **241**(1), 1–14 (2015)
5. Real world optimisation with life-long learning. http://www.rollproject.org/rvrp
6. Mendoza, J., Gueret, C., Hoskins, M., Lobit, H., Pillac, V., Vidal, T., Vigo, D.: Vrp-rep: the vehicle routing community repository. In: Third meeting of the EURO Working Group on Vehicle Routing and Logistics Optimization (VeRoLog), Oslo, Norway (2014)

7. Mendoza, J., Rousseau, L.-M., Villegas, J.: A hybrid metaheuristic for the vehicle routing problem with stochastic demand and duration constraints. J. Heuristics **1**, 1–28 (2015)
8. NEO: Vrp instances. http://neo.lcc.uma.es/vrp/vrp-instances/
9. Goeke, D., Schneider, M.: Routing a mixed fleet of electric and conventional vehicles. Eur. J. Oper. Res. **245**(1), 81–99 (2015)
10. GraphHopper. https://graphhopper.com
11. Road haulage association. http://www.rha.uk.net
12. Carbon earth. http://www.carbonearth.co.uk/carbon-offset.asp

Genetic Algorithm Applied to Design Knowledge Discovery of Launch Vehicle Using Clustered Hybrid Rocket

Masahiro Kanazaki, Kazuhisa Chiba, Shoma Ito, Masashi Nakamiya, Koki Kitagawa and Toru Shimada

Abstract The conceptual design of a multi-stage launch vehicle (LV) using a clustered hybrid rocket engine (HRE) is carried out through multi-disciplinary design optimization. This LV designed in this study can deliver micro-satellites to sunsynchronous orbits (SSO). The optimum size of each component, such as an oxidizer tank containing liquid oxidizer, a combustion chamber containing solid fuel, a pressurizing tank, and a nozzle, should be strictly optimized because of the combustion mechanism is different from existent liquid/solid rocket engines. In this study, the semi-empirical based evaluation is applied to the design optimization of the multi-stage LV. For clustered HRE, paraffin (FT-0070) is used as a propellant for the HRE, and three cases are compared to examine the commonization effect of the engine for each stage: In the first case, HREs are optimized for each stage. In the second case, HREs are optimized together for the first and second stages but separately for

M. Kanazaki (✉) · S. Ito
Tokyo Metropolitan University, 6-6 Asahigaoka, Hino, Tokyo, Japan
e-mail: kana@tmu.ac.jp

S. Ito
e-mail: sequentia_11@yahoo.co.jp

K. Chiba
The University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo, Japan
e-mail: kazchiba@uec.ac.jp

M. Nakamiya
Technical University Darmstadt, Dolivostr. 15, 64293 Darmstadt, Germany
e-mail: nakamiya@dyn.tudarmstadt.de

K. Kitagawa · T. Shimada
Japan Aerospace Exploration Agency, 3-1-1 Yoshinodai, Chuo-ku, Sagamihara,
Kanagawa, Japan
e-mail: kitagawa.koki@jaxa.jp

T. Shimada
e-mail: shimada.toru@jaxa.jp

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_31

the third stage. In the third case, HREs are optimized together for each stage. The optimization results show that the performance of the design case that uses the same HREs in all stages is 40% reduced compared with the design case that uses optimized HREs for each stage.

1 Introduction

The hybrid rocket (HR) is a type of rocket that uses a liquid oxidizer and solid fuel. This rocket has the advantages of being highly safe, inexpensive, and environment-friendly in comparison with liquid and solid fuel rocket engines. Therefore, the HR is expected to be a safe and green means of propulsion for future space transportation.

The thrust of the HR is influenced by the mass flow of the vaporized fuel, which in turn is determined by the oxidizer mass flow, the fuel grain length, and the inner radius of the fuel grain port. If these parameters are combined optimally, the thrust can be expected to be sufficient. Since these parameters also constitute the engine geometry, they also affect the weight and trajectory. As a result, knowledge discovery techniques are desirable for the multi-disciplinary design of an HRE for a launch vehicle (LV). A previous study [4, 5] developed a multi-disciplinary optimization (MDO) methodology that includes a technique for an empirical-model-based evaluation of the performance of a three-stage LV with an HRE. Several solutions have achieved good performance suitable for space transportation. However, performance in terms of parameters such as the maximum payload has been limited thus far because only one HRE is installed for each stage.

In this study, a developed evaluation method was applied to the design of using clustered HRE as shown in Fig. 1a. A cluster rocket is expected to incur a low

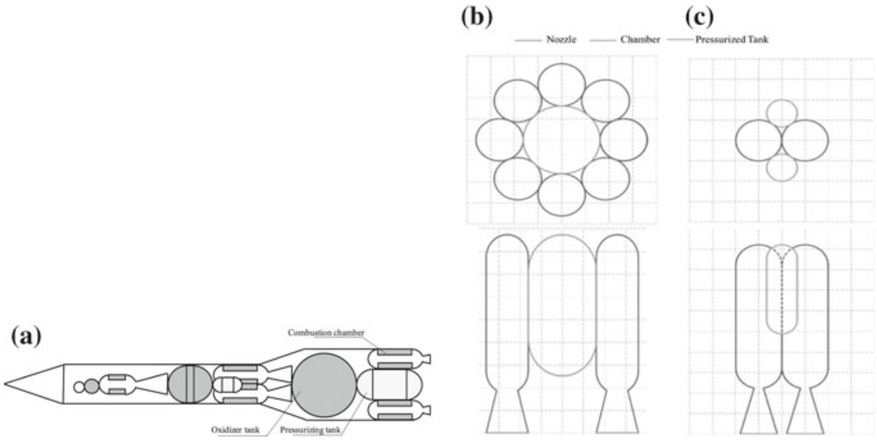


Fig. 1 Schematic illustration of an LV with clustered HREs. **a** LV with a clustered HRE, **b** cross-sectional view of the first stage, and **c** cross-sectional view of the second stage

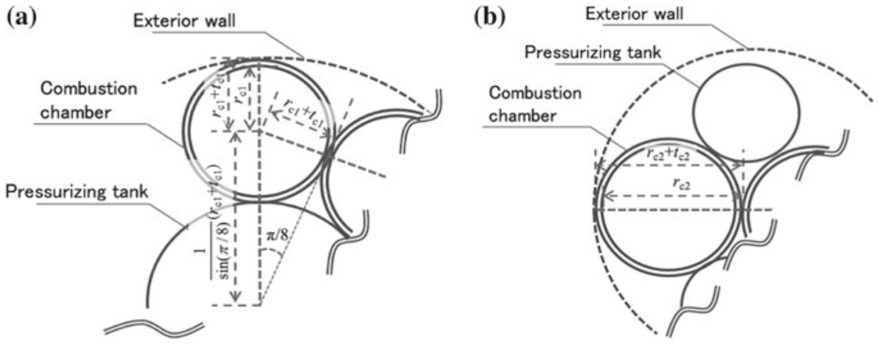


Fig. 2 Geometrical expressions of the radius for the clustered HRE. **a** First stage with eight combustion chambers and **b** second stage with two combustion chambers

development cost if the same engine designs can be used for all stages. The LV considered in this paper can deliver 10.0–100.0 kg micro-satellites, which can be used for the scientific observation of the Earth, to a sun-synchronous orbit (SSO) that is 800 km altitudes. To maximize the volume efficiency, eight combustion chambers, an oxidizer tank, and a pressurizing tank are installed in the first stage. In the second stage, two combustion chambers, an oxidizer tank, and a pressurizing tank are installed. Figures 1b, c and 2a, b shows the cross-sectional view of the first and second stages designed in this study. Paraffin fuel (FT-0070) [3] is employed as a fuel, liquid oxygen (LOX) is employed as an oxidizer, and the combustion type is a swirling-oxidizer-type engine.

To investigate this possibility, a multi-objective genetic algorithm (MOGA) is applied to the three design cases for the combination of optimum engines in each stage. In the first case, HREs are optimized for each stage. In the second case, HREs are optimized together for the first and second stages with taking the oxidizer and the fuel mass ratio and the oxidizer mass flux into the fuel of the first and the second stages are the same values. HRE for the third stage separately designed. In the third case, HREs are optimized together for each stage with assuming the oxidizer and the fuel mass ratio and the oxidizer mass flux into the fuel of each stage are the same values. Exploration results are visualized by means of Self-Organizing Map (SOM) to understand the relationship among objective functions and design variables.

2 Design Methods

2.1 Evaluation of Clustered Hybrid Rocket

This study discusses an LV of a three-stage rocket with a chamber, an oxidizer tank, a pressurizing tank, a nozzle, and a payload (Fig. 1a), that has an HR. Figure 3 shows

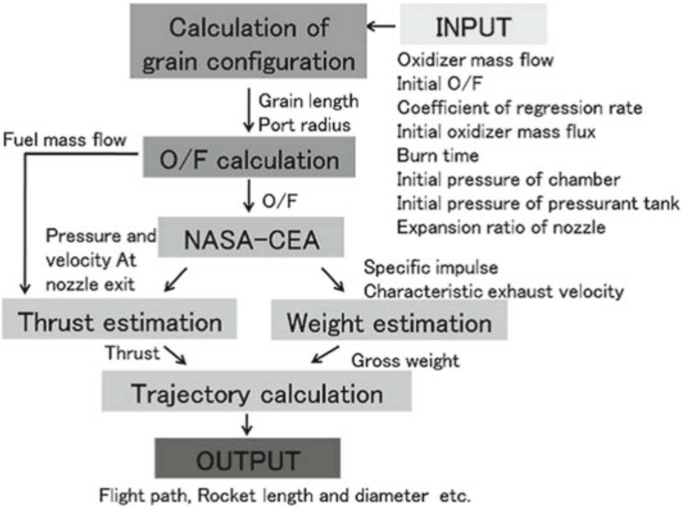

Calculation of grain configuration
INPUT
Oxidizer mass flow
Initial O/F
Coefficient of regression rate
Initial oxidizer mass flux
Burn time
Initial pressure of chamber
Initial pressure of pressurant tank
Expansion ratio of nozzle
Grain length
Port radius
Fuel mass flow
O/F calculation
O/F
NASA-CEA
Pressure and velocity At nozzle exit
Specific impulse
Characteristic exhaust velocity
Thrust estimation
Weight estimation
Thrust
Gross weight
Trajectory calculation
OUTPUT
Flight path, Rocket length and diameter etc.

Fig. 3 Evaluation procedure of HRE

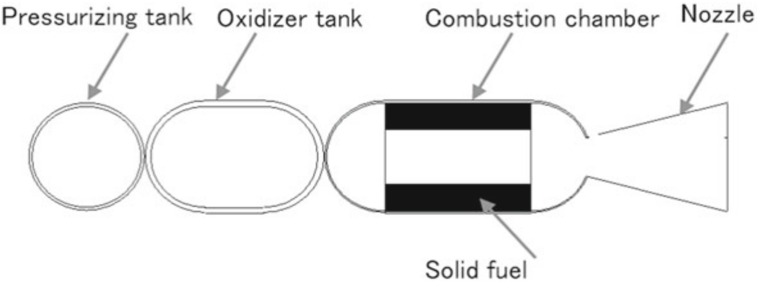


Fig. 4 HRE as a unit engine

the evaluation procedure for the unit engine shown in Fig. 4, as proposed in a previous study [4, 5].

Considering a single-port fuel grain, the regression rate $\dot{r}_{\mathrm{port}}$ of the fuel is expressed as follows:

$$\dot{r}_{\mathrm{port}}(t) = a \times G^{n}_{\mathrm{oxi}}(t). \tag{1}$$

Equation 1 is empirically defined and the coefficient G^{n}_{oxi} is the mass flux of the oxidizer which go through the port of the fuel. a and index n are generally determined through experiments for fuels with a single port. The HRE considered herein supplies the swirling oxidizer into the Paraffin fuel (FT-0070) [3]. a and n in Eq. (1) are determined from an experiment for a non-swirling oxidizer with the FT-0070; then, Eq. 1 can be written as

$$\dot{r}_{\mathrm{port}}(t) = 0.1561 \times 10^{-3} \times G^{0.3905}_{\mathrm{oxi}}(t). \tag{2}$$

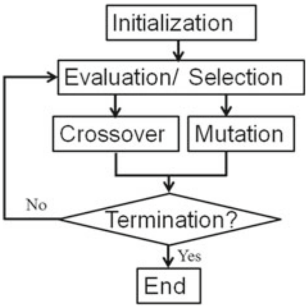


Fig. 5 Flowchart of MOGA

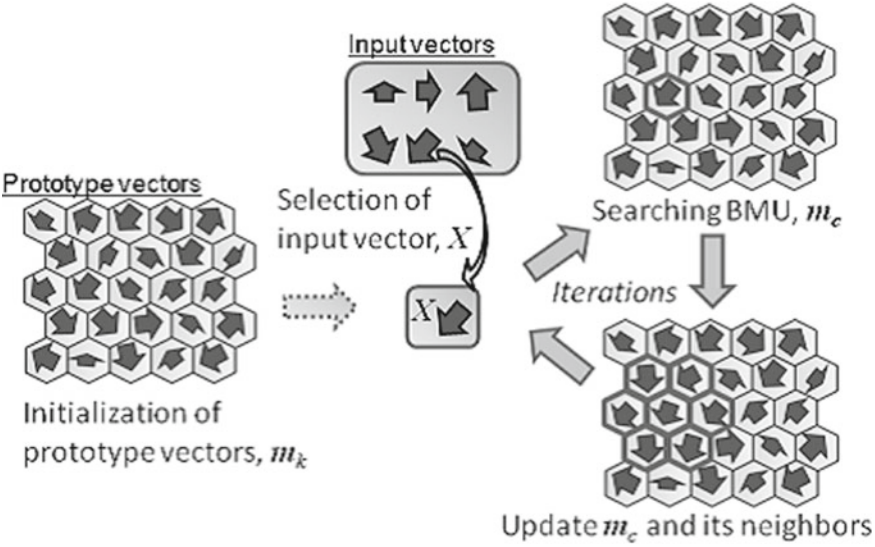

Input vectors
Prototype vectors
Selection of input vector, X
X
Initialization of prototype vectors, m_k
Searching BMU, m_c
Iterations
Update m_c and its neighbors

Fig. 6 Schematic illustration of SOM's learning

In this study, a swirling-oxidizer-type HRE that can achieve a higher regression rate [7, 8] is assumed. For this purpose, the empirical multiplication with the coefficient of Eq. 2 is carried out.

$$\dot{r}_{\text{port}}(t) = a_m \times G_{\text{oxi}}^{0.3905}(t), \tag{3}$$

where m is the stage number. This coefficient $a_m = \alpha_m \times 0.1561$ is a part of the design variables that determine the strength of the oxidizer swirl. The range of α_m, which is used to determine the design range of a_m, is between four and ten.

Equation 3 is used to determine the pressure of the combustion chamber, the oxidizer tank, and the pressurizing tank. Consequently, the volume and mass of these tanks can be estimated. After the size of the engine and LV are estimated, the flight of the LV can be calculated by solving the equation of motion.

