Chapter 9
MDO Related Issues: Multi-Objective and Mixed Continuous/Discrete Optimization

Loïc Brevault (✉), Julien Pelamatti, Ali Hebbal, Mathieu Balesdent, El-Ghazali Talbi, and Nouredine Melab
e-mail: loic.brevault@onera.fr

9.1 Introduction and Concepts

In addition to the multi-fidelity aspects in MDO discussed in Chapter 8, two additional topics of interest to solve complex MDO problems are discussed in this chapter: multi-objective MDO and mixed continuous/discrete variable design optimization problems.

Firstly, in Section 9.2, multi-objective problems in MDO are introduced. The design of complex aerospace systems often involves antagonist objectives that require to find trade-off among them to identify a set of interesting design candidates. Many aerospace design problems have competing objectives, for instance the optimization of a wing lift and drag in aerodynamics or a wing structural strength and weight. As a consequence, methodologies have been proposed that combine multidisciplinary and multi-objective optimization techniques. Some approaches consist in applying multi-objective optimization algorithms to the formally described MDO formulations (see Chapter 1). Alternatively, some strategies try to develop new MDO formulations to account for the decomposition of the process into several disciplines and several objectives. In Section 9.2, a focus on Bayesian Optimization (BO) for multi-objective is performed to solve optimization problems involving computationally expensive functions. An extension of BO for problems involving non-stationary phenomena is also discussed.

Secondly, in Section 9.3, the challenge of taking into account continuous, discrete, and categorical design variables in the MDO process is introduced. Within the framework of complex system design, it is often necessary to solve mixed variable optimization problems, in which the objective and constraint functions can depend simultaneously on continuous variables (e.g., structure sizing parameters, combustion chamber pressure) as well as on discrete variables (e.g., number of engines, number of rocket stages) and categorical variables (e.g., type of materials, choice of technologies). Due to the complexity and computational cost of this type

L. Brevault et al., *Aerospace System Analysis and Optimization in Uncertainty*,
Springer Optimization and Its Applications 156,
https://doi.org/10.1007/978-3-030-39126-3_9

of problems, optimization algorithms commonly used in the presence of discrete variables such as mixed variable Genetic Algorithm (Stelmack et al. 1998) and Mesh Adaptive Discrete Search (MADS) (Audet and Dennis Jr 2006) are inadequate, as they require a large number of function evaluations in order to converge. Alternative methodologies relying on BO have to be used to solve this type of problems and are presented in Section 9.3.

In order to tackle these two challenges, recent works focused on the use of Gaussian processes (Emmerich and Klinkenberg 2008; Pelamatti et al. 2018; Brevault et al. 2019; Hebbal et al. 2019). As presented in Chapter 3, Gaussian processes have interesting properties in terms of prediction and estimation of the associated prediction error that can be exploited for these MDO related issues. In this chapter, after an introduction of the issues associated with multi-objective MDO and mixed continuous/discrete optimization and a brief literature review of the existing works, an example of work using Gaussian process is presented for the field of Multi-objective and Mixed continuous/discrete optimization. One has to note that the design optimization problems considered in this chapter are deterministic. Moreover, only single discipline design problems are considered in this chapter.

9.2 Multi-Objective MDO

Aerospace vehicle design problems can ideally be modeled as multi-objective and multidisciplinary optimization problems. Different antagonist and conflicting objectives have to be considered in order to find an appropriate trade-off and identify interesting design candidates. Taking into account several objectives make the MDO problem solving more complex compared to single-objective MDO. For instance, for a launch vehicle design, a set of objectives to optimize for a single vehicle may be considered such as the Gross Lift-Off Weight (GLOW) and the payload mass to inject into orbit. Alternatively, multiple quantities of interest for a family of vehicles may be considered. Moreover, it is possible to design a vehicle for different missions, for example, a reusable mission (partially reusable first stage using toss-back, glide-back, or fly-back return strategies) and an expendable mission with the same first stage and additional solid boosters. This multi-mission (which can be defined as a multi-objective problem) offers the possibility to increase the flexibility and adaptability to different target missions.

In Arias-Montano et al. (2012), a rich taxonomy of the applications of multi-objective optimization in aerospace engineering is presented. For multi-objective MDO problems, the existing MDO approaches in the literature combine the multi-objective optimization algorithms with the existing MDO formulations presented in Chapter 1. Multi-level formulations are particularly suited to the multi-objective problems as they enable to decompose the problem according to the disciplines and the objectives. Kurapati and Azarm (2000) proposed an immune network system multi-objective genetic algorithm (MOGA) approach for MDO problems. For each subsystem of the hierarchically decomposed multidisciplinary system,

a MOGA which focuses on a specific set of design variables holds within the subsystem population representation. Gunawan et al. (2003) proposed a MDO method applicable to multi-objective optimization problems that can be decomposed hierarchically into multi-objective subproblems and of which the objective functions are either separable or additively separable. The method uses MOGA to optimize the subproblems. Within the CO framework (Tappeta and Renaud 1997) proposed a multi-objective MDO formulation, with the weighted sum technique to account for multiple objectives. Exploiting also the CO formulation, McAllister et al. (2005) used the linear physical programming to help the designers to express their preferences for conflicting objectives using physically meaningful parameters. Alternatively, multi-objective MDO integration methods have also been adapted to the CSSO decomposition. Zhang et al. (2008) proposed a novel integration of the adaptive weighted sum method within the CSSO framework to provide a relatively uniformly spaced, widely distributed Pareto front. Huang (2003) and Huang et al. (2007) developed the multi-objective Pareto CSSO in which each discipline has substantial control over its own objective function during the MDO process, while ensuring the responsibility for the constraint satisfaction in coupled subspaces. Parashar and Bloebaum (2006) extended the previous work by developing a Genetic Algorithm-based heuristic solution strategy for Multi-Objective CSSO approach, in which several non-dominated candidates (Pareto solutions) are generated at each cycle of the proposed algorithm, with subsequent update and refinement. Kang et al. (2014) proposed to solve multi-objective optimization problems using Quasi-Separable Decomposition (QSD) and Analytical Target Cascading (ATC) MDO formulation. The objective function of QSD may be viewed as a weighted sum of competing objectives of a multi-objective problem with equal weights. The authors proposed to solve the multi-objective optimization problem formulated as a quasi-separable MDO one using ATC. The aggregate objective function is decomposed by formulating a subproblem for each objective and using an ATC formulation to coordinate the candidate solution of the decomposed problem. Xiao et al. (2015) presented a new methodology for the multi-objective MDO problems in non-cooperative environments based on gene expression programming (Ferreira 2001) (GEP) and the use of Nash equilibrium in the game theory. In this approach, the GEP method is used as a surrogate model to build the approximate rational reaction sets of the players in the Nash model.

Within the framework of launch vehicle design, multi-objective MDO approaches have been proposed in the literature (Castellini and Lavagna 2012; Fazeley et al. 2016; Fujikawa et al. 2015). Castellini and Lavagna (2012) proposed a comparison of seven population-based algorithms (e.g. NSGA-II (Deb et al. 2000), MOPSO (Coello Coello and Lechuga 2002), PAES (Knowles and Corne 2000)) dealing with multi-objective problems and compared them on expendable launch vehicle design problems. Two types of problem have been considered, ascent trajectory optimization with a fixed launcher and launcher design optimization (with optimization of architectures: number of boosters, type of propulsion, type of engine cycle, etc.). In this work, MDF is carried out to couple the different disciplines and conceptual design models are used. Fazeley et al. (2016) proposed to compare MDF

and CO for the multi-objective optimization of a bi-propellant space propulsion system design for an expendable launcher. Non-Sorted Genetic Algorithm II algorithm is used to perform multi-objective optimization. In comparison, for this specific problem, the authors concluded that MDF required less calls to the disciplines than CO. Fujikawa et al. (2015) performed a conceptual design study for a Two-Stage-To-Orbit space plane with ethanol-fueled rocket-based combined cycle engine using a multi-objective MDO approach. AAO is employed in the proposed process. Three objectives are considered for the problem: the payload mass, the GLOW, and the take-off velocity. At each iteration, the multi-objective problem is transformed into its relevant single-objective problem via min-max goal programming, and is subsequently solved using a gradient-based optimization method (SQP) (Fujikawa et al. 2013). Kosugi et al. (2011) used a multi-objective genetic algorithm to design a hybrid rocket. The two considered objectives are the GLOW and the maximal reached altitude by the sounding rocket and MDF is carried out. Brevault et al. (2019) proposed a multi-level MDO approach to design aerospace vehicles addressing a multi-mission problem.

These multi-objective design problems may be mathematically formalized as an optimization problem characterized by q objectives optimized under m constraints in a n-dimensional design space:

$$\begin{aligned} \min \quad & \mathbf{y} = \mathbf{f}(\mathbf{z}) = \left[f_1(\mathbf{z}), \ldots, f_q(\mathbf{z}) \right] && (9.1) \\ \text{w.r.t.} \quad & \mathbf{z} \in \mathbb{R}^n \\ \text{s.t.} \quad & g_i(\mathbf{z}) \leq 0, \ i = 1, \ldots, m && (9.2) \\ & \mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max}, && (9.3) \end{aligned}$$

where $\mathbf{z} = (z_1, \ldots, z_n) \in \mathbb{Z} \in \mathbb{R}^n$, and $\mathbf{y} = (y_1, \ldots, y_q) \in \mathbb{Y} \in \mathbb{R}^q$ (here y stands for the objective function response (and not the coupling variables like the other chapters)). $\mathbf{z}$ is called the decision vector, $\mathbb{Z}$ the decision space, $\mathbf{y}$ the objective vector, $\mathbb{Y}$ the objective space.