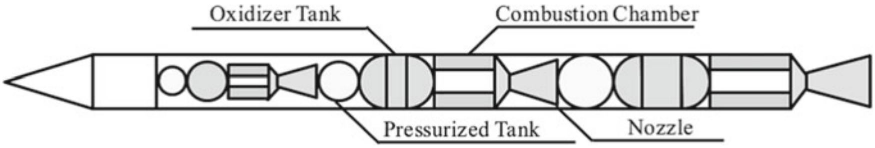


Fig. 7 Schematic illustration of non-clustered LV for Case 0

2.1.1 Layout of Components

The LV considered in this study has eight clustered combustion chambers with a nozzle in the first stage and two clustered combustion chambers with a nozzle in the second stage (Fig. 1b, c). The third stage contains a single combustion chamber with a nozzle. The first stage contains a pressurizing tank at the center of the combustion chambers, the second stage contains two pressurizing tanks at the side of the combustion chamber, and the third stage contains a pressurizing tank on the oxidizer tank. All stages have an oxidizer tank as the common component.

To maximize volume efficiency, the combustion chamber, oxidizer tank, and pressurizing tank for each stage are placed as shown in Fig. 1. The radius of the first stage r_{clu1} (Fig. 2a) can be expressed as

$$r_{\mathrm{clu1}} = \left(\frac{1}{\sin(\pi/8)}\right) \cdot (r_{\mathrm{c1}} + t_{\mathrm{c1}}), \tag{4}$$

where r_{c1} is the radius of the combustion chamber and t_{c1} is the thickness of the exterior wall. r_{c1} can be obtained by the combustion time tc_1 based on Eq. 3, and tc_1 is determined by the initial pressure of the combustion chamber $P_{\mathrm{ch\_1}}(0)$.

The radius of the second stage r_{clu2} (Fig. 2b) can be expressed as

$$r_{\mathrm{clu2}} = 2(r_{\mathrm{c2}} + t_{\mathrm{c2}}), \tag{5}$$

where r_{c2} is the radius of the combustion chamber and t_{c2} is the thickness of the exterior wall. r_{c2} can also be obtained by the combustion time tc_2 based on Eq. 3, and t_{t2} is decided by the initial pressure of the combustion chamber $P_{\mathrm{ch\_2}}(0)$.

3 Design Optimization

3.1 Non-dominated Sorting Genetic Algorithm-II (NSGA-II)

Genetic algorithms (GAs) are popular holistic optimization techniques that use operators such as selection, crossover, and mutation, as shown in Fig. 5. The non-dominated sorting genetic algorithm-II (NSGA-II) [1]. NSGA-II is characterized by non-dominated sorting and crowding distance sorting. The individuals of the

Table 1 Design variables for Case0 and Case1

	Unit	Variable
dv1	kg/s	$10.0 \leq \dot{m}_{\text{oxi\_1}} \leq 150.0$
dv2	–	$2.0 \leq O/F_{\_1}(0) \leq 3.0$
dv3	m	$2.5 \leq a_{\_1} \leq 7.0$
dv4	$\text{kg/m}^2\text{s}$	$150.0 \leq G_{\text{oxi\_1}} \leq 400.0$
dv5	s	$30.0 \leq tc_{\_1} \leq 60.0$
dv6	MPa	$0.5 \leq P_{\text{ch\_1}}(0) \leq 3.0$
dv7	MPa	$15.0 \leq P_{\text{pt\_1}}(0) \leq 40.0$
dv8	–	$2.0 \leq \varepsilon_{\_1} \leq 8.0$
dv9	kg/s	$3.0 \leq \dot{m}_{\text{oxi\_2}} \leq 35.0$
dv10	–	$2.0 \leq O/F_{\_2}(0) \leq 3.0$
dv11	m	$4.0 \leq a_{\_2} \leq 10.0$
dv12	$\text{kg/m}^2\text{s}$	$10.0 \leq G_{\text{oxi\_2}} \leq 200.0$
dv13	s	$70.0 \leq tc_{\_2} \leq 130.0$
dv14	MPa	$0.5 \leq P_{\text{ch\_2}}(0) \leq 2.0$
dv15	MPa	$10.0 \leq P_{\text{pt\_2}}(0) \leq 30.0$
dv16	–	$10.0 \leq \varepsilon_{\_2} \leq 50.0$
dv17	kg/s	$0.3 \leq \dot{m}_{\text{oxi\_3}} \leq 5.0$
dv18	–	$2.0 \leq O/F_{\_3}(0) \leq 3.0$
dv19	m	$4.0 \leq a_{\_3} \leq 9.0$
dv20	$\text{kg/m}^2\text{s}$	$5.0 \leq G_{\text{oxi\_3}} \leq 120.0$
dv21	s	$90.0 \leq tc_{\_3} \leq 180.0$
dv22	MPa	$0.5 \leq P_{\text{ch\_3}}(0) \leq 1.0$
dv23	MPa	$15.0 \leq P_{\text{pt\_3}}(0) \leq 40.0$
dv24	–	$10.0 \leq \varepsilon_{\_3} \leq 70.0$
dv25	s	$100.0 \leq t_{\text{coast}} \leq 180.0$

next generation are selected by elitism. The new generation is filled with each front sequentially until the population size exceeds the current population size. In this study, the blended crossover-α (BLX-α) [2] is applied as the crossover.

3.2 *Self-Organizing Map (SOM)*

SOM [6] is a nonlinear projection algorithm from high to two dimensional map based on self-organization of a low-dimensional array of neurons. This study used the hexagonal grid because it is more pleasing to the eye.[1]

[1] In this study, commercial software modeFrontier®ver. 4. 4. 2 is used.

Table 2 Design variables for Case2

	Unit	Variable
dv1	kg/s	$10.0 \le \dot{m}_{\text{oxi\_1}} \le 150.0$
dv2	–	$2.0 \le O/F_{\_1}(0) \le 3.0$
dv3	m	$2.5 \le a_{\_1} \le 7.0$
dv4	kg/m^{2}s	$150.0 \le G_{\text{oxi\_1}} \le 400.0$
dv5	s	$30.0 \le tc_{\_1} \le 60.0$
dv6	MPa	$0.5 \le P_{\text{ch\_1}}(0) \le 3.0$
dv7	MPa	$15.0 \le P_{\text{pt\_1}}(0) \le 40.0$
dv8	–	$2.0 \le \varepsilon_{\_1} \le 8.0$
dv9	kg/s	$3.0 \le \dot{m}_{\text{oxi\_2}} \le 35.0$
dv10	–	$=$ dv2
dv11	m	$4.0 \le a_{\_2} \le 10.0$
dv12	kg/m^{2}s	$=$ dv4
dv13	s	$70.0 \le tc_{\_2} \le 130.0$
dv14	MPa	$0.5 \le P_{\text{ch\_2}}(0) \le 2.0$
dv15	MPa	$10.0 \le P_{\text{pt\_2}}(0) \le 30.0$
dv16	–	$10.0 \le \varepsilon_{\_2} \le 50.0$
dv17	kg/s	$0.3 \le \dot{m}_{\text{oxi\_3}} \le 5.0$
dv18	–	$2.0 \le O/F_{\_3}(0) \le 3.0$
dv19	m	$4.0 \le a_{\_3} \le 9.0$
dv20	kg/m^{2}s	$5.0 \le G_{\text{oxi\_3}} \le 120.0$
dv21	s	$90.0 \le tc_{\_3} \le 180.0$
dv22	MPa	$0.5 \le P_{\text{ch\_3}}(0) \le 1.0$
dv23	MPa	$15.0 \le P_{\text{pt\_3}}(0) \le 40.0$
dv24	–	$10.0 \le \varepsilon_{\_3} \le 70.0$
dv25	s	$100.0 \le t_{\text{coast}} \le 180.0$

Each neuron k on the map is represented by an n-dimensional prototype vector $m_k = (m_{k1}, m_{k2}, \ldots, m_{kn})$, where n is the dimension of the design space. To train the map, input vector X which represents a sampling design is selected and the nearest neuron m_c (the best matching unit, BMU) is found from the prototype vectors on the map. The prototype vectors of the mc and its neighbors on the grid m_k are moved towards X as follows.

$$m_k = m_k + \alpha(t)(X - m_k), \tag{6}$$

where, $\alpha(t)$ is learning rate and it decreases monotonically with time. This process as shown in Fig. 6 is iterated until $\alpha(t)$ is converged well. During the iterative training, prototype vectors are also converged. The closer two patterns are in the original space, the closer is the response of two neighboring neurons in the low-dimensional map. Thus, SOM reduces the dimension of input data while preserving their features.

Table 3 Design variables for Case0 and Case1

	Unit	Variable
dv1	kg/s	$10.0 \le \dot{m}_{\mathrm{oxi\_1}} \le 150.0$
dv2	–	$2.0 \le O/F_{\_1}(0) \le 3.0$
dv3	m	$2.5 \le a_{\_1} \le 7.0$
dv4	kg/m^{2}s	$150.0 \le G_{\mathrm{oxi\_1}} \le 400.0$
dv5	s	$30.0 \le tc_{\_1} \le 60.0$
dv6	MPa	$0.5 \le P_{\mathrm{ch\_1}}(0) \le 3.0$
dv7	MPa	$15.0 \le P_{\mathrm{pt\_1}}(0) \le 40.0$
dv8	–	$2.0 \le \varepsilon_{\_1} \le 8.0$
dv9	kg/s	$3.0 \le \dot{m}_{\mathrm{oxi\_2}} \le 35.0$
dv10	–	$=$ dv2
dv11	m	$4.0 \le a_{\_2} \le 10.0$
dv12	kg/m^{2}s	$=$ dv4
dv13	s	$70.0 \le tc_{\_2} \le 130.0$
dv14	MPa	$0.5 \le P_{\mathrm{ch\_2}}(0) \le 2.0$
dv15	MPa	$10.0 \le P_{\mathrm{pt\_2}}(0) \le 30.0$
dv16	–	$10.0 \le \varepsilon_{\_2} \le 50.0$
dv17	kg/s	$0.3 \le \dot{m}_{\mathrm{oxi\_3}} \le 5.0$
dv18	–	$=$ dv2
dv19	m	$4.0 \le a_{\_3} \le 9.0$
dv20	kg/m^{2}s $=$ dv4	
dv21	s	$90.0 \le tc_{\_3} \le 180.0$
dv22	MPa	$0.5 \le P_{\mathrm{ch\_3}}(0) \le 1.0$
dv23	MPa	$15.0 \le P_{\mathrm{pt\_3}}(0) \le 40.0$
dv24	–	$10.0 \le \varepsilon_{\_3} \le 70.0$
dv25	s	$100.0 \le t_{\mathrm{coast}} \le 180.0$

The trained SOM is systematically converted into visual information, and qualitative information can be obtained.

4 Formulation

In this study, an LV that can deliver a payload to an Earth orbit at 800 km is considered. The rocket is launched toward the south at an 89°. launch angle. The second stage is immediately ignited after the combustion of the first stage is completed. After the second stage is completed, the third stage coasts along the oval orbit. Once the coasting is completed, the third stage is ignited. t_{coast}. Three cases are compared: In Case 1, the HREs are optimized separately for each stage. In Case 2, the first and

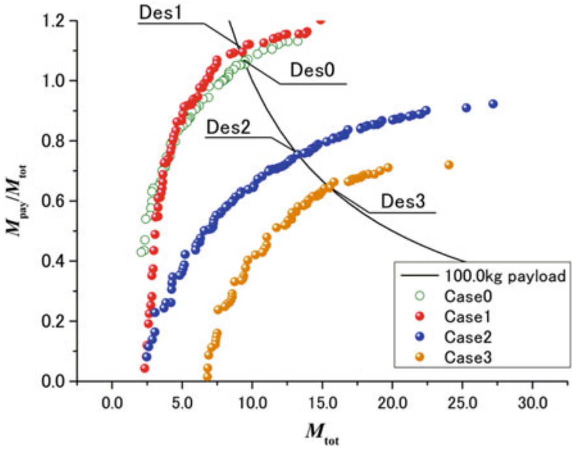


Fig. 8 Comparison of non-dominated solutions

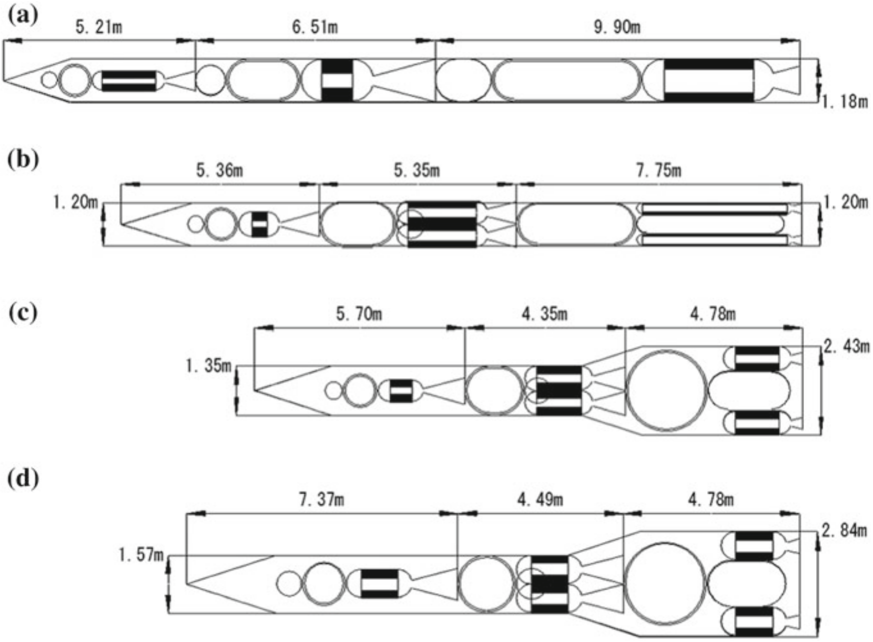


Fig. 9 Comparison of designs that can deliver 100.0 kg payloads from non-dominated solutions: **a** Des0, **b** Des1, **c** Des2, and **d** Des3

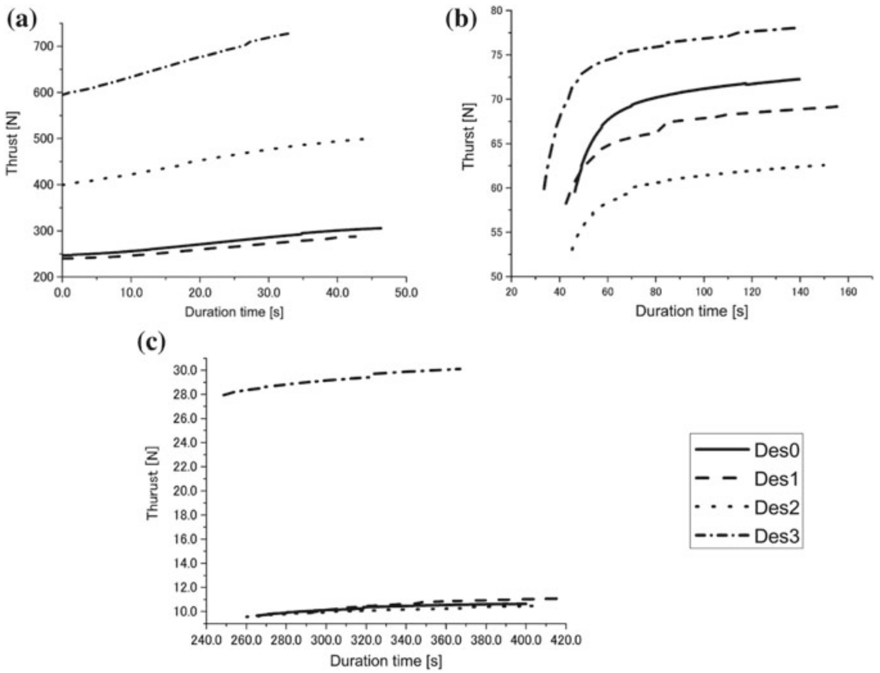


Fig. 10 Comparison of time history from Des0 through Des3: **a** first stage, **b** second stage, and **c** third stage

second stages share an engine design, while the third stage is optimized separately. In Case 3, all states share the same engine design. The non-dominated solutions for all three cases are compared with the non-dominated solutions for the design with a non-clustered HRE (Case 0) as shown in Fig. 7 [4].