To solve this type of problems Multi-Objective Evolutionary Algorithms (MOEAs) (Deb 2001) are classically used. Among the most popular MOEAs, NSGA-II (Non-dominated Sorting Genetic Algorithm II) (Deb et al. 2000) or SMPSO (Speed-constrained Multi-objective PSO) (Nebro et al. 2009) can be cited. These algorithms present the advantages of using a population-based search and diversity mechanisms making them less prone to be trapped in local minima. Moreover, the use of simple operators for crossover and mutation allows the handling of highly non-linear or non-differentiable functions. However, MOEAs tend to need a consequent number of evaluations to converge to the exact Pareto front which represents the set of solutions that are equivalently optimal with respect to the different objectives in the sense of the Pareto dominance. MOEAs are not suited for computationally expensive MDO problems, where the concern is to minimize the number of discipline evaluations.

To overcome this issue, Bayesian Optimization for multi-objective problems using Efficient Global Optimization (EGO (Jones et al. 1998)) have been proposed (Beume et al. 2007) by using new infill sampling criteria based on the concept of Pareto-Dominance as the Expected HyperVolume Improvement (Wagner et al. 2010). In the following, EGO with Gaussian process for multi-objective problems is briefly described.

9.2.1 *Efficient Global Optimization for Multi-Objective Problems*

A variety of Bayesian algorithms to solve multi-objective optimization has been proposed. These algorithms can be classified into the aggregation-based methods (using EGO on a weighted sum of objective functions) (Knowles 2006; Zhang et al. 2010) and the domination-based approaches (using new infill sampling criteria based on the concept of Pareto-dominance) (Emmerich et al. 2006; Svenson and Santner 2010). A focus on the dominance-based techniques is realized. Multi-objective EGO shares the same structure as standard EGO, with the difference that a surrogate model is built for each objective and constraint function, and an infill sampling criterion based on the concept of Pareto-dominance is used as for instance the Expected HyperVolume Improvement (EHVI) (Wagner et al. 2010).

EVHI is a derivation of the Expected Improvement (EI) described by Jones et al. (1998) to the multi-objective case. Considering firstly, an unconstrained multi-objective optimization and an initial DoE of size N, $\mathscr{X}_N = \left\{\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(N)}\right\}$ the input data set and the corresponding objective function responses $\mathscr{Y}_N = \left\{\mathbf{y}^{(1)} = \mathbf{f}\left(\mathbf{z}^{(1)}\right), \ldots, \mathbf{y}^{(N)} = \mathbf{f}\left(\mathbf{z}^{(N)}\right)\right\}$. Let $\mathbb{V} = \left\{\mathbf{y} \in \mathbb{R}^q \mid \mathbf{y}^L \leq \mathbf{y} \leq \mathbf{y}^U\right\}$ be a finite hypervolume of the objective space where all possible solutions lie, with $\mathbf{y}^L = \left[\min f_1(\mathbf{z}), \ldots, \min f_q(\mathbf{z})\right]$ the ideal objective candidate and $\mathbf{y}^U$ a chosen upper point (nadir point). The dominated hypervolume $\mathrm{H}_{\mathscr{Y}_N}$ of the DoE responses $\mathscr{Y}_N$ is defined as:

$$\mathrm{H}_{\mathscr{Y}_N} = \left\{\mathbf{y} \in \mathbb{V} \mid \exists i \in \{1, \ldots, N\}, \quad \mathbf{f}\left(\mathbf{z}^{(i)}\right) \prec \mathbf{y}\right\}. \tag{9.4}$$

$\mathrm{H}_{\mathscr{Y}_N}$ is a subset of $\mathbb{V}$ whose points are dominated by the DoE responses $\mathscr{Y}_N$ ($\prec$ refers to Pareto dominance). Considering a new candidate solution $\mathbf{z}^{(N+1)}$, since $\mathrm{H}_{\mathscr{Y}_N} \subseteq \mathrm{H}_{\mathscr{Y}_{N+1}}$, the hypervolume improvement by adding this candidate is given by $\mathrm{I}_N\left(\mathbf{z}^{(N+1)}\right) = |\mathrm{H}_{\mathscr{Y}_{N+1}}| - |\mathrm{H}_{\mathscr{Y}_N}|$. The notions introduced previously are illustrated for the two objective case in Figure 9.1.

Considering the set of GPs $\hat{\mathbf{Y}} = \left\{\hat{Y}_1 \sim \mathscr{N}(\hat{y}_1, \hat{\sigma}_1), \ldots, \hat{Y}_q \sim \mathscr{N}(\hat{y}_q, \hat{\sigma}_q)\right\}$ modeling the exact functions $f_1, \ldots, f_q$. The Expected HyperVolume Improvement for a candidate solution $\mathbf{z}$ is given by

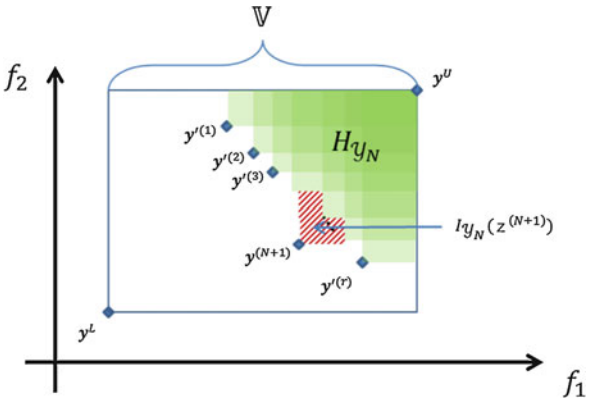


Fig. 9.1 Hypervolume improvement (hatched area) in a two objective case

$$\mathrm{EHVI}_{\mathscr{Y}_N}(\mathbf{z}) = \mathbb{E}\left[|\mathrm{H}_{\mathscr{Y}_{N+1}}| - |\mathrm{H}_{\mathscr{Y}_N}|\right] = \int_{\mathbb{V}\setminus \mathrm{H}_{\mathscr{Y}_N}} \mathbb{P}\left[\hat{\mathbf{Y}}(\mathbf{z}) \prec \mathbf{p}\right] \mathrm{d}\mathbf{p}. \tag{9.5}$$

For constrained optimization problems, a constrained infill criterion can be considered which combines EHVI with for instance the probability of feasibility introduced by Schonlau et al. (1998) or the expected violation of the constraints (Audet et al. 2000). The adaptation of the infill criterion to take into account the constraints in multi-objective EGO is similar to single-objective EGO (see Chapter 5). By multiplying this probability of feasibility $P_f(\cdot)$ to the Expected HyperVolume Improvement, the magnitude of the resulting quantity tends to zero in the design space areas where there is a low likelihood of constraint feasibility and it tends to the EHVI value where there is a high likelihood of constraint feasibility.

At each iteration of the Multi-Objective EGO (MO-EGO), this infill criterion $C(\mathbf{z}) = EHVI(\mathbf{z}) \times P_f(\mathbf{z})$ is maximized in order to identify the most promising candidate. This latter is evaluated on the exact objective and constraint functions and added to the DoE. All the surrogate models are updated and a new iteration can start if the convergence criterion is not satisfied. The different steps of MO-EGO are summarized in Figure 9.2.

The computation of EHVI for many objectives is a non-trivial problem. Several methods (Emmerich and Klinkenberg 2008; Bader and Zitzler 2011) have been proposed to calculate the EHVI formula, however, the complexity increases exponentially with the number of objectives.

In many design optimization problems, the objective functions or the constraints are non-stationary. Due to the abrupt change of a physical behavior depending on the design space location, the simulation response may vary with a different smoothness along the input space. GP is not adapted to predict these non-stationary functions since it relies on a stationary covariance function which implies a uniform smoothness of the prediction. To be able to approximate a non-stationary response, different methods have been proposed. A first category of methods consists in adapting existing covariance functions to non-stationary behaviors. For instance, the

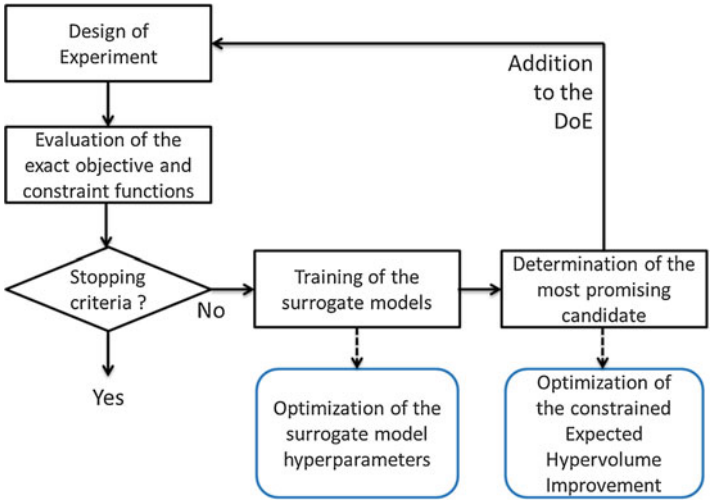


Fig. 9.2 Steps of Multi-objective EGO (MO-EGO)

squared exponential covariance function (Higdon et al. 1999) and the Matern covariance function (Paciorek and Schervish 2006) have been adapted for non-stationary modeling. However, these approaches are limited to low dimensional problems as the adaptation becomes intractable for higher dimensions. Alternatively, approaches combining multiple local stationary covariance functions have been proposed. Haas (1990) presented a moving window technique where the training and prediction regions move along the input space. Rasmussen and Ghahramani (2002) introduced different stationary GPs in different subspaces of the input space to account for the non-stationary behavior. An important issue with these approaches is their limitation of the size the data set for the GP training in a computationally expensive problem. Using a local surrogate model with sparser data may cause a poor approximation. Finally, non-linear mapping methods have been proposed. Xiong et al. (2007) introduced a piece-wise density function with parameterized knots to map the input space with a deformed space to model the non-stationary response by a stationary model. These techniques may not be adapted to high dimensional problems due to the complexity of the non-linear mapping to catch the function behavior.

Recently, to handle the non-stationary issue, a new class of surrogate models consisting of a functional composition of GPs called Deep Gaussian processes (DGPs) (Damianou and Lawrence 2013) has been developed. DGPs are interesting to model non-stationary functions when coupled with EGO for multi-objective problems (Hebbal et al. 2019). In many design optimization problems, the objective functions or the constraints may present non-stationary behavior. In fact, due to the abrupt change of some physical properties, the response may vary with a different smoothness along the input space. For example, in the structure discipline, the stress–strain curve of a material can be non-stationary i.e. with a different

trend in the elastic region, the strain hardening region and the necking region. In aerodynamics, computational fluid dynamics (CFD) problems often have different specific flow regimes due to separation zones, circulating flows, vortex bursts, transitions from subsonic to transonic, supersonic, and hypersonic conditions. GP regression may not be adapted to predict these non-stationary functions since it is based on a stationary covariance function which implies a uniform smoothness of the prediction. In the following section, a focus on DGP is done and its connection with Bayesian Optimization is discussed.

9.2.2 *Deep Gaussian Process for Multi-Objective Problems*

A DGP (Damianou and Lawrence 2013) is a deep architecture developed by the deep learning community where each layer is a GP. DGP takes the advantage of Deep Neural Network (LeCun et al. 2015) and Gaussian process to provide a multi-layer generalization of GP intrinsically enabling the modeling of complex (non-stationarity) function behavior while providing the modeling uncertainty. The layers are chained together by using the outputs of each layer as the inputs to the next layer. The statistical relationship between the inputs and the responses is expressed as a functional composition of several GPs. Each layer of DGP is composed of several input nodes and output nodes with GPs mapping between them. DGP may be formulated as Damianou and Lawrence (2013) (Figure 9.3):

$$y = f_L(\mathbf{f}_{L-1}(\ldots \mathbf{f}_l(\ldots(\mathbf{f}_1(\mathbf{f}_0(\mathbf{z}) + \epsilon_0) + \epsilon_1)\ldots) + \epsilon_l)\ldots + \epsilon_{L-1}) + \epsilon_L \tag{9.6}$$

with L the number of layers, $\mathbf{f}_l(\cdot)$ an intermediate GP and $\epsilon_l \sim \mathcal{N}(0, \sigma_l^2\mathbf{I})$ a Gaussian noise introduced in each layer. Each layer l is constituted of an input node $\mathbf{h}_l$, an output node $\mathbf{h}_{l+1}$ and a GP $\mathbf{f}_l(\cdot)$ connected the two nodes, leading to the

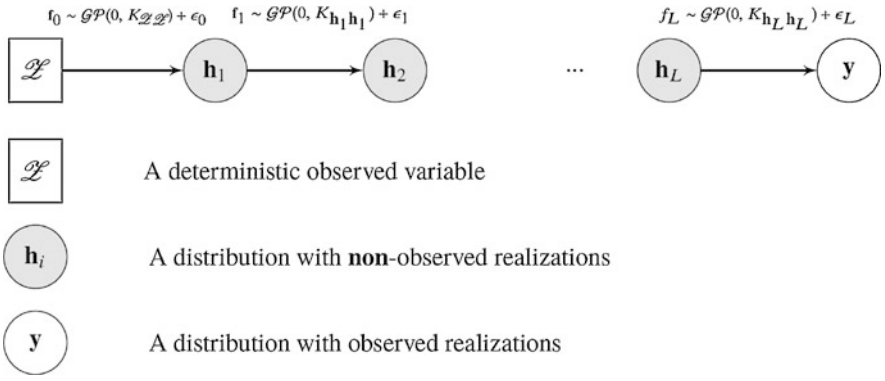


Fig. 9.3 A representation of the structure of a DGP

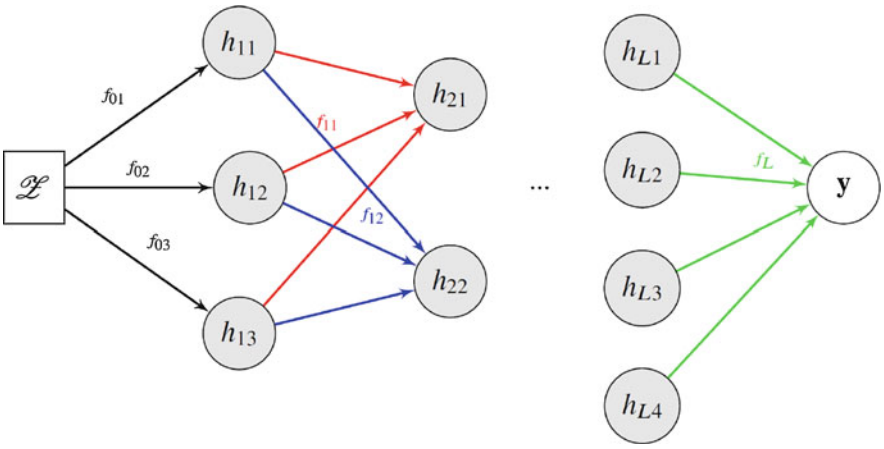


Fig. 9.4 Example of a decomposed view of the structure of a DGP

recursive equation: $\mathbf{h}_{l+1} = \mathbf{f}_l(\mathbf{h}_l) + \epsilon_l$. $\mathbf{h}_l$, $\mathbf{h}_{l+1}$ and $\mathbf{f}_l(\cdot)$ may be multidimensional and therefore for each component $h_{l+1,i}$ of $\mathbf{h}_{l+1}$, a GP $f_{li}(\cdot)$ maps between $\mathbf{h}_l$ and $h_{l+1,i}$ (Figure 9.4).

In order to train a DGP model, similarly as the GP regression (see Chapter 3 for more details), the marginal likelihood $p(\mathbf{y}|\mathscr{X})$ is maximized using an optimization algorithm (Equation 9.9).

$$p(\mathbf{y}|\mathscr{X}) = \int_{\mathbf{h}_1} \dots \int_{\mathbf{h}_l} \dots \int_{\mathbf{h}_L} p(\mathbf{y}, \mathbf{h}_1, \dots, \mathbf{h}_l, \dots, \mathbf{h}_l|\mathscr{X}) \, \mathrm{d}\mathbf{h}_1 \dots \mathrm{d}\mathbf{h}_L \quad (9.7)$$

$$= \int_{\{\mathbf{h}_l\}_1^L} p\left(\mathbf{y}, \{\mathbf{h}_l\}_1^L|\mathscr{X}\right) \mathrm{d}\{\mathbf{h}_l\}_1^L \quad (9.8)$$

$$= \int_{\{\mathbf{h}_l\}_1^L} p(\mathbf{y}|\mathbf{h}_L) p(\mathbf{h}_L|\mathbf{h}_{L-1}) \cdots p(\mathbf{h}_1|\mathscr{X}) \mathrm{d}\{\mathbf{h}_l\}_1^L$$

with $\{\mathbf{h}_l\}_1^L$ the set of hidden layers $\{\mathbf{h}_1, \dots, \mathbf{h}_L\}$.

However, at the difference with standard GPs, the intermediate nodes in DGP are latent variables, meaning that they are not observable, leading to the analytical computation of the marginal likelihood intractable. It is a consequence of the integration of the conditional probability $p(\mathbf{h}_{l+1}|\mathbf{h}_l)$ non-linearly involving the latent variable $\mathbf{h}_l$ inside the inverse of the covariance matrix $K_{\mathbf{h}_l\mathbf{h}_l} + \sigma_l^2\mathbf{I}$.

To bypass this problem, the marginal likelihood is approximated by a variational tractable lower bound which is optimized (Damianou and Lawrence 2013). To obtain this lower bound, two concepts are required. First, inducing variables are introduced at each layer. Inducing variables have been introduced in the context of sparse GP (Titsias 2009). It consists in augmenting by additional input-output pairs $\mathscr{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_M\}$ and $\mathbf{u} = f(\mathscr{X})$, the latent space, where $M << N$.

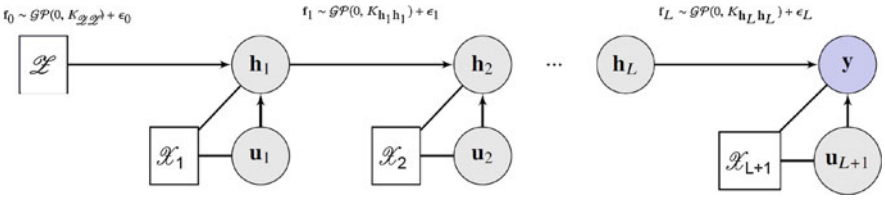


Fig. 9.5 Representation of the introduction of the inducing variables in DGPs

By introducing these variables, it avoids to compute the inverse of the covariance matrix of the whole data set $K_{\mathscr{Z},\mathscr{Z}} \in \mathscr{M}_{NN}$ and instead estimate the inverse of the covariance matrix of the inducing inputs $K_{\mathscr{X},\mathscr{X}} \in \mathscr{M}_{MM}$. This modification of standard GP allows to reduce the computational complexity in the training process and prediction of a sparse GP. In DGP, inducing variables $\mathscr{X}_l = \{\mathbf{x}_{l1}, \ldots, \mathbf{x}_{lM_l}\}$ and $\mathbf{u}_l = \mathbf{f}_l(\mathscr{X}_l)$ are introduced at each layer (Figure 9.5). Then, by marginalizing the variables $\{\mathbf{u}_l\}_1^L$, the marginal likelihood can be derived as:

$$p\left(\mathbf{y}|\mathscr{Z}\right) = \int_{\{\mathbf{h}_l,\mathbf{u}_l\}_1^L} p\left(\mathbf{y}, \{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L | \mathscr{Z}, \{\mathscr{X}_l\}_1^L\right) \mathrm{d}\{\mathbf{h}_l\}_1^L \mathrm{d}\{\mathbf{u}_l\}_1^L. \tag{9.9}$$