The objective functions are to maximize the payload to total weight ratio $M_{\text{pey}}/M_{\text{tot}}$ and minimize the total weight (M_{tot}). The design problem for each case can be expressed as

$$\begin{cases} \text{Maximize} \quad Alt_{\max} \\ \text{Minimize} \quad M_{\text{tot}} \end{cases} \tag{7}$$

The trajectory constraints assumed are as follows:

- The flight altitude is over 250 km after the combustion of the third stage.
- The angular momentum is more than 52413.5 kg km^2/s after the combustion of the third stage in order to ensure that the rocket reaches 800 km at the apogee.
- The flight path angle after combustion of the third stage is between -5.0 and +5.0°.

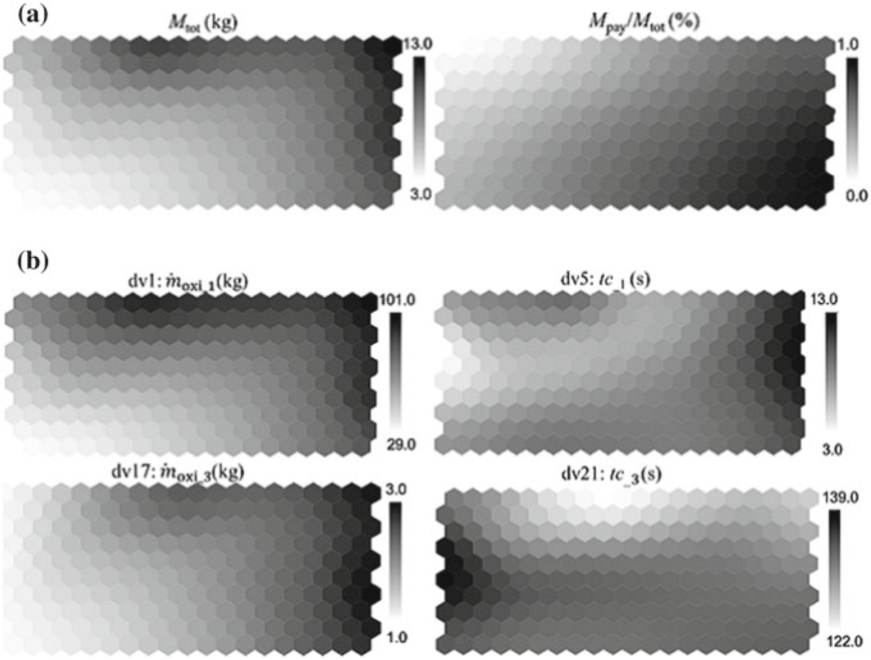


Fig. 11 Component maps by SOM for Case0. **a** SOM colored by objective functions and **b** Maps colored by selected design variables

The constraints for the structure are as follows:

- The aspect ratio of the rocket is less than 20.0.
- The diameter of the nozzle exit is less than that of all stages.
- The area of the grain port is more than twice the nozzle throat area of all stages.

The design variables and their ranges are listed in Tables 1, 2 and 3. dv1-dv8, dv9-dv16, dv17-dv26 correspond to the first, second, third stages, respectively. (dv26 is t_{coast}.) $\dot{m}_{\text{oxi\_m}}$ is the mass flow of the oxidizer, $O/F_{\_1}(0)$ is the oxidizer to fuel ratio, $a_{\_m}$ is the coefficient for Eq. 3, $G_{\text{oxi\_m}}$ is the mass flux of the oxidizer through the fuel port, $tc_{\_m}$ is the combustion time, $P_{\text{ch\_m}}(0)$ and $P_{\text{pt\_m}}(0)$ are respectively the initial pressure of the combustion chamber and the pressurizing tank, $\varepsilon_{\_m}$ is the appature ratio of the nozzle in the mth stage. In Case2, dv10 and dv12 are equivalent to dv2 and dv4, respectively. In Case3, dv19 and dv21 are also equal to dv2 and dv4, respectively.

NSGA-II for each case is carried out with a total generation number of 200 and population size of 50 for each generation.

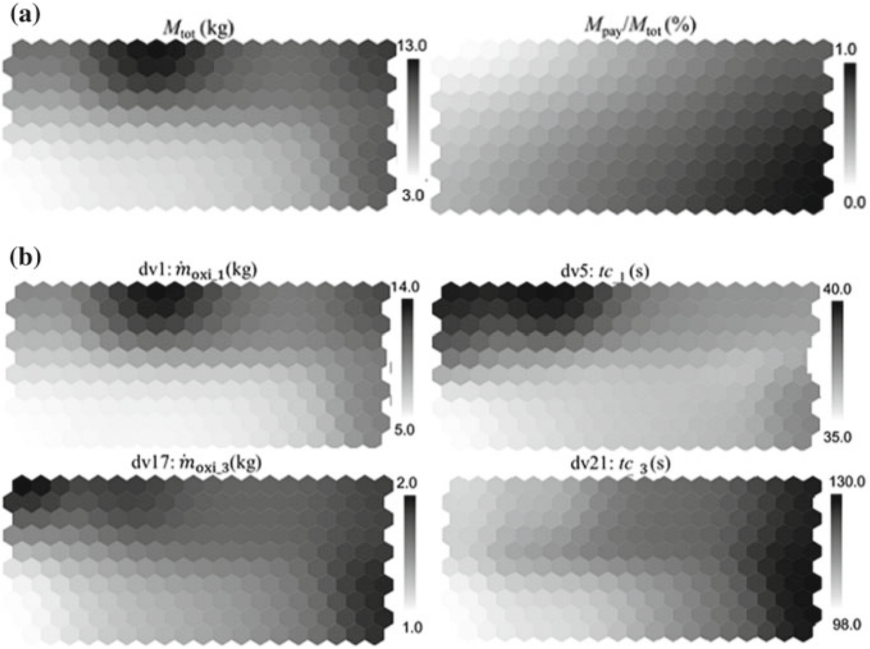


Fig. 12 Component maps by SOM for Case1. **a** SOM colored by objective functions and **b** Maps colored by selected design variables

5 Results

5.1 Design Exploration Results

Figure 8 shows a comparison of the non-dominated solutions for each case. The result for the single-engine rocket is derived from a previous study [4]. Figure 8 helps establish the trade-off between the objective functions. Case 1 can reach a greater value of $M_{\text{pay}}/M_{\text{tot}}$ at a lower M_{tot} than either Case 2 or 3 because all the stages have an optimized engine. The solid line in Fig. 8 is the line that can deliver 100.0 kg payloads. According to Fig. 8, a design that can deliver 100.0 kg payloads could be found in all three cases.

5.2 Comparison of Designs from Non-dominated Solutions

Rocket designs that can deliver payloads of 100.0 kg found from where the design curves intersect the solid line in Fig. 8 are picked up and named as Des0, Des1, Des2

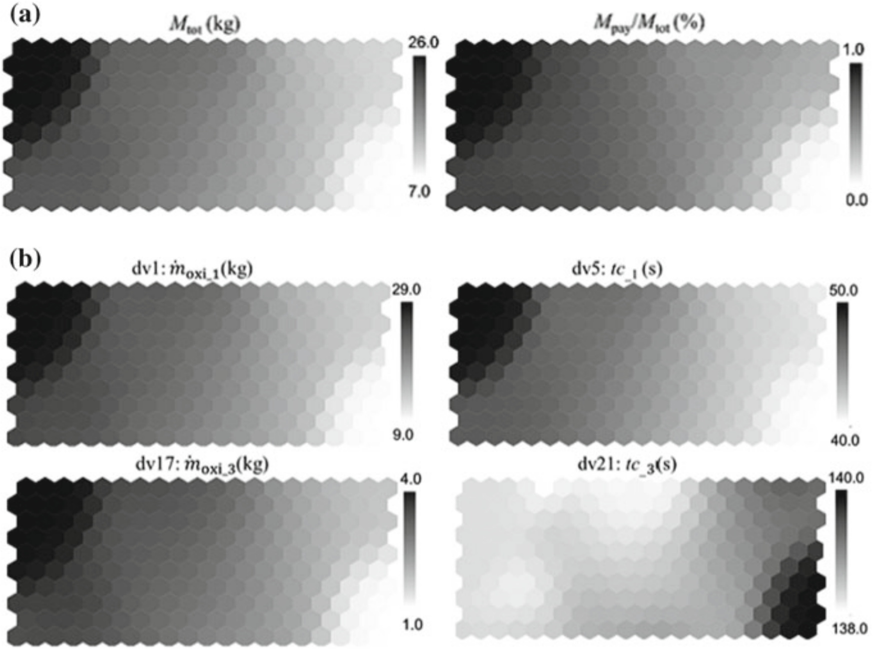


Fig. 13 Component maps by SOM for Case2. **a** SOM colored by objective functions and **b** Maps colored by selected design variables

and Des3, respectively. The value of $M_{\text{pay}}/M_{\text{tot}}$ for Des3 is 60% of that for Des1, because the engines are not optimized for each stage in Case 3.

Figure 9 shows a visual comparison of the design results for each case and also for the single-engine rocket. As can be seen from Fig. 9, the diameters in the first stage of Des2 and Des3 are larger than that in the first stage of Des1. Because the first stage is independently designed in Case 1, the resulting engine size is the same as that in Des0. However, when the engine designs are shared, the width constraint in the second stage causes the first stage to have an increased diamet.

Figure 10 shows a comparison of the time history of the thrust of each stage from Des0 through Des3. In the first and second stages (Fig. 10a, c), Des0 and Des1 showed a similar trend because the size of the first stage is similar in Fig. 9. In addition, Des3 also shows a trend similar to Des0 and Des1 because Case 2 independently designs the third stage, which is similar to Cases 0 and 1. The thrusts in the first stage of Des2 and Des3 are higher than that in the first stage of Des0 and Des1, while that in the first stage of Des3 is the highest. This suggests that the common use of an engine for several stages reduces the fuel efficiency while improving the productivity cost.

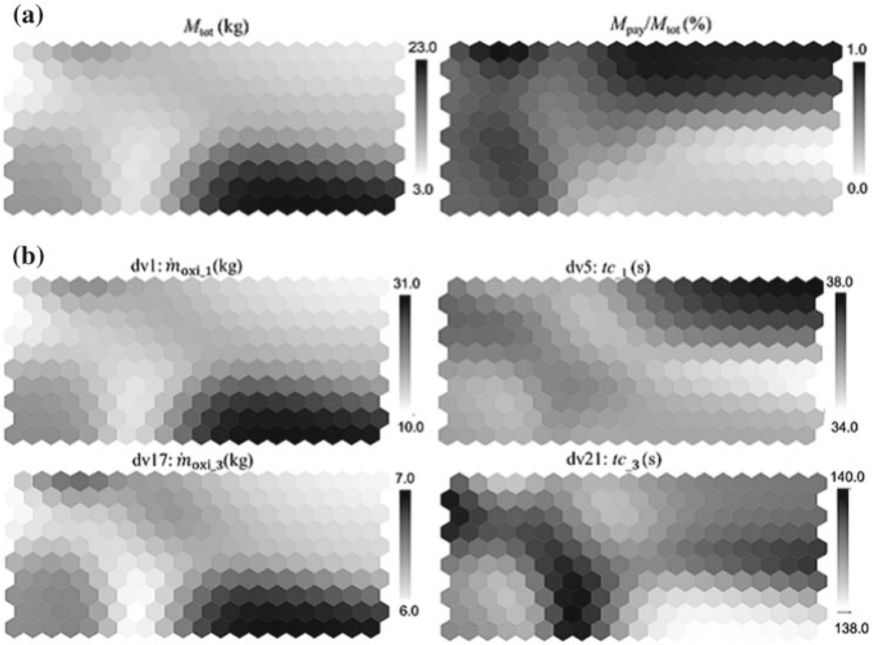


Fig. 14 Component maps by SOM for Case3. **a** SOM colored by objective functions and **b** Maps colored by selected design variables

5.3 Visualization of Design Space by SOM

To generate SOM, the objective functions are used as trained data, and the trained map is colored according to every objective functions and design variables. In this paper, dv1($\dot{m}_{oxi\_1}$), dv5(tc_1), dv17($\dot{m}_{oxi\_3}$), dv5(tc_3) which decide the engine size for the first and the third stages are selected for comparison. Figures 11, 12, 13 and 14 shows the visualization of the design problem for Cases 0 through 3, respectively.

Figure 11a shows the component planes about objective functions for Case0. According to this figure, low M_{tot} can be shown around left side of the map, while high M_{tot} can be shown around right side. This suggests that the trage-off between the objective functions is severe. Figure 11b shows the component planes about selected design variables. Comparing Fig. 11a, b, higher dv1, dv5 and dv17 achieve the higher M_{pay}/M_{tot}. On the other hand, to minimize M_{tot}, dv1 should be low because there is a high correlation between M_{tot} and dv1. (Low $\dot{m}_{oxi\_1}$ is desirable for the first stage to achieve a low weight.)

According to Fig. 12a which are the component planes about objective functions for Case1, there is the trage-off between the objective functions. Comparing Fig. 12a, b, higher dv17 and dv21 achieve the higher M_{pay}/M_{tot}. To minimize M_{tot}, dv1 should also be lowest value because there is a high correlation between M_{tot}

and dv1. However, midium value of dv1 can achieve high $M_{\mathrm{pay}}/M_{\mathrm{tot}}$. This trend is distinct from Case0 owing to the employment of the cluster engine in the first stage.

Figure 13a shows the component planes about objective functions for Case2. According to this figure, there is the severe trage-off between the objective functions because there is a high correlation between objective functions. Comparing Fig. 13a, b, higher dv1, dv5 and dv17 achieve the higher $M_{\mathrm{pay}}/M_{\mathrm{tot}}$. To minimize M_{tot}, not only dv1 but also dv5 and dv17 should also be lowest value because there is a high correlation between M_{tot} and dv1. dv5 and dv17.

Figure 14a shows the component planes about objective functions for Case3. According to this figure, there is the severe trage-off but it is not severe compared with the Case0, 1 and 2. Comparing Fig. 13a, b, dv1 and dv5 which are design variables for the first stage deveided the M_{tot}. On the other hand, dv17 and dv21 which are design variables for the first stage deveided the $M_{\mathrm{pay}}/M_{\mathrm{tot}}$. The dv5 of Case3 is lower than that of other cases when $M_{\mathrm{pay}}/M_{\mathrm{tot}}$ is higher, suggesting that the solutions of Case3 obtained earlier consumption of the fuel to reduce the fuel weight of the first stage earlier. On the other hand, the dv1 of Case 3 is lower because reduce the fuel weight of the first stage.

6 Conclusions

In this study, we considered the conceptual design of a multi-stage launch vehicle with a clustered hybrid rocket engine. Three optimization cases were considered. The results suggest that an LV that uses the same engines for each stage can only achieve 60% of the payload mass ratio, compared with an LV that has engines optimized separately for each stage. In addition, an LV that uses same engines for each stage has a lower aspect ratio than an LV with an optimized engine for each stage. These findings are significant for LV development and manufacturing.

Acknowledgements We thank members of the hybrid rocket research working group in ISAS/JAXA for giving their experimental data and their valuable advices.

References

1. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
2. Eshelman, L., Schaffer, J.: Real-Coded Genetic Algorithms and Interval-schemata. pp. 187–202 (1993)
3. Hikone, S.: Regression Rate Characteristics and Combustion Mechanism of Some Hybrid Rocket Fuels (2010)
4. Kanazaki, M., Kanamori, F., Kitagawa, Y., Nakamiya, M., Kitagawa, K., Shimada, T.: Conceptual design: dependence of parameterization on design performance of three-stage hybrid rocket. Trans. Jpn. Soc. Aeronaut. Sp. Sci. **9** (2014)

5. Kitagawa, Y., Kitagawa, K., Nakamiya, M., Kanazaki, M., Shimada, T.: Multi-stage hybrid rocket conceptual design for mi-cro-satellites launch using genetic algorithm. Trans. Jpn. Soc. Aeronaut. and Sp. Sci. **55**(4), 229–236 (2012)
6. Kohonen, T.: Self-Organizing Maps. Springer (2000)
7. Kosugi, Y., Oyama, A., Fujii, K., Kanazaki, M.: Multidisciplinary and multi-objective design exploration methodology for conceptual design of a hybrid rocket. In: AIAA Paper 2011-1634. AIAA (2011)
8. Yuasa, S., Yamamoto, K., Hachiya, H., Kitagawa, K., Owada, Y.: Development of a small sounding hybrid rocket with a swirling-oxidizer-type engine. In: 37th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit AIAA paper 01-3537 (2001)

Topology Optimization of Flow Channels with Heat Transfer Using a Genetic Algorithm Assisted by the Kriging Model

Mitsuo Yoshimura, Takashi Misaka, Koji Shimoyama and Shigeru Obayashi

Abstract A global optimization method for topology optimization using a genetic algorithm is proposed in this paper. The genetic algorithm used in this paper is assisted by the Kriging surrogate model to reduce computational cost required for function evaluation. To validate the global topology optimization method in flow problems, this research works on two single-objective optimization problems, where the objective functions are to minimize pressure loss and to maximize heat transfer of flow channels, and the multi-objective optimization problem, which combines these two problems. The shape of flow channels is represented by the level set function, and the pressure loss and the temperature of the channels are evaluated by the Building-Cube Method (BCM), which is a Cartesian-mesh CFD approach. The proposed method resulted in an agreement with previous study in the single-objective problems in its topology, and achieved global exploration of non-dominated solutions in the multi-objective problem.

1 Introduction

Shape optimization has been attracting much attention in flow problems, which defines the boundary between fluid and solid regions. However, shape optimization cannot deal with the change of topology, e.g., making new holes into an object. Topology optimization is the most flexible optimization method, which can not only modify the shape of an object but also allow the connectivity of the object to change.

Topology optimization has been applied to a variety of engineering optimization problems [1] such as structural mechanics problems, heat transfer problems, and acoustic problems since Bendsøe and Kikuchi first proposed the so-called homogenization method [2]. However, the first application to flow problems was later than the aforementioned applications. It was performed for the Stokes flow by Borrvall and Petersson [3].

M. Yoshimura (✉) · T. Misaka · K. Shimoyama · S. Obayashi
Institute of Fluid Science, Tohoku University, 2-1-1 Katahira, Aoba-Ku, Sendai 980-8577, Japan
e-mail: yoshimura@edge.ifs.tohoku.ac.jp

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_32

The basic concept of topology optimization is the replacement of the optimization problem with a material distribution problem in a fixed design domain using the characteristic function that indicates whether material exists or not. However, conventional topology optimization tends to suffer from numerical instabilities such as grayscale material and checkerboard pattern. The level set method [4, 5] is one of the approaches to avoid such instabilities. The level set method introduces a signed scalar function and distinguishes solid and fluid regions according to the sign of the function. Thus, zero-contours indicate the boundaries of the regions.

Conventional topology optimization generally explores the optimal solution by the gradient-based method according to the sensitivity of an objective function. However, the gradient-based method tends to get stuck to the local optima rather than the global optimum. On the other hand, Evolutionary Algorithm (EA) is one of the metaheuristic optimization methods, which is more capable to explore the global optimum. However, EA requires numerous function evaluations to realize population-based multipoint simultaneous exploration. Thus, EA is not efficient to solve the optimization problems with expensive calculations (e.g. Computational Fluid Dynamics (CFD)) for function evaluation if EA is employed solely. Moreover, topology optimization involves a large design space due to a high degree of freedom for shape and topology representation.

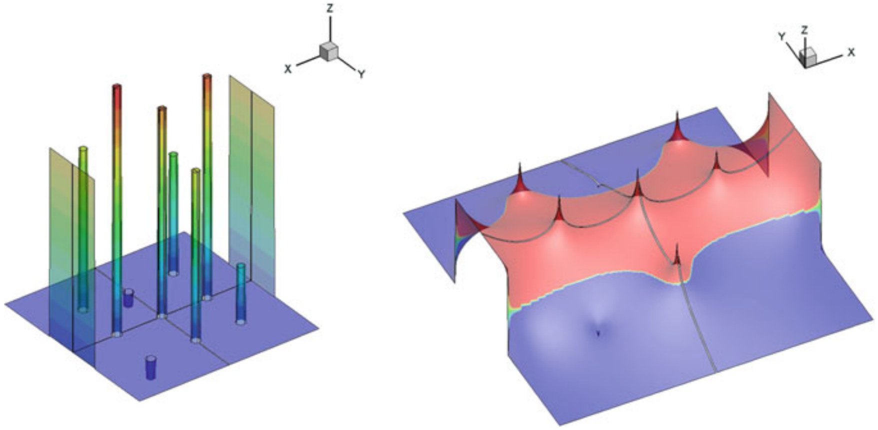
Thus, it requires much expensive computational cost (i.e., large population and many generations) to obtain competitive solutions. In this case, surrogate models are effective to reduce computational cost required for function evaluation. This model approximates the response of each objective or constraint function to design variables in an algebraic expression. This model is derived from several sample points with real values of the objective or constraint function given by expensive numerical simulations. Thus, it can promptly give estimates of function values at arbitrary design variable values.

In order to find a global optimum effectively, a global optimization method for topology optimization using a genetic algorithm assisted by the Kriging surrogate model is proposed. To validate the global topology optimization methods applied to flow problems, this research works on two single-objective optimization problems, each of which is to minimize pressure loss or to maximize heat transfer, and one multi-objective optimization problem to minimize pressure loss and to maximize heat transfer of flow channels.

2 Computational Methods

2.1 Flow Channel Representation

The boundaries between fluid and solid regions are represented by the level set representation that introduces a signed scalar function (level set function) $\phi(\boldsymbol{x})$ where $\boldsymbol{x}$ represents the location in the design domain. This research sets the range of $\phi(\boldsymbol{x})$



(a) Initial distribution

(b) 3D distribution

Fig. 1 Distribution of level set function

as $|\phi(x)| \leq 1$, and assumes $\boldsymbol{x}$ is in the fluid region if $\phi(x) > 0$ or in the solid region if $\phi(x) < 0$. The example of the distribution of the level set function is shown in Fig. 1. At the outer boundary of the design domain, $\phi(\boldsymbol{x})$ is given as the step functions corresponding to the width of the inlet and outlet of the channel. Given $\phi(\boldsymbol{x})$ at several discrete control points inside the design domain (which are treated as the current design variables and stated later), the Laplace's equation is solved in the entire domain to obtain the distribution of $\phi(\boldsymbol{x})$.

The distribution of $\phi(\boldsymbol{x})$ derived from solving the Laplace's equation is non-linear. Thus several shapes cannot be represented by any value of $\phi(\boldsymbol{x})$ given at the control points as the design variables as long as the control points are infinitesimal. For example, the channels whose boundary is exactly straight cannot be represented. This issue must be solved to compare the proposed method with previous study. This problem results from the distance between the adjacent control points where the value of $\phi(\boldsymbol{x})$ is given. Thus, in this study, the control point is considered as a circle with a certain radius, and the value of $\phi(\boldsymbol{x})$ is uniformly given in the circle. Then, the extrema in the distribution of $\phi(\boldsymbol{x})$ get closer, and the boundary can be drawn as a smoother curve that allows us to represent a straight channel. Such a treatment is applied to the optimization cases if necessary; the radius of the control point is fixed to zero (i.e., the control points are infinitesimal) or a certain value in the cases aiming at the comparison with the previous study, otherwise the radius is allowed to change as an additional design variable.

Conventional level set method [5] for topology optimization employs the Hamilton-Jacobi partial differential equation to update the value of $\phi(\boldsymbol{x})$ for a modified shape. The partial differential equation is solved with design sensitivities derived from the sensitivity analysis. In this study, on the other hand, the partial differential equation is not solved, and the value of $\phi(\boldsymbol{x})$ is updated based on the genetic algorithm stated in Sect. 2.3.

2.2 Building-Cube Method

In order to evaluate the pressure loss of the channels, CFD simulations are conducted by the Building-Cube Method (BCM) [6], which is a Cartesian-mesh CFD approach. The governing equations of BCM are the 2D incompressible Navier-Stokes equations for unsteady state flow, and the 2D energy equation for unsteady state heat transfer. The convection terms are evaluated by a third-order upwind differencing, and the viscous terms are evaluated by a second-order central differencing. Time integration is conducted by the Crank-Nicolson method for the viscous terms and the Adams-Bashforth scheme for the convective terms, and the coupling of velocity and pressure is conducted by a fractional step method.

Since BCM is a Cartesian-mesh CFD approach, it is easy to deal with the complicated shapes of flow channels with topological change. However, in the Cartesian-mesh CFD approach, the object surface is represented by a staircase pattern, instead of smooth surface. For high accuracy computation, the Immersed Boundary Method (IBM) [7] using ghost cell and image point is employed at the wall boundary.

2.3 Genetic Algorithm

The genetic algorithm mimics the evolution of organisms, which selects individuals from the current generation as parents, generates new individuals as children by the crossover and mutation of the parents, and inherits better individuals to the next generation. In this study, Non-dominated Sorting Genetic Algorithm II (NSGA-II) [8] proposed by Deb et al. is employed for exploration because this algorithm is effective and widespread employed to many optimization problems [9, 10]. Initially, a parent population $P_{t=1}$ with the size of N is created randomly. Here, t indicates the number of generation. Each feasible solution is assigned a rank (the solution with lower rank is better) according to its objective function value. On the other hand, each infeasible solution is assigned a rank which is higher than the minimum rank for the feasible solutions. Between two infeasible solutions, the solution with a smaller constraint violation has a better rank. Then, after choosing N solutions with lower rank in the parent population, recombination and mutation are conducted to create an offspring population Q_t with the size of N. In order to introduce elitism, first, a combined population $R_t = P_t \cup Q_t$ with the size of 2 N is formed. Then, the solutions in R_t are sorted according to the ranks based on objective function values and constraint violation. Now, N solutions are chosen from R_t in the order of their ranks and make up a new population P_{t+1}. The procedure as described above is for one generation. The non-dominated solutions with the lowest rank are explored by repeating this procedure for a certain number of generations.

2.4 Kriging Model

Although GA is capable of finding the global optimum, it requires numerous function evaluations to realize population-based multipoint simultaneous exploration. For efficient global optimization, the Kriging surrogate model [11] is employed together with GA. The Kriging model is based on Bayesian statistics, and can adapt well to nonlinear functions. In addition, the Kriging model estimates not only the function values themselves but also their uncertainties. Based on these uncertainties, the expected improvement (EI) of an objective function, which may be achieved on the Kriging model by adding a new sample point, is estimated. Maximizing the EI instead of the original objective function itself, the location of an additional sample point is determined for updating the Kriging model. Adding new samples to the Kriging model based on EI iteratively, these samples are expected to reach the global optima under the uncertainty of the Kriging model. Efficient Global Optimization (EGO) proposed by Jones et al. exploits these characteristics and is widely employed for optimization [12].

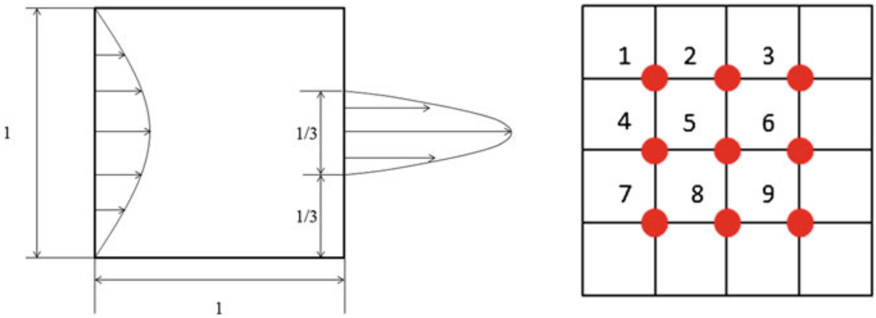
Since the present optimization is capable of topological change, the flow channels may often become unconnected depending on design variable values. Since such unconnected channels make it difficult to evaluate the objective function values, they should not be considered as an additional sample point for the Kriging model. In order to deal with this issue, the original EI value of the objective function is multiplied by the probability that the objective function value may be below a certain threshold estimated on the Kriging model. Maximizing this value, the location of an additional sample point is determined for searching the global optima while assuring the connectivity of flow channels under the uncertainty of the Kriging model.