Secondly, the variational approach employed by Titsias and Lawrence (2010) for sparse GP can also be used in DGP. It consists in approximating the joint distribution of the true posterior of the latent variables $\mathbf{u}_l$ and $\mathbf{h}_l$ by multivariate Gaussian variational distributions $q(\mathbf{u}_l, \mathbf{h}_l)$ with the assumption of independency between the layers (Damianou and Lawrence 2013):

$$q\left(\{\mathbf{h}_l, \mathbf{u}_l\}_1^L\right) = \prod_{l=1}^{L} q(\mathbf{h}_l) q(\mathbf{u}_l).$$

This approximation of the posterior in the marginal loglikelihood $\log p(\mathbf{y}|\mathscr{Z})$ and the use of the Jensen's inequality allows to derive a variational lower bound on the marginal likelihood:

$$\begin{aligned}\log p(\mathbf{y}|\mathscr{Z}) &= \log \int_{\{\mathbf{h}_l,\mathbf{u}_l\}_1^L} \frac{q\left(\{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L\right)}{q\left(\{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L\right)} p\left(\mathbf{y}, \{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L | \mathscr{Z}, \{\mathscr{X}_l\}_1^L\right) \\ &\quad \mathrm{d}\{\mathbf{h}_l\}_1^L \mathrm{d}\{\mathbf{u}_l\}_1^L \\ &\geq \mathbb{E}_{q(\{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L)} \left[\log \frac{p\left(\mathbf{y}, \{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L | \mathscr{Z}, \{\mathscr{X}_l\}_1^L\right)}{q(\{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L)}\right] = \mathscr{L}. \end{aligned} \tag{9.10}$$

Using some developments from the variational sparse GP (Titsias 2009), an analytical tractable bound is obtained for kernels that are feasibly convoluted with

the Gaussian density, as for instance the Automatic Relevance Determination (ARD) exponential kernel. The analytical optimal form of $q(\mathbf{u}_l)$ as a function of $q(\mathbf{h}_l)$ is obtained *via* the derivative of the variational lower bound $\mathscr{L}$ with respect to $q(\mathbf{u}_l)$. By collapsing $q(\mathbf{u}_l)$ in the lower bound approximation by injecting its optimal form (derived from the gradient of the lower bound) it is possible to obtain a tighter lower bound depending on the following parameters:

- the kernel parameters: $\{\Theta_l\}_{l=1}^{l=L}$,
- the inducing inputs $\{\mathscr{X}_l\}_{l=1}^{l=L}$,
- the variational distributions parameters $\{q(\mathbf{h}_l) \sim \mathscr{N}(\mathbf{m}_l, \mathbf{S}_l)\}_{l=1}^{l=L}$.

Therefore, training a DGP model comes back to maximize the lower bound with respect to these parameters:

$$\begin{aligned} &\max \ \mathscr{L} \\ &\text{w.r.t. } \{\Theta_l\}_{l=1}^{l=L}, \{\mathscr{X}_l\}_{l=1}^{l=L}, \{\mathbf{m}_l\}_{l=1}^{l=L}, \{\mathbf{S}_l\}_{l=1}^{l=L}. \end{aligned}$$

The number of hyperparameters to optimize in the training of a DGP is more important than for a regular GP as only the kernel hyperparameters are considered. Different methods for training a DGP have been proposed. In Dai et al. (2015) instead of considering the hyperparameters of the variational posteriors $q(\mathbf{h}_l)$ as independent parameters, the authors considered them as a transformation of observed data $\mathscr{Y}$ through multi-layer perceptron. Bui et al. (2016) proposed a deterministic approximation for DGPs based on an approximated Expectation Propagation energy function, and a probabilistic back-propagation algorithm for learning. Salimbeni and Deisenroth (2017) developed the Doubly Stochastic technique in order to drop the assumption of independence between layers and the restriction to special form of kernels. Indeed, the posterior approximation maintains the exact model conditioned on $\mathbf{u}_l$:

$$q\left(\{\mathbf{h}_l, \mathbf{u}_l\}_1^L\right) = \prod_{l=1}^{L} p(\mathbf{h}_l|\mathbf{h}_{l-1}, \mathbf{u}_l) q(\mathbf{u}_l).$$

Nevertheless, this approach results in the loss of the analytical tractability of the lower bound $\mathscr{L}$. The variational lower bound is then rewritten as follows (the mention of the dependence on $\mathscr{Z}$ and $\mathscr{X}$ is omitted for the sake of simplicity):

$$\begin{aligned} \mathscr{L} &= \mathbb{E}_{q(\{\mathbf{h}_l, \mathbf{u}_l\}_1^L)} \left[\log \frac{p\left(\mathbf{y}, \{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L\right)}{q\left(\{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L\right)} \right] \\ &= \mathbb{E}_{q(\{\mathbf{h}_l, \mathbf{u}_l\}_1^L)} \left[\log \frac{p\left(\mathbf{y}|\{\mathbf{h}_l\}_1^L, \{\mathbf{u}_l\}_1^L\right) \prod_{l=1}^L p(\mathbf{h}_l|\mathbf{h}_{l-1}, \mathbf{u}_l) p(\mathbf{u}_l)}{\prod_{l=1}^L p(\mathbf{h}_l|\mathbf{h}_{l-1}, \mathbf{u}_l) q(\mathbf{u}_l)} \right] \end{aligned}$$

$$= \mathbb{E}_{q(\{\mathbf{h}_l, \mathbf{u}_l\}_1^L)} \left[\log \frac{\prod_{i=1}^{N} p\left(y^{(i)} | \mathbf{f}_L^{(i)}\right) \prod_{l=1}^{L} p(\mathbf{u}_l)}{\prod_{l=1}^{L} q(\mathbf{u}_l)} \right]$$

$$\mathscr{L} = \sum_{i=1}^{N} \mathbb{E}_{q\left(\mathbf{h}_L^{(i)}\right)} \left[\log p(y^{(i)} | \mathbf{h}_L^{(i)}) \right] - \sum_{l=1}^{L} KL\left[q(\mathbf{u}_l || p(\mathbf{u}_l)\right]. \tag{9.11}$$

With $KL[\dot{]}$ the Kullback-leibler divergence. This expression of the variational lower bound enables a factorization over the training data $\mathscr{X}, \mathscr{Y}$ offering the possibility to make parallel steps in the training. The expected value of this bound is estimated with Monte Carlo sampling using the propagation of each data sample $\mathbf{z}^{(i)}$ through all the GPs:

$$q(\mathbf{h}_L^{(i)}) = \int \prod_{l=1}^{L-1} q\left(\mathbf{h}_l^{(i)} | \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l, \mathbf{h}_{l-1}^{(i)}, \mathscr{Z}_{l-1}\right) \mathrm{d}\mathbf{h}_l^{(i)}$$

with $\mathbf{h}_0^{(i)} = \mathbf{z}^{(i)}$. The optimization of this formulation of the bound is done according to:

- the kernel parameters: $\{\Theta_l\}_{l=1}^{l=L}$,
- the inducing inputs $\{\mathscr{Z}_l\}_{l=1}^{l=L}$,
- the variational distributions of the inducing variables: $\{q(\mathbf{u}_l) \sim \mathscr{N}(\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)\}_{l=1}^{l=L}$.

The coupling of multi-objective and EGO with DGPs consists of a DGP surrogate model for the objective functions and/or the constraints that can be potentially non-stationary instead of a GP surrogate model. This coupling as in the single-objective EGO induces some challenges that need to be solved.

9.2.3 DGPs and Bayesian Optimization

The deep architecture of a DGP increases the model capability compared to a standard GP enabling to capture non-stationary phenomena (Figures 9.6 and 9.7). Therefore, the optimization of non-stationary functions with DGP is a promising framework. In fact, for single-objective optimization problems, experimentations in Hebbal et al. (2018) showed that Bayesian Optimization (BO) coupled with DGPs outperforms BO based on standard GPs and BO with non-linear mapping. For more details on BO with DGPs for single-objective optimization, please refer to Hebbal et al. (2019). This section is focused on multi-objective problems. Three concepts are essential for BO with DGPs:

- **Training approach:** different methods have been proposed to train DGPs as mentioned previously. In the first attempts to train DGPs in BO, the auto-encoded variational technique was used. However, in order to keep the dependency

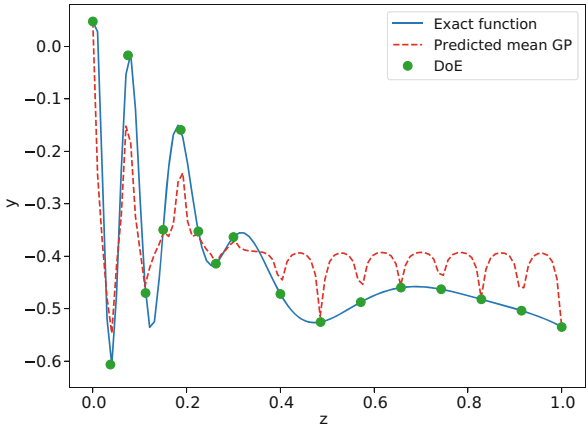


Fig. 9.6 Approximation of the modified-Xiong function by a regular GP. The model cannot capture the stability of the region [0.4, 1] and continues to oscillate

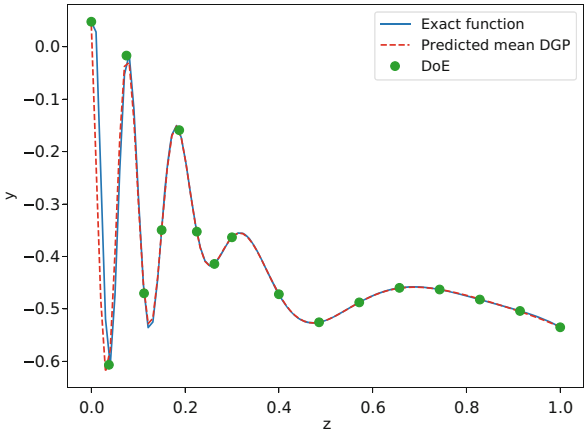


Fig. 9.7 Approximation of the modified-Xiong function by a DGP. The DGP model appropriately capture the two regions with different smoothness

between layers and increasing the robustness of the training, Hebbal et al. (2018) used the doubly stochastic variational approach. The experimental results of BO with DGPs using this type of training confirms this choice as it results in more robust results with respect to the initial training set especially when the architecture of the DGP gets deeper.