3 Optimization Problems of Minimizing Pressure Loss

3.1 Nozzle Example (Case1)

3.1.1 Problem Definition

First, the single-objective optimization to minimize pressure loss of a nozzle shown in Fig. 2a is conducted. In this problem, since the effect of body force such as gravity is neglected, the flow field is vertically symmetric. Therefore, the nozzle shape is also considered to be vertically symmetric. The current design variables are the values of $\phi(\boldsymbol{x})$ given at six red points 1–6 as shown in Fig. 2b. $\phi(\boldsymbol{x})$ values at the



(a) design domain

(b) Layout of control points

Fig. 2 Geometry of nozzle (Case 1)

remaining points 7–9 are the same as those at the points 1–3 due to the vertical symmetry. Thus, the number of the design variables is six. At the outer boundary of the design domain, a fixed value (0.5 or −0.5) is given to represent inlets and outlets of the channels. The wall boundaries of the channels are then obtained by solving the Laplace's equation in the entire design domain. This domain is discretized as a 240×240 uniform Cartesian grid. The Reynolds number based on the width of inlet is 10. At the inlets, velocity is set as the Dirichlet boundary condition (parabolic profile with the reference velocity of 1) and pressure is set as the Neumann condition. At the outlets, on the other hand, velocity is set as the Dirichlet boundary condition (parabolic profile with the reference velocity of 3) and pressure is set as the Dirichlet condition (zero pressure).

The objective function is to minimize pressure loss. The following three constraints are considered; (1) the area of flow channels is less than 50% of the simulation domain, (2) flow channels go from inlets to outlets without dead ends, and (3) pressure loss is less than a threshold value. The third constraint aims to avoid aberrant flow channels with excessive pressure loss, which may make the estimation accuracy of the Kriging model worse. Moreover, in this case, since the boundary of flow channel is curved as can be seen in the previous work [13], infinitesimal control points are suitable to represent various topological changes.

3.1.2 Results

For the Kriging surrogate model, 300 initial sample points, satisfying the constraints (1) and (2) are generated randomly and their objective function (pressure loss) is evaluated by CFD. Then, a threshold of pressure loss in the constraint (3) is considered to remove sample points with excessive pressure loss from the initial sample points of the Kriging model. The Kriging model is constructed with sample points whose pressure loss is below the threshold. The appropriate threshold is chosen by observing the estimation accuracy of the Kriging model with different threshold. In this case,

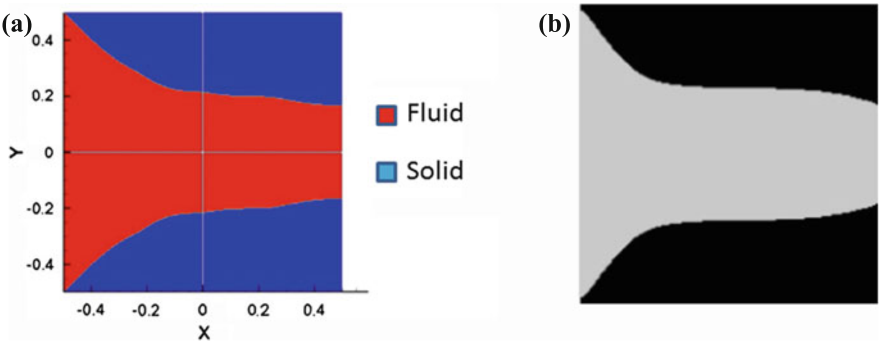


Fig. 3 Optimal solutions (Case 1) **a** Proposed method **b** Previous work [13]

the threshold of the pressure loss is set to be 100, and the number of the initial sample points satisfying all constraints is 175. GA identifies the solution, which maximizes the EI value on the Kriging model, as an additional sample point. The Kriging model is updated after adding this new sample point. In Case 1, the Kriging model is updated 15 times.

Figure 3 shows the optimal flow channel and it is similar to the flow channel reported by the previous study [13]. Several other studies [3, 14] indicate that the inlet of the optimal shape gets narrowed. However, such shape has larger pressure loss than the optimal shape obtained in this study. Moreover, when finer mesh is employed, the width of inlet becomes larger as shown in Fig. 3b. Thus, it is thought that the optimal shapes with narrow inlet reported in the previous study [3, 14] come from the mesh resolution or the grayscale material caused by the relaxation scheme.

3.2 *Double Pipe Example (Case 2)*

3.2.1 Problem Definition

Next, the design domain with two inlets and two outlets as shown in Fig. 4 is considered. In this case, two types of the design domain with $\delta = 1$ (square domain) and $\delta = 1.5$ (rectangular domain) are considered to compare with previous study [13].

In this case, control points need to be able to represent the straight boundaries. The number, location and radius of control points are investigated before conducting the optimization. Consequently, the control points are assigned to the layout as shown in Fig. 5. Furthermore, for the square domain, each control point is treated as a circle to represent the boundary as a smooth curve. The radius of the circle is fixed to 5 cells so as not to interfere with the circles at adjacent control points. $\phi(\boldsymbol{x})$ values are assigned to the control points assuming vertical symmetry for each domain (i.e., first and sixth rows, second and fifth rows, and third and fourth rows are symmetric in the

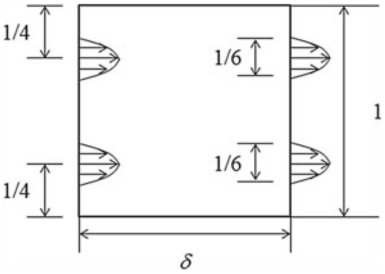
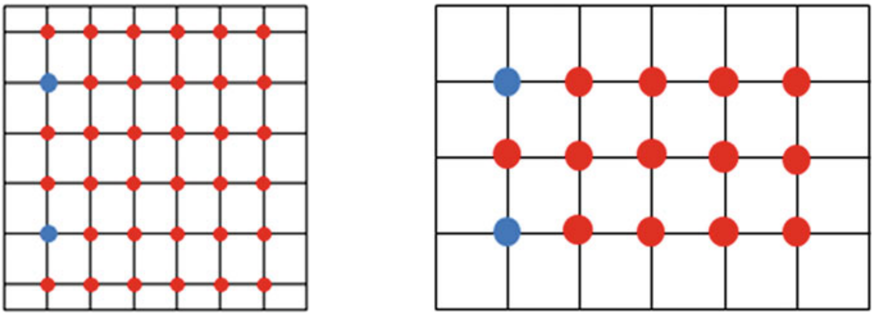


Fig. 4 Geometry of double pipe (Case 2)



(a) Square domain

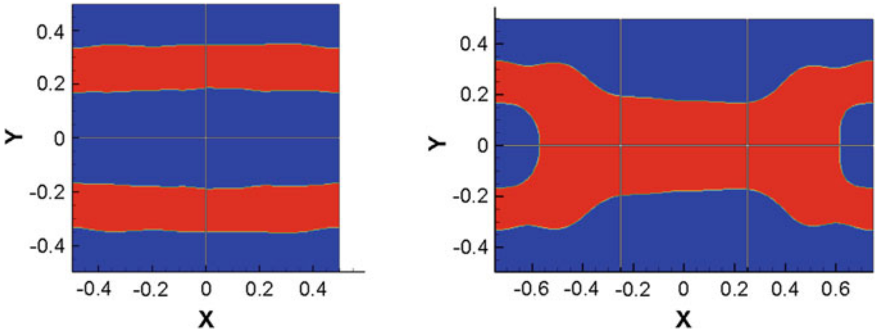
(b) Rectangular domain

Fig. 5 Layout of control points (Case 2)

square domain, first and third rows are symmetric in the rectangular domain). The number of the design variables is 18 in the square domain and 10 in the rectangular domain. In both domains, only a positive value of $\phi(\boldsymbol{x})$ is allowed at the blue point to assure the flow channel connectivity. The flow velocity is given as a parabolic profile with the reference velocity of 1 for both inlets and outlets in both domains. This case employs the same Reynolds number, mesh resolution (i.e., 240×240 uniform Cartesian grid for square domain, 240×360 uniform Cartesian grid for rectangular domain), and boundary conditions as Case 1. The constraints considered here are (1) the volume fraction (i.e., the area of flow channels is less than 40% of the design domain) and other two constraints (2) and (3) which are the same as those of Case 1.

3.2.2 Results

In order to prepare the initial sample points of surrogate model, 400 sample points satisfying the constraints (1) and (2) are generated randomly for each domain. Then, pressure loss is evaluated as the objective function for each sample point, and a threshold of pressure loss in the constraint (3) is set to be 50 for the square domain



(a) Square domain

(b) Rectangular domain

Fig. 6 Optimal Solutions (Case 2)

and 20 for the rectangular domain. Thus, the number of the initial sample points is reduced to 225 for the square domain and 244 for the rectangular domain. Figure 6a shows the optimal flow channel in the square domain obtained after updating the Kriging model 16 times, and Fig. 6b shows the optimal flow channel in the rectangular domain obtained after updating the Kriging model 20 times.

The square domain yields two single straight channels from the inlets to the outlets, and the rectangular domain yields two channels joined together in the center of the domain. These results agree with the analytical results obtained in the past [3, 13, 14].

4 Optimization Problems of Maximizing Heat Transfer (Case 3 and 4)

4.1 Problem Definition

The single-objective optimization to maximize heat transfer is conducted. In this case, in addition to the 2D incompressible Navier-Stokes equation, the 2D energy equation is also solved to evaluate the temperature field. The design domain is shown in Fig. 7. This case employs the same pressure and velocity boundary conditions as Case 2. The flow velocity is given as a parabolic profile with the reference velocity of 1 for the inlet and the outlet. Fluid temperature is set to be 0 at the inlet, and given by the Neumann condition at the outlet. Furthermore the temperature at the solid-fluid interface is expressed by the third type boundary condition as

$$q_w = h(T_\infty - T_w)$$

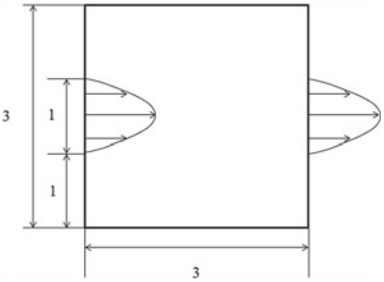


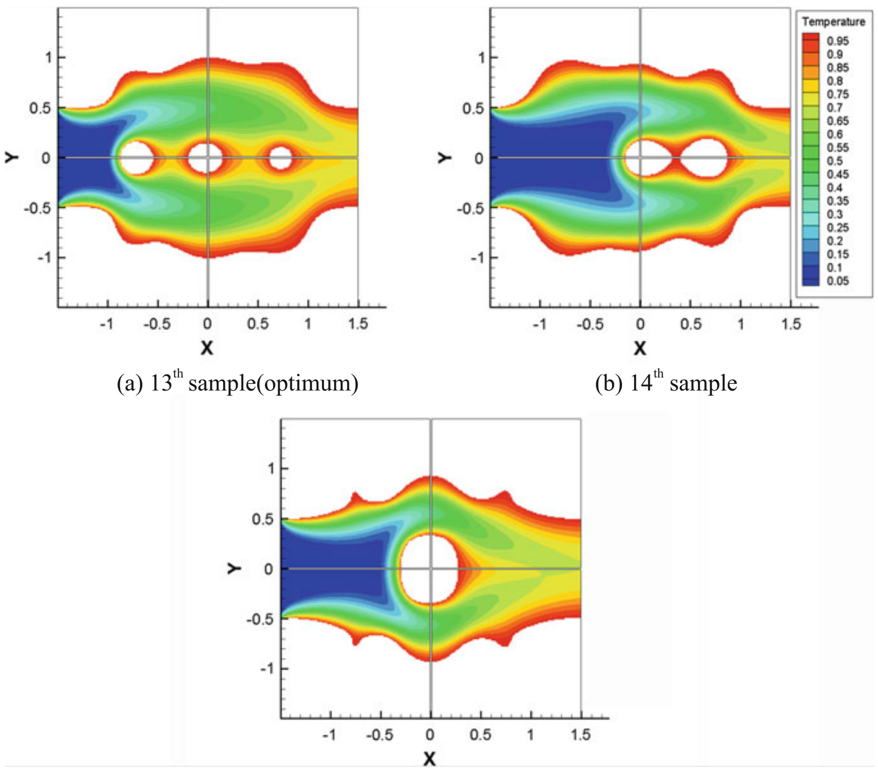
Fig. 7 Geometry of single pipe (Case 3 and 4)

where T_∞ is the temperature of the solid set to be 1, T_w is the fluid temperature on the wall, and h corresponds to the heat transfer coefficient. In this study, h is given by the Nusselt number: $Nu = hl/k$. Here, the reference length l is the width of inlet set to be 1 and thermal conductivity k is also set to be 1. Thus, the heat transfer coefficient is equal to the Nusselt number. The objective function is to maximize the bulk mean temperature at the outlet. In this problem, since the possible range of the objective function value is known thermodynamically (i.e., the maximum is not greater than the wall temperature: 1 and the minimum is not less than the inlet temperature: 0), different from Cases 1 and 2, this case does not introduce a threshold into the objective function value because it can reduce the diversity of the population in GA.

In order to compare the present method with the previous study [15], this case employs the same dimensionless numbers as those in the previous study; the Reynolds number, the Nusselt number, and the Prandtl number are set to be 5, 50, and 6.78, respectively. The control points are assigned the same as Fig. 2b. In this problem, two cases are considered. In the first case (Case 3), the control points are treated as infinitesimal points, and the number of the design variables is six. In the second case (Case 4), the control points are treated as circles whose radius can change in the range of $0 \leq r \leq 5$ as an additional design variable, thus the number of the design variables is seven.

4.2 Results

First, the Case 3 with six design variables is discussed. In this case, 112 initial sample points satisfying the connectivity from the inlet to the outlet are used to construct the Kriging model. The Kriging model is updated 15 times. Figure 8 shows representative flow channels found in the additional samples. In this case, several flow channels, each of which has similar objective function values as shown in Table 1 but with different topology, are found as the local optima. The optimum reported in the previous study [15] has one solid island inside the channel whose topology is the same as Fig. 8c. This result indicates that the objective function is a multi-modal function and there are other local optima with similar performance to the global



(a) 13th sample(optimum)

(b) 14th sample

(c) 15th sample

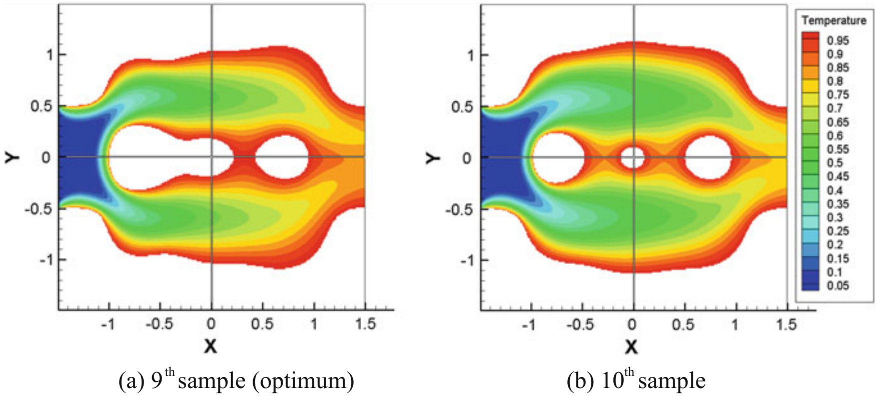
Fig. 8 Temperature distributions of the representative local optima in the additional samples (Case 3)

Table 1 Objective function value of local optima (Case 3)

	Temperature
(a) 13th sample	0.769
(b) 14th sample	0.730
(c) 15th sample	0.751

optimum. Thus, it is required to employ a method of population-based multipoint simultaneous exploration such as GA. The result also indicates that, as the number of the solid islands in the channels increases, the size of each island becomes smaller. This is because the size of island is determined by the interface between the solid and fluid to achieve an equal amount of heat flux on the interface.

Next, Case 4 with seven design variables is discussed. In this case, 126 initial sample points are used to construct the Kriging model. The Kriging model was updated 10 times. Figure 9 shows representative flow channels. Also in this case, several flow channels with different topology but with similar objective function



(a) 9th sample (optimum) (b) 10th sample

Fig. 9 Temperature distributions of the representative local optima in the additional samples (Case 4)

Table 2 Objective function values of local optima (Case 4)

	Temperature
(a) 9th sample	0.833
(b) 10th sample	0.790

values, as shown in Table 2, are found. Moreover, different from Case 3, this case finds the channels with a coupled island. Case 4 that allows a coupled island leads to the optimum with a better objective function value than Case 3. However, since the bulk mean temperature reaches its upper limit thermodynamically in this case, several flow channels with different topology are still found as the local optima. Thus, it can be concluded that the objective function of this problem is a multi-modal function and the length of the interface between fluid and solid has a significant effect to increase the temperature.

Note that the authors conducted another case with different Reynolds and Nusselt numbers. It is revealed that as the Reynolds number increases, putting large coupled islands are required to increase temperature. Moreover, several flow channels with different topology were also found. These results agree with the knowledge obtained in Cases 3 and 4.

5 Multi-objective Optimization Problems

5.1 Problem Definition

Finally, a multi-objective optimization problem to minimize pressure loss and maximize heat transfer of flow channels is considered. The design domain is the same as that shown in Fig. 7, the layout of the control points is same as that shown in Fig. 2b. This case employs the same velocity, pressure, temperature boundary conditions, dimensionless numbers, and mesh resolution as Case 3. In this problem, the design variables are treated the same as Case 4 with seven design variables (i.e., $\phi(\boldsymbol{x})$ values are given at the control points assuming vertical symmetry in the range of $|\phi(x)| \leq 1$, and the radius of the control points changes in the range of $0 \leq r \leq 5$).

5.2 Results

There are a number of non-dominated solutions, which are not worse than any other solution regarding all objective functions, in a multi-objective optimization problem while there is only one optimal solution in a single-objective optimization problem. Thus, it is important to ensure the diversity of the solutions in GA and capture the trade-off among objective functions. Therefore, it should be careful to introduce a threshold of the pressure loss while keeping the diversity of the solutions in GA. However, without a threshold, initial sample points including excessive pressure loss are used to construct the Kriging model, which may make the estimation accuracy of the Kriging model worse. Since the width of the flow channels with excessive pressure loss is very narrow, these flow channels can be regarded as disconnected. Moreover, the bulk mean temperature of such disconnected or nearly disconnected flow channels is evaluated to be 0 in Cases 3 and 4. Thus, such solutions are hardly able to be the non-dominated solutions, and it will not lose the diversity of solutions even if such solutions are removed from initial sample points.

First, in order to prepare the initial sample points of the Kriging model, 400 sample points satisfying connectivity from the inlet to the outlet are generated randomly. In this case, a threshold of the pressure loss is set to be 13. Thus, the number of the initial samples is reduced to 215. Multi-objective optimization employs several additional sample points every time the Kriging model is updated in contrast to a single-objective optimization employs one additional sample point for every update. This study performs cluster analysis using the k-means method [16] to select representative sample points from many non-dominated solutions obtained by maximizing the EI value of each objective function on the Kriging model. In this case, 4 additional sample points are chosen for every update.

Figure 10 indicates the initial sample points, the non-dominated solutions among them, and the non-dominated solutions obtained after the 19th update of the Kriging model in the objective space. The front of non-dominated solutions obtained after the 19th update is more convex to the optimization direction than that of the initial non-dominated set. The non-dominated solutions in the 19th update can be classified into 4 groups according to their characteristics.

First, in the yellow group, the flow channels have low pressure loss and low bulk mean temperature. These flow channels do not include any solid island inside the channels. This is consistent with the results of the single-objective problems as shown

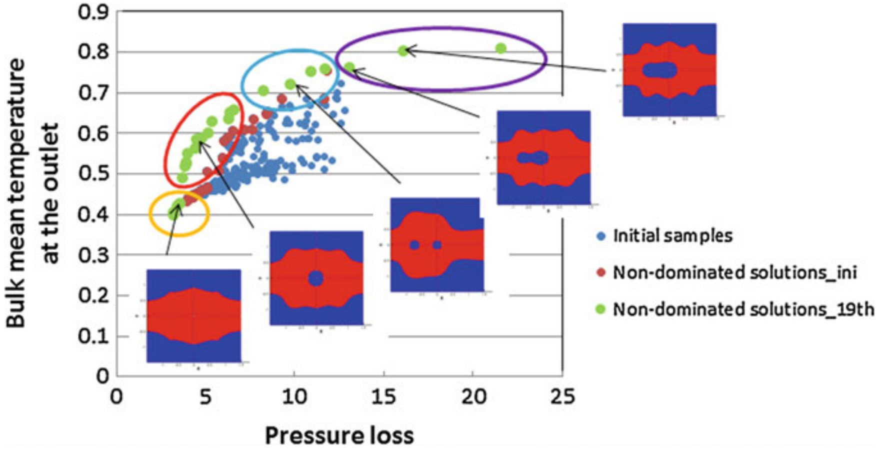


Fig. 10 The solution set in the multi-objective problem

in the previous sections that there are no islands if the pressure loss is low, and there is at least one island if the bulk mean temperature is high.

Second, the non-dominated front is discontinuous between the red group and the yellow group. The solutions in the red group get higher bulk mean temperature while their pressure loss does not increase drastically. All these solutions contain one island in the channel. In this group, as the island gets larger, the pressure loss and the temperature go higher. Thus, it is revealed that putting an island into a channel contributes to better performance of heat transfer.

Next, the solutions in the blue group have two isolated islands. In this group, the bulk mean temperature increases by inserting another island. However, the pressure loss also increases.

Finally, the solutions in the purple group have a coupled island. In this group, two significant results are revealed. First, these solutions keep the bulk mean temperature almost equal to the optimum found in Case 3 whereas the pressure loss is larger than any other solution in other three groups. Second, the trade-off between each objective function becomes weak. This result indicates that the bulk mean temperature reaches its upper limit thermodynamically in this case while the pressure loss can change drastically according to a subtle change of island's shape. Thus, recalling that the objective function of temperature is a multi-modal function as seen in Cases 3 and 4, this result indicates the possibility that if the shape of islands is closer to the streamline, it is able to design the flow channel that satisfies both high temperature and low pressure loss.

6 Conclusion

Global topology optimization was conducted in flow channel design problems that minimize pressure loss and/or maximize heat transfer using a Kriging-surrogate-based genetic algorithm.

In the single-objective optimization to minimize pressure loss, two cases with different layout were conducted and the results agreed with the previous study.

In the single-objective optimization to maximize the bulk mean temperature, the GA found not only the optimal shape, but also several shapes that have quite similar objective function values but with different topologies from each other. Thus, the objective function of temperature seemed to be a multi-modal function. The results also indicated that putting a solid island in a fluid region and increasing the interface between fluid and solid have significant effects to increase the temperature.

Finally, combining above two problems, multi-objective optimization to minimize pressure loss and to maximize heat transfer was conducted. As a result, it was revealed that the size and the shape of a solid island inserted in a flow channel were the most important factors to increase the temperature and determine the pressure loss. However, with a certain size of island, the temperature reached its upper limit and the trade-off between pressure loss and heat transfer became weak. Thus, it was significant in the multi-objective optimization to capture an entire trade-off between these objective functions and to find the points where the pressure loss increases drastically and the heat transfer was saturated. The present result was expected to help us to be able to design the flow channels that satisfy both high temperature and low pressure loss with further investigation.

Thus, the proposed method showed its capability to explore global optima for both single-objective and multi-objective topology optimization in flow problems.

References

1. Bendsoe, M.P., Sigmund, O.: Topology Optimization: Theory, Methods and Applications. Springer (2003)
2. Bendsøe, M.P., Kikuchi, N.: Generating optimal topologies in structural design using a homogenization method. Comput. Methods Appl. Mech. Eng. **71**(2), 197–224 (1988)
3. Borrvall, T., Petersson, J.: Topology optimization of fluids in Stokes flow. Int. J. Numer. Methods Fluids **41**(1), 77–107 (2003)
4. Sethian, J.A., Wiegmann, A.: Structural boundary design via level set and immersed interface methods. J. Comput. Phys. **163**(2), 489–528 (2000)
5. Allaire, G., Jouve, F., Toader, A.M.: Structural optimization using sensitivity analysis and a level-set method. J. Comput. Phys. **194**(1), 363–393 (2004)
6. Nakahashi, K., Kim, L.S.: Building-cube method for large-scale, high resolution flow computations, AIAA paper 2004-0423 (2004)
7. Mittal, R., Dong, H., Bozkurttas, M., Najjar, F.M., Vargas, A., von Loebbecke, A.: A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. J. Comput. Phys. **227**(10), 4825–4852 (2008)
8. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGAII. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)

9. Richardson, J.N., Rajan, F.C., Sigrid, A.: Robust topology optimization of truss structures with random loading and material properties: a multiobjective perspective. Comput. Struct. **154**, 41–47 (2015)
10. Guirguis, D., Hamza, K., Aly, M., Hegazi, H., Saitou, K.: Multi-objective topology optimization of multi-component continuum structures via a Kriging-interpolated level set approach. Struct. Multidiscip. Optim. **51**(3), 733–748 (2015)
11. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. J. Glob. Optim. **13**(4), 455–492 (1998)
12. Viana, F.A.C., Simpson, T.W., Balabonov, V., Toropov, V.: Metamodeling in multidisciplinary design optimization: how far have we really come? AIAA J. **52**(4), 670–690 (2014)
13. Challis, V.J., Guest, J.K.: Level set topology optimization of fluids in Stokes flow. Int. J. Numer. Methods Eng. **79**(10), 1284–1308 (2009)
14. Guest, J.K., Prévost, J.H.: Topology optimization of creeping fluid flows using a Darcy-Stokes finite element. Int. J. Numer. Methods Eng. **66**(3), 461–484 (2006)
15. Matsumori, T., Kondoh, T., Kawamoto, A., Nomura, T.: Flow chanel design in heat exchanger for maximum heat transfer by topology optimization. In: Proceedings of 20th Design & Systems Conference, vol. 3103, pp. 1–4 (2010) (in Japanese)
16. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Comput. Surv. (CSUR) **31**(3), 264–323 (1999)

Topology Optimization Using GPGPU

Stefan Gavranovic, Dirk Hartmann and Utz Wever

Abstract In this paper we present a matrix-free geometric multigrid method for solving a linear system of equations needed at every iteration of the topology optimization process. The multigrid solver is parallelized on an Nvidia graphics card using CUDA, therefore reducing simulation time drastically. This enables users to derive optimal topologies represented with a high number of elements while having low execution time. Computational domain is discretized with a regular structured hexahedral mesh. To improve the accuracy of the non-conformal discretizazion, the Dirichlet boundary conditions are imposed in a weak form using Nitsche method.

1 Introduction

Additive manufacturing is driving a revolution in manufacturing [17]. With this technique we can produce objects by successively adding thin layers of material. Nowadays this procedure is used to obtain a wide variety of items such as plastic prototypes for engineers and designers, customized medical devices such as dental implants, hip implants, or hearing aids. Significant breakthrough was the use of additive manufacturing in aerospace industry [7], which meant that less material could be used compared to conventional production techniques. Therefore, the production costs were reduced, and the lighter aircraft components lead to significant fuel savings.

Since additive manufacturing results in nearly infinite design spaces, the importance of topology optimization [1] is constantly growing. Topology optimization represents an optimal placement of material within a given design space,

S. Gavranovic (✉)
Technical University of Munich, 80333 Munich, Germany
e-mail: stefan.gavranovic@tum.de; stefan.gavranovic@siemens.com

D. Hartmann (✉) · U. Wever
Siemens AG, Corporate Technology, 80200 Munich, Germany
e-mail: hartmann.dirk@siemens.com

U. Wever
e-mail: utz.wever@siemens.com

E. Minisci et al. (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, Computational Methods in Applied Sciences 48, https://doi.org/10.1007/978-3-319-89988-6_33

boundary conditions, and loads in order to satisfy the prescribed objective functions. During the design process, topology optimization enables engineers to explore different design solutions that meet the design requirements with optimum material usage while preserving required structural integrity.

In recent years a lot of research was invested in exploring and establishing the theory of topology optimization. The application field of topology optimization has expanded beyond structural analysis to include fluid flow, acoustics, heat transfer, nanophotonic devices, and material designs [4]. However, most of the research was carried out for 2D models. Due to high computational costs, performing topology optimization on 3D models may require hours, or in some cases even days, which hinders rapid prototyping design process. Ideally, the designer would like to have almost instantaneous feedback when exploring the design space. Not as much research was conducted in improving computational efficiency as it was done for establishing the theory of topology optimization, hence it stays still an open topic for the research [13]. Therefore, in this paper we investigate the use of the multi-core architecture such as GPU (Graphics Processing Unit) by utilizing parallel programming framework CUDA (Compute Unified Device Architecture) [11]. Since the optimization process comes with a high computational price of performing the finite-element method (FEM) analysis at each optimization step, the main focus of this work is to develop an efficient solver for performing FEM analysis. In order to accomplish this, several steps are taken. The computational domain under consideration is discretized with the help of hexahedral elements, yielding a system of linear equations. This enables the use of highly efficient matrix-free geometric multigrid methods [10] for solving the linear system of equations. Geometric multigrid algorithm is adapted in such a way that it maps to GPU hardware [5], therefore resulting in execution times far superior to those when solving the problem on CPU.

2 Previous Work

The goal of this section is to give a short overview of previous research conducted in the field of topology optimization with a focus on computational efficiency. In one of the earliest works [2] in this field, parallel computing in combination with domain decomposition was used. Test geometries were discretized using approximately 196,000–884,000 elements depending on the test model. System of linear equations was solved by using the preconditioned conjugate gradient method. Simulations for several test cases were performed on a Cray T3E using 16–24 processors depending on the test case, with solution times ranging from 4 to 43 h. A lot of improvements in numerical algorithms and in hardware were introduced since then, reducing simulation times drastically.

One of the first works demonstrating the use of GPU in topology optimization was presented in [14]. Here, the most time consuming part of the topology optimization process, which is solving a system of linear equations, is parallelized on GPU using CUDA. The author implemented a matrix-free conjugate gradient method with modifications to the single precision computation, necessary due to the hardware limitations. The investigated GPU was a GeForce GTX280 with 1 GB device memory and 240 CUDA cores each running at 1.30 GHz. The GPU execution time was up to 60 times faster in comparison to CPU. For a test case with resolution of nearly 1,000,000 elements and a lower class GPU such as GeForce 9600M GT with 32 CUDA cores, each running at 0.78 GHz, the authors reported simulation time under 2 h.