- **Infill criterion:** in single-objective BO with standard GPs, infill criteria such as the Expected Improvement, the Probability of Improvement, or the Expected Violation are computed using closed analytic formulae. These formulae are obtained based on the Gaussian distribution of the Gaussian process prediction. However, in DGPs the overall process prediction is no longer Gaussian. Thus, in order to use a valid approximation of the infill criteria, it is necessary to approximate the distribution of the prediction by a Gaussian distribution, and if not, to use a

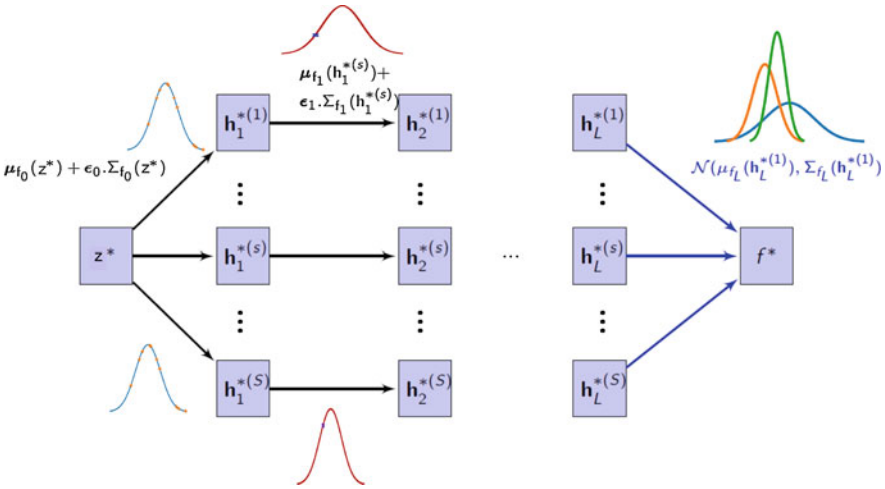


Fig. 9.8 The approximation of the prediction of a DGP model by a mixture of Gaussian distribution. S samples are drawn from the first layer, then, each sample is propagated through the whole network, with a realization at each hidden layer, until reaching the final layer where the mean and the variance of the final GP are considered for each sample. Thus, the prediction is approximated by a Gaussian mixture of the S samples

sampling technique for the prediction (Hebbal et al. 2018). In the multi-objective case the closed form analytic equation of the EHVI in Equation 9.5 is also derived with the assumption that the prediction of the objective functions follows a Gaussian distribution. Hence, the same approximations in the prediction used for the EI are necessary for the EHVI (Figure 9.8).

- **Configuration of the architecture:** the architecture of the DGPs concerns the number of layers, the number of hidden units at each layer and the number of inducing variables at each layer. DGPs tend to perform better (in terms of prediction accuracy and variability) when getting deeper as experimentally observed in Hebbal et al. (2018). The configuration of the architecture directly influences the computational complexity of the evaluation of the lower bound $\mathscr{L}$ given by $O(N(M_1^2 D_1 + \cdots + M_l^2 D_l + \cdots + M_L^2 D_L))$, where N is the size of the data set, L is the number of layers, M_l is the number of inducing inputs at the layer l and D_l is the number of hidden units at layer l. This is more expensive in the multi-objective case when multiple objectives have to be approximated. Therefore, a trade-off between the prediction performance and the training computational cost has to be found. Moreover, the particularity of using DGPs in a BO framework is that the number of data points changes at each iteration. Thus, the configuration of the architecture has to be adapted to the current iteration. In fact, in the early iterations when the datasize is small a simple architecture (a standard GP, a 1-layer DGP) is sufficient. Then, along the evolution of the size of the data set a more complex architecture can be developed. If the stationary behavior is known a priori for some objective

functions or constraints, one can use only GPs for some functions while using DGPs for the unknown or non-stationary functions.

9.2.4 Application to Toy Case

The challenge in EGO using DGP is the choice of the architecture for the DGP in terms of number of layers, number of nodes and number of the inducing variables. This promising area has to be further explored to be applicable to complex aerospace vehicle design.

In this section, experimentations on an analytical test problem are performed to compare between standard EGO, NSGA II, and EGO using DGPs. The analytical test case is a two objective problem with a non-stationary constraint. The problem (P_1) has been inspired by the TNK test problem (Deb et al. 2001) with a modification on the constraint making it non-stationary. In fact, there are two regions one where the function varies with a high frequency and the other where the function varies slowly (Figure 9.9).

$$P_1 \left| \begin{array}{l} \min \ f_1(\mathbf{z}) = -z^{(1)} \\ \min \ f_2(\mathbf{z}) = -z^{(2)} \\ \text{s.t} \ g_1(\mathbf{z}) = 0.5{z^{(1)}}^2 + 0.5{z^{(2)}}^2 - 0.2 \times \cos\left(20 \times \arctan\left(0.3\frac{z^{(1)}}{z^{(2)}}\right)\right) \leq 0 \\ \qquad \mathbf{z} = \left[z^{(1)}, z^{(2)}\right] \\ \qquad 0 < z^{(1)} < 1 \\ \qquad 0 < z^{(2)} < 1 \end{array} \right. \tag{9.12}$$

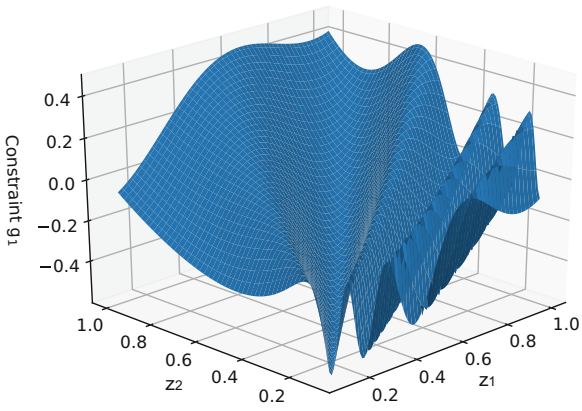


Fig. 9.9 Constraint function

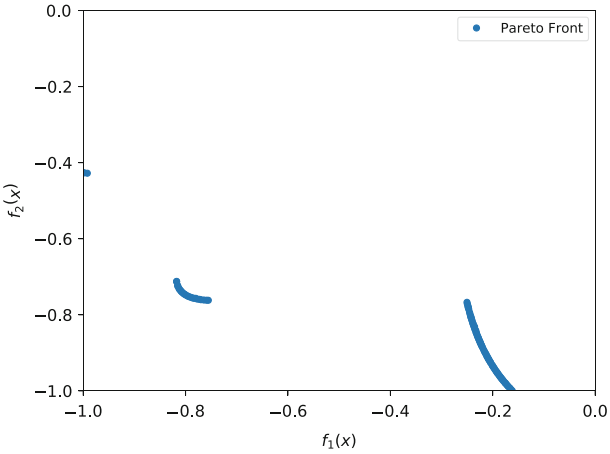
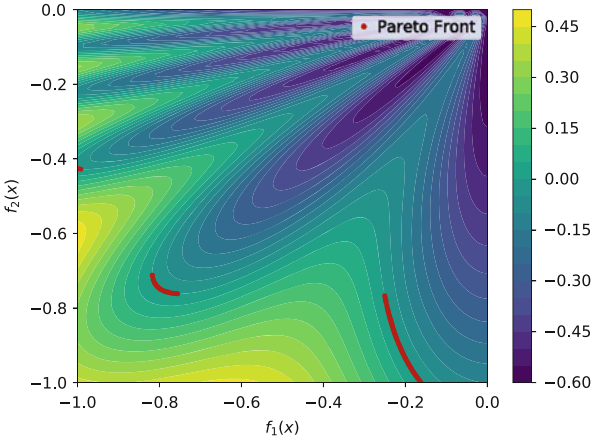


Fig. 9.10 Exact Pareto Front

Fig. 9.11 Exact Pareto Front with constraint contour plot



The Pareto front given by this problem has three separated regions (Figures 9.10 and 9.11). The value of the hypervolume dominated in the rectangle $[[-1, -1], [0, 0]]$ is 0.752.

A population of five individuals is used for NSGA-II and it is run until 45 evaluations are reached. For standard EGO (using standard GP) and for EGO with DGPs, 25 initial points are generated using a Latin Hypercube Sampling and 25 points are added using the EHVI with the probability of feasibility optimized with a Differential Evolution algorithm (Qin et al. 2009). To evaluate the robustness of each algorithm the experimentation is repeated for 10 different initial DoEs.

- For NSGA-II, a simulated binary crossover is used, with a distribution index of 15, a probability of 0.9, and a polynomial mutation with a distribution index

of 20 and a probability of 1/6. The constraint dominance is used to handle the constraints.

- In standard EGO, an Automatic Relevance Determination (ARD) exponential kernel (Jones et al. 1998) is used: $k(\mathbf{z}, \mathbf{z}') = \exp\{-\sum_{i=1}^{n} \theta_i \left(z^{(i)} - z'^{(i)}\right)^2\}$.
- In EGO with DGP, only the constraint is approximated by a DGP since the objective functions are stationary. An ARD Gaussian kernel is used in each layer. The training of the DGP is performed using the Doubly Stochastic training approach (Salimbeni and Deisenroth 2017). Configurations with 1, 2, and 3 layers are tested with a number of inducing inputs equal to the data set size (it evolves along the iteration of EGO+DGP). The number of units in the hidden layers is fixed to six. The prediction of the DGP is approximated with 500 samples (Figure 9.12).