In a more recent work [16], the author used the Jacobi-preconditioned conjugate-gradient method for solving the system of linear equations on a GPU. The author was using hexahedral elements which conform the boundary. The used hardware was the Nvidia GeForce GTX 480 with 480 CUDA cores and 1.5 GB of device memory. Numerical results for several test cases were presented which we will discuss in the Results section.

3 Finite Element Formulation

Discretization of the computational domain is performed using hexahedral elements. CAD (Computer-Aided Design) geometry in STEP (Standard for the Exchange of Product model data) format is passed as an input and discretized using the Open CASCADE (Computer Aided Software for Computer Aided Design and Engineering) [12] library. Using hexahedral elements implies that we do not have to perform element rotation in space nor any other transformation. Every hexahedral element has an identical stiffness matrix which can be analytically pre-computed.

3.1 Linear Elasticity Equations

Elastic deformation of a continuum body on the domain Ω with Dirichlet Γ_D and Neumann Γ_N boundaries is described by the following equations [3, 18]:

$$\nabla \cdot \sigma + f = 0 \qquad \forall x \in \Omega \tag{3.1.1}$$

$$\sigma = C : \varepsilon \qquad \forall x \in \Omega \tag{3.1.2}$$

$$\varepsilon = \frac{1}{2} \cdot (\nabla u + (\nabla u)^\top) \qquad \forall x \in \Omega \tag{3.1.3}$$

$$u = \hat{u} \qquad \forall x \in \Gamma_D \tag{3.1.4}$$

$$\sigma \cdot n = t \qquad \forall x \in \Gamma_N \tag{3.1.5}$$

where σ and ε represent stress and strain tensor respectively. f represents the force term, C is the elasticity tensor, $\hat{u}$ is the prescribed displacement on the Dirichlet boundary, n is a normal vector to the Neumann boundary, t is a traction vector. After obtaining the weak form from Eq. (3.1.1) we use tri-linear shape functions to approximate our solution $\mathbf{u}_h$ of the problem:

$$\mathbf{u}_h = \sum^{i} N_i(\mathbf{x}) \cdot \mathbf{u}_i \tag{3.1.6}$$

where $N_i(\mathbf{x})$ represent the shape functions that span the finite dimensional space $\mathbb{V}_h$, and $\mathbf{u}_i$ are coefficients associated with shape functions. The element stiffness matrix $\mathbf{K}_e$ obtained from discretization can be written as:

$$\mathbf{K}_e = \int\limits_{\Omega_e} \mathbf{B}_e^\top \mathbf{C} \mathbf{B}_e d\Omega \tag{3.1.7}$$

where $\mathbf{B}_e$ represents the strain-displacement matrix and $\mathbf{C}$ is the elasticity matrix. The element stiffness matrix $\mathbf{K}_e$ is computed only once and we use it for all elements in our computational domain.

3.2 Per Node Equations

Following the work of [5], we do not want to assemble and store the full stiffness matrix. We rather operate on per node equations. That is, each node is assigned with 3 degrees of freedom, and with 27 3 × 3 matrices which represent the interaction of the node with its neighbors. These matrices are extracted from the element stiffness matrix $\mathbf{K}_e$ which we precomputed. In order to show the assembling process of the aforementioned per node equations, let us set up the per element equations in the following manner:

$$\sum_{j=1}^{8} \mathbf{k}_{i,j}^{e_k} \cdot \mathbf{u}_j^{e_k} = \mathbf{f}_i^{e_k} \qquad\qquad i = 1, \ldots, 8. \tag{3.2.1}$$

We sum over all 8 nodes of a hexahedral element e_k, where $\mathbf{k}_{i,j}^{e_k}$ represents 3 × 3 block matrix extracted from k-th element stiffness matrix, and associated with each node j. $\mathbf{u}_j^{e_k}$ is the corresponding displacement and $\mathbf{f}_i^{e_k}$ is the force term acting on the given element. Equation (3.2.1) for a given node i represents the influence of an element e_k on the displacement of the node i. Since this node is shared by the 8 adjacent elements, we add up equations yielding from all k elements as well. Thus, we obtain the per node equations:

$$\sum_{i=(-1,-1,-1)}^{(1,1,1)} \mathbf{M}_i^{nod} \mathbf{u}_{nod+i} = \mathbf{f}_{nod}, \tag{3.2.2}$$

where nod represents discrete coordinates of the node. By adding vector i to the discrete coordinates of the node we visit all the neighboring nodes. Here, $\mathbf{M}^{nod}$ is an array of 27 block matrices corresponding to the observed node and subindex i

determines the block matrix corresponding to the neighbor of node. $\mathbf{u}_{nod+i}$ are the displacements of the corresponding nodes and $\mathbf{f}_{nod}$ is the force term. For more details on the equation assembly process the reader may refer to the works [5, 8].

4 Multigrid Solver

Since the bottleneck of the optimization process is the performance of FEM analysis on each iteration step, we propose a multigrid method based on the work of [5]. Multigrid methods [10], used to provide fast numerical solvers especially for elliptic partial differential equations, can greatly improve simulation times if implemented efficiently. In this work we implemented a CUDA based matrix-free geometric multigrid method. We used a standard multigrid V-cycle with Gauss-Seidel relaxation.

Algorithm 1 Multigrid

```
function V- CYCLE($A^h$,$u^h$,$f^h$,$n_{steps}$,$level$)
    if $level = numLevels$ then
        Solve directly $A^h u^h = f^h$
    else
        Gauss-Seidel relaxation $A^h u^h = f^h$
        Compute residual $r^h = f^h - A^h u^h$
        Restrict residual $r^{h+1} = R_h^{h+1} r^h$
        $e^{h+1} \leftarrow$V- CYCLE($A^{h+1}$,0,$r^{h+1}$,$n_{steps}$,$level + 1$)
        Interpolate coarse grid error $e^h = P_{h+1}^h e^{h+1}$
        Apply correction $u^h = u^h + e^h$
        Gauss-Seidel relaxation $A^h u^h = f^h$
```

4.1 Per Node Equations Assembly for All Levels

Each level of the multigrid hierarchy is organized in such a way that if there is at least one active fine grid cell that is covered by the coarse grid cell, we consider the cell on the coarse level to be active as well. Before we start performing the V-cycle, we assemble per-node equations for all simulation levels [5, 8]. On the finest level we assemble equations using precomputed element stiffness matrix, afterwards we use Galerkin coarse grid operator to assemble equations on the coarser levels.

4.2 Gauss-Seidel Relaxation

To apply the smoothing step to our linear system of equations we use a Gauss-Seidel smoother. We first divide the set of nodes in 8 groups, such that we could perform smoothing in parallel, as suggested in [5, 8]. With respect to our per node equations the smoothing step can be formulated as:

$$\mathbf{u}_{nod}^{k+1} = \mathbf{u}_{nod}^{k} + \omega \cdot (\mathbf{M}_{0,0,0}^{nod})^{-1} \cdot \left(\mathbf{f}_{nod} - \sum_{i=(-1,-1,-1)}^{(0,0,-1)} \mathbf{M}_i^{nod} \mathbf{u}_{nod+i}^{k+1} - \sum_{i=(0,0,0)}^{(1,1,1)} \mathbf{M}_i^{nod} \mathbf{u}_{nod+i}^{k} \right) \tag{4.2.1}$$

where $\mathbf{u}_{nod}^{k}$ is the current value, and $\mathbf{u}_{nod}^{k+1}$ is the updated value of displacement at a given node, and ω is a relaxation coefficient.

5 Topology Optimization Formulation

In our work we decided to use SIMP (Solid Isotropic Microstructure with Penalization) approach for performing the topology optimization. This method is proposed as "artificial density approach" by [1]. The domain is discretized with hexahedral elements, where to each hexahedra a density variable ρ is assigned. These density variables are used as design variables in the optimization process for meeting the desired objective function requirements. The main advantage of the SIMP method over other methods is an easy implementation and a well established theoretical foundation. The topology optimization problem formulation is given by:

$$\begin{aligned} \underset{\rho}{\text{minimize}} \quad & c(\rho) = \mathbf{f}^{\top}\mathbf{u} = \mathbf{u}^{\top}\mathbf{K}(\rho)\mathbf{u} \\ \text{subject to} \quad & \frac{V(\rho)}{V_0} = \alpha \\ & \mathbf{K}(\rho)\mathbf{u} = \mathbf{f} \\ & 0 < \rho_{min} \leq \rho \leq 1. \end{aligned}$$

That is, we wish to minimize the compliance $c(\rho)$ subjected to a volume constraint of a given volume fraction α, being the ratio between the material volume $V(\rho)$ and the design domain volume V_0. Displacement and force vectors are denoted respectively $\mathbf{u}$ and $\mathbf{f}$. The element stiffness matrix is denoted as $\mathbf{K}(\rho)$. For solving the aforementioned optimization process we use the Optimality Criteria method. Identical to work of [1, 15], we update our density design variables ρ as follows:

$$\rho_e^{new} = \begin{cases} \max(\rho_{min}, \rho_e - \delta\rho) \\ \quad \text{if} \quad \rho_e B_e^{\eta} \leq \max(\rho_{min}, \rho_e - \delta\rho) \\ \min(1, \rho_e + \delta\rho) \\ \quad \text{if} \quad \min(1, \rho_e + \delta\rho) \leq \rho_e B_e^{\eta} \\ \rho_e B_e^{\eta} \\ \quad \text{if} \quad \max(\rho_{min}, \rho_e - \delta\rho) < \rho_e B_e^{\eta} < \min(1, \rho_e + \delta\rho) \end{cases} \tag{5.0.1}$$

where the element density ρ_e is our design variable, $\delta\rho$ is a non-negative increment of design variable, and the exponent $\eta = 1/2$ is a numerical damping coefficient. The update value B_e is given by the optimality condition:

$$B_e^\eta = \frac{-\partial c/\partial \rho_e}{\lambda \partial V/\partial \rho_e} \tag{5.0.2}$$

where λ is a Lagrangian multiplier that we obtain by applying a bi-section algorithm. From the equation (5.0.2) we compute the sensitivity of the objective function as:

$$\frac{\partial c}{\partial \rho_e} = -p(\rho_e)^{p-1}\mathbf{u}_e^\top \mathbf{K}_0 \mathbf{u}_e \tag{5.0.3}$$

$$\frac{\partial V}{\partial \rho_e} = 1. \tag{5.0.4}$$

6 Imposing Dirichlet Boundary Conditions

With the use of non-conformal hexahedral elements, as typically used in topology optimization problems, the challenge of accurately imposing boundary conditions arises. Different from tetrahedral meshes that conform the boundary, hexahedral meshes are embedding the boundary. Hence, it is necessary to enforce Dirichlet boundary conditions in a weak sense. By adding the terms to the weak formulation of the elasticity equation we impose values on the embedded boundary. Identical to the work of [3, 18], the strong formulation of the linear elasticity problem is transformed into the weak form using the principle of minimum of potential energy. The total potential energy Π_{tot} of the body at rest is the sum of the internal Π_{int} and the external Π_{ext} potential energy:

$$\Pi_{tot} = \Pi_{int} + \Pi_{ext} \tag{6.0.1}$$

where

$$\Pi_{int} = \frac{1}{2}\int_{\Omega} \varepsilon(\mathbf{u}) : \mathbf{C} : \varepsilon(\mathbf{u}) d\Omega \tag{6.0.2}$$

$$\Pi_{ext} = \int_{\Omega} \mathbf{u}\cdot\mathbf{f} d\Omega + \int_{\Gamma_N} \mathbf{u}\cdot\mathbf{t} d\Gamma. \tag{6.0.3}$$

After finding the first variation of the total potential energy, and setting it to zero, we obtain the weak formulation:

$$\int_{\Omega} \varepsilon(\mathbf{v}) : \mathbf{C} : \varepsilon(\mathbf{u}) d\Omega = \int_{\Omega} \mathbf{v}\cdot\mathbf{f} d\Omega + \int_{\Gamma_N} \mathbf{v}\cdot\mathbf{t} d\Gamma \tag{6.0.4}$$

where **v** are the test functions and **u** are the shape functions.

6.1 Nitsche Terms

Following the works of [3, 6, 18] constraint potentials are added to the total energy potential Π_{tot}. The Nitsche constraint potential is obtained by using a combination of Langrange Multiplier Method and Penalty Method:

$$\Pi_{Lagrange} = \int\limits_{\Gamma_D} \lambda \cdot (\mathbf{u} - \hat{\mathbf{u}}) d\Gamma \tag{6.1.1}$$

$$\Pi_{Penalty} = \frac{1}{2} \cdot \beta \int\limits_{\Gamma_D} (\mathbf{u} - \hat{\mathbf{u}})^2 d\Gamma \tag{6.1.2}$$

where λ is defined as in [18]:

$$\lambda = -\mathbf{C} : \varepsilon(\mathbf{u}) \cdot \mathbf{n}. \tag{6.1.3}$$

The β from equation (6.1.2) is typically chosen [6] as the ratio of the area of the embedded surface Γ_e and the volume of the partial hexahedron H_e, cut by the embedded surface:

$$\beta \geq \frac{measure(\Gamma_e)}{measure(H_e)}. \tag{6.1.4}$$

Having added the constraint energy potentials (6.1.1) and (6.1.2) to the total potential energy, we find the minimum of the potential by setting the first variation to zero. Hence we obtain the weak formulation with the Nitsche terms:

$$\begin{aligned} &\int\limits_{\Omega} \varepsilon(\mathbf{u}) : \mathbf{C} : \varepsilon(\mathbf{u}) d\Omega - \int\limits_{\Gamma_D} (\varepsilon(\mathbf{v}) : \mathbf{C}) \cdot \mathbf{n} \cdot \mathbf{u} d\Gamma - (\varepsilon(\mathbf{u}) : \mathbf{C}) \cdot \mathbf{n} \cdot \mathbf{v} d\Gamma + \beta \int\limits_{\Gamma_D} \mathbf{v} \cdot \mathbf{u} d\Gamma \\ &\quad = \int\limits_{\Omega} \mathbf{v} \cdot \mathbf{f} d\Omega + \int\limits_{\Gamma_N} \mathbf{v} \cdot \mathbf{t} d\Gamma - \int\limits_{\Gamma_D} (\varepsilon(\mathbf{v}) : \mathbf{C}) \cdot \mathbf{n} \cdot \hat{\mathbf{u}} d\Gamma + \int\limits_{\Gamma_D} \mathbf{v} \cdot \hat{\mathbf{u}} d\Gamma. \end{aligned} \tag{6.1.5}$$

After discretization [18], we have the following formulation for a hexahedral element with the embedded interface:

$$\begin{aligned} &\int\limits_{\Omega_e} \mathbf{B}_e^\top \mathbf{C} \mathbf{B}_e d\Omega - \int\limits_{\Gamma_D^e} \mathbf{B}_e^\top \mathbf{C} \cdot \mathbf{n} \cdot \mathbf{N} d\Gamma - \int\limits_{\Gamma_D^e} \mathbf{N}^\top \cdot \mathbf{n}^\top \cdot \mathbf{C} \mathbf{B}_e d\Gamma + \beta \int\limits_{\Gamma_D} \mathbf{N}^\top \cdot \mathbf{N} d\Gamma \\ &\quad = \int\limits_{\Omega} \mathbf{N}^\top \cdot \mathbf{f} d\Omega + \int\limits_{\Gamma_N} \mathbf{N}^\top \cdot \mathbf{t} d\Gamma - \int\limits_{\Gamma_D} \mathbf{B}_e^\top \mathbf{C} \cdot \mathbf{n} \cdot \hat{\mathbf{u}} d\Gamma + \int\limits_{\Gamma_D} \mathbf{N}^\top \cdot \hat{\mathbf{u}} d\Gamma. \end{aligned} \tag{6.1.6}$$

Additional terms in this equation with respect to the original discretization are used to enforce Dirichlet boundary conditions by modifying the original system of linear equations.