Table 9.1 displays the best hypervolume value, the worst hypervolume value, the mean hypervolume and its standard deviation for each algorithm ("MO-DEGO l HL qD" corresponds to MO-EGO with a DGP of l-hidden q-dimensional layers and a number of inducing inputs that is equal to the size of the data set at each iteration). Figure 9.13 gives the Pareto front of each algorithm in all ten repetitions.

As expected NSGA-II is the less efficient algorithm. It needs more evaluations to converge to the optimal Pareto front and with only 45 evaluations, NSGA-II is far from convergence, which explains the high dispersion of the Pareto fronts according to the repetitions. BO with standard GP gives valuable results for some

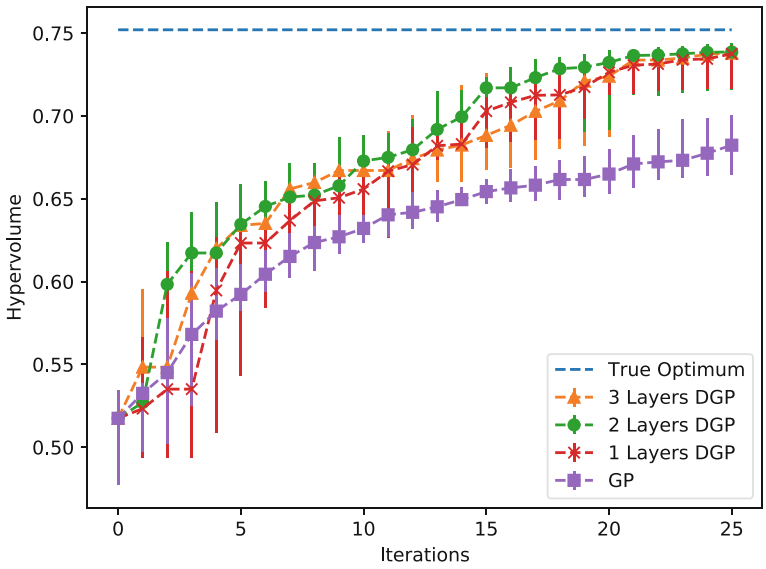
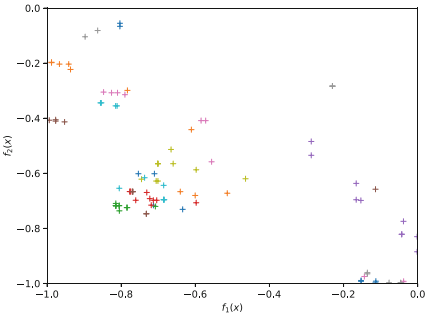
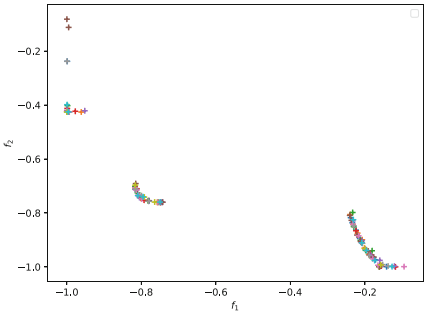


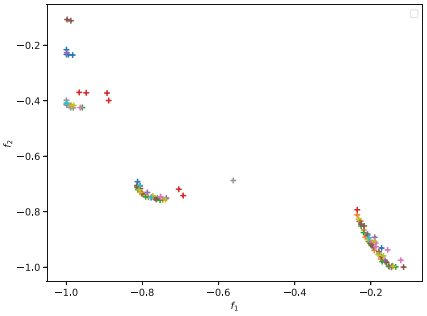
Fig. 9.12 Convergence plot of BO with different architectures of DGPs and a regular GP. The markers indicate the median of the hypervolume obtained while the errorbars indicate the first and the third quartiles



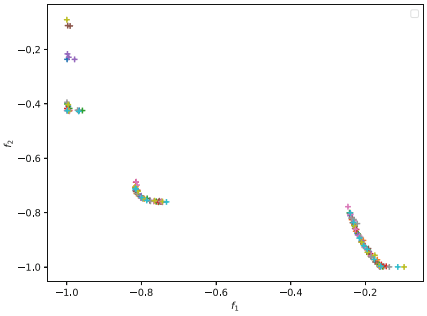
NSGA-II Pareto Fronts



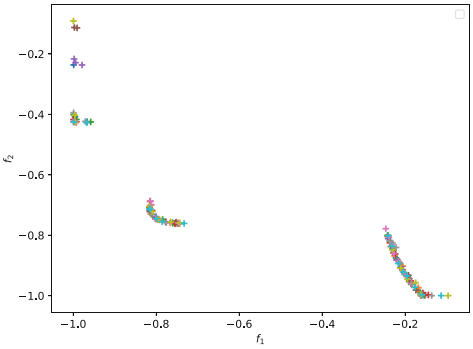
DGP 1HL Pareto Fronts



Standard EGO Pareto Fronts



DGP 2HL Pareto Fronts



DGP 3HL Pareto Fronts

Fig. 9.13 Pareto fronts of the different repetitions for each algorithm. Each repetition corresponds to a certain color

repetitions, however it has an important variability among the repetitions. This behavior is explained by the fact that the initial DoE for the worst repetitions is concentrated in the region of high frequency and so the Gaussian process cannot capture the region of low frequency and vice versa. BO with DGPs performs better than standard GP regardless of the number of layers considered in the architecture. It is also more robust to the initial DoE as shown in Figure 9.13 where each repetition reaches the exact Pareto front. The convergence plot of the different BO shows a separation between BO with GP and with DGPs, and this can be noticed since the early iterations. The trade-off between computational complexity in the training of a DGP and the capability of representation is important to be considered. In fact, in this problem there is no clear difference between the three considered configurations of DGPs. Hence, the capacity of the DGP with only one layer is sufficient to capture the non-stationarity of this problem, and there is no need to go deeper.

In the context of Bayesian Optimization using DGP, open issues concern the development of an adaptive framework for the configuration of the DGP according to the problem at hand. Also the training duration of the DGP can be problematic with complex models, an interesting direction of research is to investigate ways to accelerate the training process. Finally, here in the multi-objective case the objectives were considered independent, one may gain some information by creating a dependence between the objectives using the concept of multi-output GPs and co-regionalization.

9.3 Mixed Continuous/Discrete MDO

In complex engineering system, the early stages of design involve making numerous decisions that ultimately define the final architecture of the system. These choices include the technology selection for all the subsystems and components, the operational modes, etc. For instance, in the design of a launch vehicle, there are propulsion type choices (liquid bi-propellant, solid propellant, hybrid engines, etc.), architectural choices (number of stages, presence of booster or not, number of engines, etc.), or thermal control operational modes (active, passive). These selections form an architecture design space of discrete variables. In addition, each architecture and technology selection has different configuration corresponding to

Table 9.1 Performance of the algorithms

Algorithm	Hypervolume median	First quartile Hypervolume	Third quartile Hypervolume
NSGA-II	0.485	0.186	0.664
MO-BO GP	0.682	0.664	0.700
MO-BO DGP 1HL	0.737	0.716	0.743
MO-BO DGP 2HL	0.738	0.715	0.744
MO-BO DGP 3HL	0.739	0.726	0.741

a degree of variation involving continuous decision (propellant masses, structural sizing parameters, trajectory command controls, etc.) that further compound the decision space. This design problem can be formulated by solving an optimization problem involving objective and constraint functions which depend simultaneously on continuous and discrete decision variables. These discrete variables can be described by either integer values (i.e., 1,2,3) or categorical variables (e.g., materials, type of propellants).

For complex system design in the early design phases, a systematic methodology that enables both the exploration of a large combinatorial design space and also supports quantitative trade-off analyses to facilitate the selection of a design is necessary. The challenge is the associated computational cost of such a methodology. In the literature, three different types of approaches in the conceptual design phase may be identified: the baseline design process (Mavris and Kirby 1999; Soban and Mavris 2013), the architecture optimization methods (Villeneuve and Mavris 2005; Armstrong et al. 2008) and the architecture comparison approaches (Donahue 2001; Walton and Hastings 2004; Kothari and Webber 2010; Prasadh et al. 2014). In the first category, only few configurations around a single architecture baseline are considered to infuse new technologies and meet new requirements. However, the simulations and analyses are accurate and detailed (using CDF and FEA simulations for instance). These design process approaches are classical for civil transport aircraft design using existing aircraft as baseline and improving aerostructural design for instance. The architecture comparison process caries out a trade-off between many architectures but with a level of analysis and simulation very limited often based on qualitative knowledge and expert opinion. Moreover, for each architecture considered, only a single configuration is studied, there is no optimization of the continuous design variables. Compared to the baseline design process, the architecture optimization approaches consider fewer architecture baselines but carry out separate optimizations before performing a trade-off between the optimal configurations of each architecture. Therefore, even if less architectures are considered, the simulation and modeling are more accurate due to the need to find the optimal configuration for each architecture. The architecture optimization methods are a trade-off between the two other alternative approaches. They are less accurate than the baseline design process but explore more the architecture space, and are more accurate than the architecture comparison processes but less architectures are considered.

Only a limited number of works focused on directly optimizing both the types of architectures and the continuous variables describing them. Frank et al. (2016) and Frank (2016) proposed an evolutionary multi-architecture optimization algorithm to enable designers to generate variable-oriented architectures that can be further optimized and compared. Three general steps are considered in the proposed approach, with firstly the generation of alternative concepts and the associated architectures, secondly a sequential process of optimization of each architecture and comparison across the architectures, and finally a selection of the best architecture and associated optimal configuration. The proposed methodology lies within the

architecture optimization methods but stay limited in terms of modeling to low-fidelity model and simplified analyses.

Chepko et al. (2008) and Chepko (2009) proposed a methodology to enable the architecture selection along with the configuration optimization in a general framework. Firstly, a functional decomposition of the considered system is performed while listing the different technological options for the different components and using logical Boolean functions ("AND"/"OR") to establish the set of the possible architectures. Then, depending on the size of the architecture set, if it is small enough then each problem is solved by running an optimization on the continuous variables for each architecture and then a comparison of the results is performed. In case the architecture space is too large to consider all the combinations, a nested optimization problem is set in which the discrete design variables are controlled at the outer-level and the continuous variables are optimized at the inner-level. Alternatively, instead of a nested optimization problem, Chepko et al. also proposed to solve a single-level search with a decision space combining discrete and continuous search spaces.

Some classical optimization algorithms used in the presence of both discrete and continuous variables such as mixed variable Genetic Algorithm (Stelmack et al. 1998) and Mesh Adaptive Discrete Search (MADS) have been adapted to solve these mixed optimization problems. Stelmack et al. (1998) applied a mixed Genetic Algorithm to a MDO problem solving using a Concurrent Sub-Space Optimization (CSSO) formulation. Haftka et al. extended the Quasi-Separable Decomposition (QSD) (Haftka and Watson 2006) MDO formulation to account for discrete variables in addition to continuous ones. Quasi-Separable optimization problems enable the authors to propose a rigorous decomposition theory. Barjhoux et al. (2018) proposed a bi-level methodology to solve structural optimization problem with both continuous and categorical variables. The methodology relies on the separation of the continuous variables from the categorical ones that are treated in two different optimization problems: an inner-level problem in which the continuous sizing variables are optimized at given categorical choices, and an outer-level problem that handles the categorical variables formulated as integers. Aliakbargolkar et al. (2013) combined genetic algorithm and sequential quadratic programming to solve the optimization problem of launch vehicle transportation architectures using simplified modeling. For the previous mentioned studies, the employed optimization algorithms are able to solve mixed continuous and discrete optimization problems but require a large number of function evaluations in order to converge.

To alleviate this problem, surrogate model-based design optimization (SMBDO) (Queipo et al. 2005) may be employed. A few surrogate modeling techniques for functions depending on both continuous and discrete variables exist in the literature (Qian et al. 2008; Swiler et al. 2014; Zhang and Notz 2015). However, only few works extended the use of these mixed variable surrogate models for SMBDO for continuous/discrete optimization problems. For instance, several variants of radial basis functions (RBF) based SMDBO techniques for constrained mixed continuous/integer problems have been proposed (Beauthier et al. 2014; Holmström

et al. 2008; Müller et al. 2013; Rashid et al. 2008; Regis 2014). A surrogate model assisted GA is discussed in Bajer and Holena (2013) where both Generalized Linear Models (GLM) and RBF are considered. Roy et al. (2017) proposed to solve an aircraft design optimization problem along with an airline allocation problem in order to capture the coupling that exists between the aircraft design and the airline allocation disciplines. The authors proposed an algorithm combining branch and bound, Efficient Global Optimization, Kriging Partial Least Squares, and gradient-based optimization. The main idea of this framework is to leverage the efficiency of a gradient-based optimizer while globally exploring the integer design space with the help of the Efficient Global Optimization (EGO).

Two drawbacks can be outlined within the existing mixed variable SMDBO techniques mentioned above. Most of these techniques only deal with integer variables and cannot control generic discrete design variables such as categorical variables (e.g., type of materials). Furthermore, the handling of the constraints in the majority of the previously mentioned optimization methods relies on direct penalization of the objective function values for solutions that are not feasible. Although popular, this approach is inadequate when confronted with expensive computations and usually requires large numbers of function evaluations.

Mixed-variable optimization problems involve continuous and discrete variables. Continuous variables (e.g., geometrical sizing parameters, trajectory command law) refer to real numbers usually defined within given bounds. In their generic form, discrete variables are non-relaxable variables defined within a finite set of choices. Within the context of the design of complex system, they may represent technological choices such as architectural configurations (number of boosters, number of stages for launch vehicle), material alternatives and can be either ordinal or nominal (Agresti 1996). Ordinal category includes variables with an existing notion of order (e.g., integer variables, 'small/medium/large' types of decision variables), while the nominal one includes unordered decision variables for which no concept of metrics can be defined (e.g., propulsion cycle, materials). Although discrete variables often lack a conceptual numerical representation, it is common practice to assign an integer value to every considered alternatives in order to be able to include the related choices in the numerical optimization. For the sake of generality, no distinction between nominal and ordered discrete variables is made in the rest of the section.

9.3.1 *Mixed Continuous and Discrete Variable Optimization Problem*

In the following of this section, a general mixed variable optimization problem is formulated as follows (only one single objective is considered in this section):

$$\min \quad f(\mathbf{z}, \mathbf{x}) \qquad f : \mathscr{F}_z \times \mathscr{F}_x \rightarrow \mathscr{F}_y \subseteq \mathbb{R} \tag{9.13}$$

$$\begin{aligned} \text{w.r.t.} \quad & \mathbf{z} \in \mathscr{F}_z \subseteq \mathbb{R}^n \\ & \mathbf{x} \in \mathscr{F}_x \\ \text{s.t.} \quad & g_i(\mathbf{z}, \mathbf{x}) \leq 0 \qquad g_i : \mathscr{F}_z \times \mathscr{F}_x \rightarrow \mathscr{F}_{g_i} \subseteq \mathbb{R} \\ & \text{for } i = 1, \ldots, m, \end{aligned} \tag{9.14}$$

where $f(\cdot)$ is the objective function defined over the co-domain $\mathscr{F}_y$. $g_i(\cdot)$ is one of the m constraints of the problem in the co-domain $\mathscr{F}_{g_i}$ and $\mathbf{z} = \left(z^{(1)}, \ldots, z^{(n)}\right)$ is the vector containing the continuous decision variables and $\mathbf{x} = \left(x^{(1)}, \ldots, x^{(r)}\right)$ is the vector containing the discrete decision variables. n and r being the size of the continuous and discrete dimensions of the functions $f(\cdot)$ and $g(\cdot)$. For the sake of simplicity, the input vector containing both continuous and discrete variables is represented with the following notation: $\mathbf{w} = \{\mathbf{z}, \mathbf{x}\}$. Each discrete variable $x^{(j)}$ is characterized by b_j possible values, also known as *levels*, which therefore results in a total number of categorical combinations, or categories, of $d = \prod_{k=1}^{k=r} b_k$.

To solve mixed continuous/discrete variable optimization problems, a recent work proposed to combine Gaussian process using suited mixed variable kernels with Efficient Global Optimization algorithm (Pelamatti et al. 2018). A brief overview of this approach is provided in the next section with an illustration on a toy case problem.

9.3.2 Gaussian Process for Mixed Continuous/Discrete Variables

As presented in Chapter 3, Gaussian processes are mainly characterized by a covariance function involving a kernel. A valid kernel is necessary for the covariance function which must be symmetric and positive semi-definite (Santner et al. 2003). The product between valid kernels also results in a valid one (Shawe-Taylor and Cristianini 2004). This can be derived from the fact that according to the Schur product theorem (Davis 1962), the Hadamard product between two positive semi-definite matrices results in a positive semi-definite matrix. Therefore, in practice a valid mixed variable kernels may be obtained by combining kernels defined in the continuous n-dimensional space and kernels defined in the discrete r-dimensional space (Roustant et al. 2018):

$$k\left(\mathbf{w}_i, \mathbf{w}_j\right) = k_c\left(\mathbf{z}_i, \mathbf{z}_j\right) * k_d(\mathbf{x}_i, \mathbf{x}_j\right) \tag{9.15}$$

with k_c and k_d corresponding to the continuous and discrete kernels, respectively, and $*$ is a generic operator allowing to combine kernels. Such standard operators can be the product, the sum or the ANOVA between the two kernels (Roustant et al. 2018). Kernels computed on continuous and discrete variables can be combined

without any loss of generality or applicability in order to obtain the covariance matrix of a multivariate Gaussian process. Considering the most generic case, the resulting kernel is heteroscedastic, i.e., characterized by a different GP variance for each discrete category of the problem.

Within a generic approach, the discrete kernel k_d can be represented by a $d \times d$ positive semi-definite matrix $\mathbf{T}$. Each element $\mathbf{T}_{k,s}$ contains the covariance between two generic discrete categories k and s of the modeled function. A suitable parameterization of $\mathbf{T}$ is required to ensure the kernel validity. The Cholesky decomposition of the covariance matrix (Pinheiro and Bates 1996) is often considered in the GP inference (see Chapter 3):

$$\mathbf{T} = \mathbf{L}\mathbf{L}^T \tag{9.16}$$

in which $\mathbf{L}$ is a lower triangular matrix. Zhou et al. (2011) proposed to build the matrix $\mathbf{L}$ using an hypersphere decomposition (Rebonato and Jaeckel 2011) to define a mixed GP for modeling purpose (GP regression). The elements of the kth row of the matrix represent the coordinates of a point on the surface of a k-dimensional hypersphere. The triangular matrix elements $l_{k,s}$ are defined such that:

$$\begin{cases} l_{1,1} = \alpha_{1,0} & \\ l_{2,1} = \alpha_{2,0}\cos(\alpha_{2,1}) & \\ l_{2,2} = \alpha_{2,0}\sin(\alpha_{2,1}) & \\ l_{k,1} = \alpha_{k,0}\cos(\alpha_{k,1}) & \text{with } k > 2 \\ l_{k,s} = \alpha_{k,0}\sin(\alpha_{k,1})\ldots\sin(\alpha_{k,s-1})\cos(\alpha_{k,s}) & \text{with } k > 2 \text{ and } s = 2,\ldots,k-1 \\ l_{k,k} = \alpha_{k,0}\sin(\alpha_{k,1})\ldots\sin(\alpha_{k,k-2})\sin(\alpha_{k,k-1}) & \text{with } k > 2, \end{cases} \tag{9.17}$$

where $\alpha_{k,0} > 0$ and $\alpha_{k,s} \in (0, \pi)$ (for $s \neq 0$) are the hyperparameters characterizing the covariance between the various discrete categories of the modeled function. Due to the symmetrical nature of the covariance, $(d+1)d/2$ parameters $\alpha_{k,s}$ are needed to define $\mathbf{L}$.