7 Results

In this section we intend to present some of the results computed with the help of our topology optimization tool. All of the following simulations were carried out on a machine with Intel Core i7-4710HQ processor running at 2.5 GHz, RAM memory of 8GB, and commodity graphics card GeForce GTX 860M with 640 CUDA cores and 4 GB of GDDR5 memory.

7.1 GE Challenge

General Electric (GE) raised an open challenge [9] where a jet engine bracket was to be optimized to withstand the working loads while having minimum mass. Ten selected designs were produced from a titanium alloy using a direct metal laser melting (DMLM) machine, which uses a laser beam to fuse layers of metal powder into a final shape [9]. Afterwards, the parts were sent to the destruction testing. We wish to present our design solution which resembles some of the winning designs.

Initial geometry shown in Fig. 1 is discretized with the help of approximately 950,000 hexahedral elements. Topology optimization on GPU was carried out in 14.4 s for the total of 9 iterations. Performing FEM analysis on average took 0.9 s. Optimized topology is illustrated in Fig. 2.

7.2 Bridge Design

For a design domain of a rectangular cuboid shape, subjected to the forces acting perpendicular to the mid-plane of the design domain, we obtain the optimal bridge

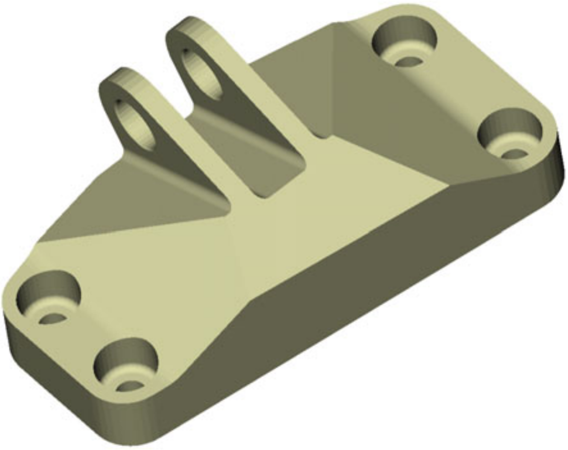


Fig. 1 Initial design of jet engine bracket provided by the GE challenge [9]

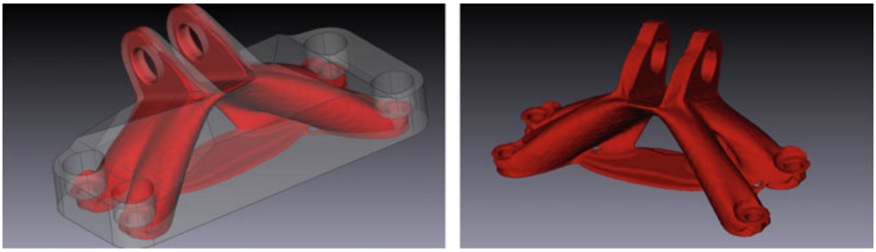
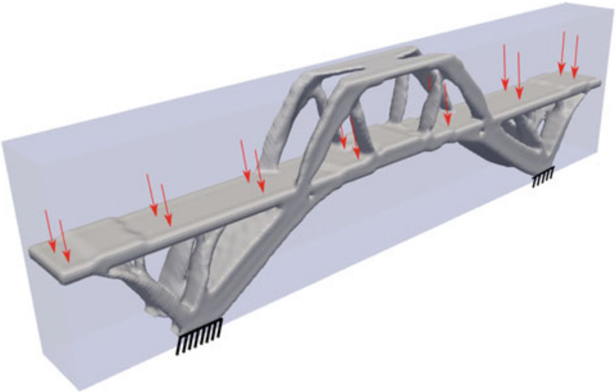
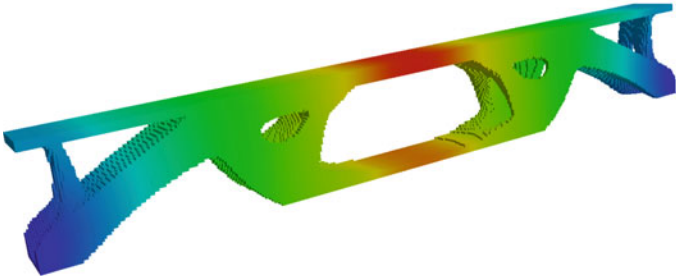


Fig. 2 Jet engine bracket, part of the GE challenge [9], optimized with volume fraction of $\alpha = 0.35$. The initial geometry is shown as a transparent body



(a) Computed for volume fraction of $\alpha = 0.2$. The initial geometry is shown as a transparent body.



(b) Computed for volume fraction of $\alpha = 0.35$. Red color depicts the areas of high displacements, while dark blue depicts the areas of low displacements.

Fig. 3 Optimal topology of a bridge

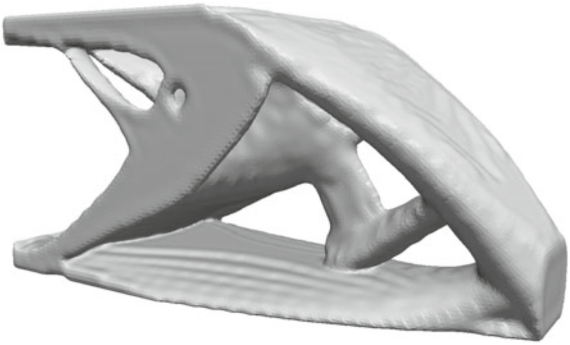


Fig. 4 Cantilever optimized for volume fraction $\alpha = 0.35$

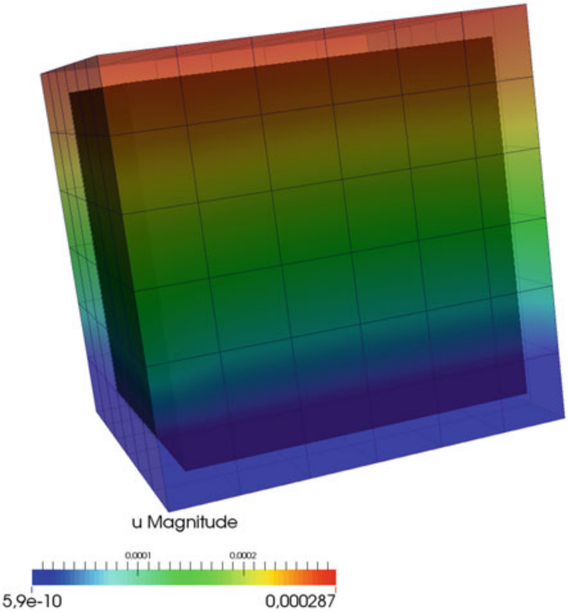
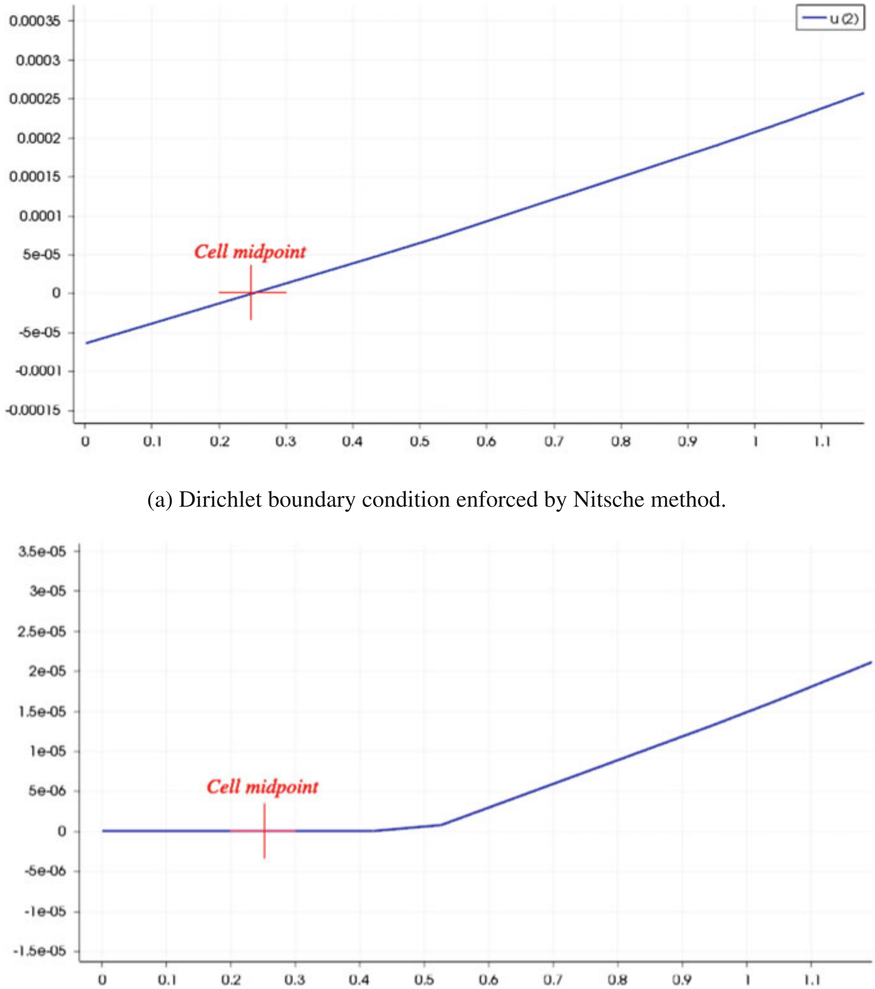


Fig. 5 Cube embedded within the hexahedral mesh

topology which is illustrated in Fig. 3a. Optimization domain was discretized using 400,000 hexahedral elements, and simulation was carried out in 10.32 s. For a comparison, in the work [16] for a similar test case of mid-plane loaded bridge with 113,000 degrees of freedom, the author reported execution time of 36.2 s when using GPU. With our solver we achieved better execution time for approximately 4 times higher mesh resolution.

Another bridge structure is shown in Fig. 3b. It was modelled using 465,000 hexahedral elements. The simulation was carried out in only 6.5 s.



(a) Dirichlet boundary condition enforced by Nitsche method.

(b) Strongly enforced Dirichlet boundary condition.

Fig. 6 Different approaches for enforcing boundary conditions at the interface of the elements cut by the cube. The vertical axis denotes the displacement along z axis. The horizontal axis denotes z coordinate of the cube hexahedral mesh

7.3 Cantilever

In this example we consider cantilever discretized using 740,000 hexahedral elements, and optimized in 13.8 s. The resulting topology is shown in Fig. 4. For a similar case of a loaded cantilever in work [2], discretized with 245,760 elements, authors reported solution time of approximately 3.9 h. In another variation of a loaded cantilever in work [16], the author reported computing time of 4 min for 324,000 elements when using GPU.

7.4 Dirichlet Boundary Conditions

We demonstrate the results with imposed zero Dirichlet boundary conditions by the Nitsche method on a cube object. For this test case the cube with dimensions 2.5 × 2.5 × 2.5 mm is discretized with 6 hexahedral elements in each direction and of side length 0.5 mm. Evenly distributed vertical forces are acting on the upper surface of the cube. As it can be seen in Fig. 5, the cube is embedded within the hexahedral mesh.

Zero Dirichlet boundary condition is imposed on embedded bottom surface of the cube. Figure 6a shows that zero displacement along z axis is located exactly at the midpoint of the hexahedral element when using the Nitsche method. On the other hand, when we use a strong method that assigns the same value to all the nodes of the boundary elements, displacement is constant across the whole hexahedral element as shown in Fig. 6b.

Thus, we observe that Nitsche method allows us to prescribe boundary conditions for boundary non-conforming meshes in a much more precise way compared to the strong method. As we have seen in the example where our mesh did not conform to the boundary, with the help of Nitsche method we were able to enforce zero displacement exactly at the intersection of the hexahedral elements and the Dirichlet boundary of the cube.

References

1. Bendsoe, M.P., Sigmund, O.: Topology Optimization: Theory, Methods and Applications. Springer Science & Business Media (2003)
2. Borrvall, T., Petersson, J.: Large-scale topology optimization in 3D using parallel computing. Comput. Methods Appl. Mech. Eng. **190**(46), 6201–6229 (2001)
3. von Danwitz, M.: Automated application of Dirichlet boundary conditions in voxel based analyses using the Finite Cell Method. Bachelor's thesis, Technical University of Munich (2013)
4. Deaton, J.D., Grandhi, R.V.: A survey of structural and multidisciplinary continuum topology optimization: post 2000. Struct. Multidiscip. Optim. **49**(1), 1–38 (2014)
5. Dick, C., Georgii, J., Westermann, R.: A real-time multigrid finite hexahedra method for elasticity simulation using CUDA. Simul. Model. Pract. Theory **19**(2), 801–816 (2011)
6. Embar, A., Dolbow, J., Harari, I.: Imposing Dirichlet boundary conditions with Nitsche's method and spline-based finite elements. Int. J. Numer. Methods Eng. **83**(7), 877–898 (2010)
7. Gausemeier J, Echterhoff, N., Kokoschka, M., Wall, M.: Thinking ahead the Future of Additive Manufacturing–Analysis of Promising Industries (2011)
8. Gavranovic, S.: Topology Optimization using GPGPU. Master's thesis, Technical University of Munich (2015)

9. General Electric jet engine bracket challange. https://grabcad.com/challenges/ge-jet-engine-bracket-challenge
10. Hackbusch, W.: Multi-grid methods and applications. In: Springer Series in Computational Mathematics. Springer (2003)
11. Nvidia CUDA: Compute unified device architecture programming guide (2014)
12. Open CASCADE library. http://www.opencascade.org/
13. Paulino, G.H.: Where are we in topology optimization? In: 10th World Congress on Structural and Multidisciplinary Optimization, Orlando, Florida (2013)
14. Schmidt, S., Schulz, V.: A 2589 line topology optimization code written for the graphics card. Comput. Vis. Sci. **14**(6), 249–256 (2011)
15. Sigmund, O.: A 99 line topology optimization code written in Matlab. Struct. Multidiscip. Optim. **21**(2), 120–127 (2001)
16. Suresh, K.: Efficient generation of large-scale pareto-optimal topologies. Struct. Multidiscip. Optim. **47**(1), 49–61 (2013)
17. Wohlers Associates: Wohlers Report 2014-3D Printing and Additive Manufacturing State of the Industry (2014)
18. Zander, N.: The Finite Cell Method for Linear Thermoelasticity. Master's thesis, Technical University of Munich (2011)