For problems with a large number of discrete categories, the number of hyperparameters becomes considerably large, making the GP training difficult (Zhou et al. 2011). Furthermore, for the GP training, the data set has to contain samples belonging to every combination of discrete categories. However, it may happen that some combinations of discrete design variables are not physically feasible or cannot be modeled, in which case this parameterization cannot be applied. An example of this problem can be found in the modeling of a rocket engine propulsive performance (Pelamatti et al. 2019), in which not all the combinations of the propellant types are feasible for a combustion process.

In order to avoid these issues, alternative parameterizations which do not require all the discrete categories to be present within the data set and which are characterized by smaller numbers of hyperparameters are discussed in the following

paragraphs. To reduce the number of hyperparameters required to describe the covariance matrix, a first choice that can be made is to represent the discrete kernel as a combination of 1-dimensional discrete kernels. Each 1-dimensional kernel can be rewritten under the form of a positive semi-definite matrix. In case a kernel-wise product is considered, the resulting discrete kernel can be defined as Roustant et al. (2018):

$$k_d(\mathbf{x}_i, \mathbf{x}_j) = \prod_{s=1}^{r} [\mathbf{T}_s]_{x_i^{(s)}, x_j^{(s)}}, \tag{9.18}$$

where each matrix $\mathbf{T}_s$ contains the values of the covariance between the various levels of the generic discrete variable s.

Heteroscedastic Dimension-Wise Hypersphere Decomposition

A first parameterization for $\mathbf{T}_s$ has been introduced by Zhou et al. (2011) and uses the hypersphere decomposition. The proposed covariance is defined dimension-wise instead of category-wise. The hyperparameter number required to characterize the r matrices $\mathbf{T}_s$ is equal to $\sum_{k=1}^{k=r} b_k(b_k + 1)/2$. The dimension-wise variant offers a better scaling with the discrete dimension compared to the complete hypersphere decomposition, but as a trade-off, it theoretically provides a less accurate modeling of the correlation between the various discrete categories (Pelamatti et al. 2019). Moreover, it is not necessary for all the problem categories to be represented in the training data set as the kernel is defined only dimension-wise.

Homoscedastic Dimension-Wise Hypersphere Decomposition

The assumption of homoscedasticity can be made to further reduce the number of hyperparameters meaning that all the categories of the problem are characterized by the same variance value. In this case, each matrix $\mathbf{T}_s$ has a constant diagonal value. Consequently, the discrete kernel can be rewritten as a product between the common GP variances and r dimension-wise correlation matrices parameterized with the help of the same hypersphere decomposition. Due to the fact that correlation matrices are characterized by a unit diagonal, their hypersphere decomposition only requires $\sum_{k=1}^{k=r} b_k(b_k - 1)/2$ hyperparameters (i.e., all the hyperparameters $\alpha_{i,0}$ are equal to 1). Nevertheless, the assumption of homoscedasticity can introduce a large modeling error when dealing with categories that present different behaviors, and might therefore not always be valid.

Compound Symmetry Parameterization

In case an additional reduction in the hyperparameter number characterizing the discrete kernel is required, the Compound Symmetry (CS) (Pinheiro and Bates 2009) may be considered. In this case, each matrix $\mathbf{T}_s$ is represented by a single value of covariance c_s and a single value of variance v_s:

$$[\mathbf{T}_s]_{x_i^{(s)},x_j^{(s)}} = \begin{cases} v_s & \text{if} \quad x_i^{(s)} = x_j^{(s)} \\ c_s & \text{if} \quad x_i^{(s)} \neq x_j^{(s)} \end{cases} \tag{9.19}$$

with $-(b_s + 1)^{-1} v_s < c_s < v_s$, in order to ensure that $\mathbf{T}_s$ is positive semi-definite. A particular case of CS parameterization can be obtained by considering the covariance in the mixed continuous discrete search space to be spatially dependent as a function of the Gower distance, as is proposed by Halstrup (2016). In the Gower distance (Gower 1971), the coordinates of the two samples that are being considered are compared dimension-wise. For the continuous dimensions, the distance is proportional to the Manhattan distance, while for the discrete dimensions the distance is a weighted binary value which depends on the similarity between the variable values. In practice, the Gower distance between two candidates can be calculated by:

$$d_{gow}(\mathbf{w}_i, \mathbf{w}_j) = \frac{\sum_{k=1}^{k=n} \frac{\left|z_i^{(k)} - z_j^{(k)}\right|}{\Delta z^{(k)}}}{r+q} + \frac{\sum_{k=1}^{k=r} S\left(x_i^{(k)}, x_j^{(k)}\right)}{r+n}, \tag{9.20}$$

where $\Delta z^{(k)}$ is the range of the continuous variables in the kth-dimension and S is a score function defined such that:

$$S\left(x_i^{(k)}, x_j^{(k)}\right) = \begin{cases} 0 \text{ if } x_i^{(k)} = x_j^{(k)} \\ 1 \text{ if } x_i^{(k)} \neq x_j^{(k)}. \end{cases} \tag{9.21}$$

From the re-definition of the distance in the joint continuous and discrete space, the *p-exponential* covariance function can be employed to create a mixed variable kernel for GP:

$$k(\mathbf{w}_i, \mathbf{w}_j) = \sigma^2 \exp\left[-\sum_{k=1}^{k=n} \theta_k \left(\frac{\left|z_i^{(k)} - z_j^{(k)}\right|}{\Delta z^{(k)}} \right)^{p_k} - \sum_{k=1}^{k=r} \theta_{k+n} \left(\frac{S\left(x_i^{(k)}, x_j^{(k)}\right)}{r+q} \right)^{p_{k+n}} \right]. \tag{9.22}$$

Furthermore, without loss of generality as only two values: 0 or 1 can be taken by $S(x_i^{(k)}, x_j^{(k)})$ (see Equation (9.21)), in practice, p_{k+n} is fixed to 1. This is equivalent to the definition of a mixed variable kernel as:

$$k(\mathbf{w}_i, \mathbf{w}_j) = k_c(\mathbf{z}_i, \mathbf{z}_j) * \prod_{s=1}^{r} [\mathbf{T}_s]_{x_i^{(s)}, x_j^{(s)}}, \tag{9.23}$$

where each matrix $\mathbf{T}_s$ is a CS covariance matrix defined as shown in Equation (9.19) with a c_s/v_s ratio equal to:

$$c_s/v_s = \exp\left[-\theta_s \left(\frac{S\left(x_i^{(s)}, x_j^{(s)}\right)}{r+n}\right)^{p_s}\right]. \tag{9.24}$$

The GP based on CS described above is characterized by $2(n+r)$ hyperparameters and scales better with the discrete dimension of the problem when compared to the previously described kernel parameterizations. Furthermore, the adaptation of a standard continuous Gaussian process into a CS-GP is relatively simple. However, because each discrete variable is only characterized by two hyperparameters θ and p, the surrogate model may present poor modeling performance in the presence of discrete variables with large numbers of discrete levels. For the same reason, the simultaneous presence of correlation and anti-correlation trends described by the same discrete variable might be more difficult to model when compared to the parameterizations described in the previous paragraphs (Pelamatti et al. 2019).

9.3.3 *Efficient Global Optimization with Mixed Discrete and Continuous Variables*

Standard EGO may be extended using adapted mixed continuous and discrete variable GP. In Pelamatti et al. (2018) a mixed variable adaptation of the Infill Criterion is proposed, defined as the product between the Expected Improvement (EI) and the Probability of Feasibility (P_f). The mixed variable GP kernel is defined in such a way that the resulting covariance matrix can be used to characterize a Gaussian distribution. By extension, the derivations of the EI and the P_f expressions remain valid for the mixed variable search space. EI defines the expected value of the predicted improvement with respect to the data set:

$$\mathbb{E}[I(\mathbf{w}^*)] = \mathbb{E}\left[\max\left(y_{\min} - Y(\mathbf{w}^*), 0\right)\right] \tag{9.25}$$

$$= (y_{\min} - \hat{y}(\mathbf{w}^*))\Phi\left(\frac{y_{\min} - \hat{y}(\mathbf{w}^*)}{\hat{s}(\mathbf{w}^*)}\right) + \hat{s}(\mathbf{w}^*)\phi\left(\frac{y_{\min} - \hat{y}(\mathbf{w}^*)}{\hat{s}(\mathbf{w}^*)}\right) \tag{9.26}$$

with $y_{\min}$ the current minimum value within the data set, while $\Phi(\cdot)$ and $\phi(\cdot)$ are the standard distribution and normal density functions, respectively. P_f defines the probability that all the constraints the problem is subject to are satisfied at the unmapped location $\mathbf{w}^*$ of the search space. Given a constraint function $g_i(\cdot)$, the

probability for it to be satisfied at $\mathbf{w}^*$ can be estimated with:

$$\mathbb{P}(g_i(\mathbf{w}^*) \leq 0) = \Phi\left(\frac{0 - \hat{g}_i(\mathbf{w}^*)}{\hat{s}_{g_i}(\mathbf{w}^*)}\right), \tag{9.27}$$

where $\hat{s}_{g_i}$ refers to the estimated error in the prediction $\hat{g}_i$ of the constraint function. By extension, the P_f for problems subject to m constraints can be computed as:

$$P_f(\mathbf{w}^*) = \prod_{i=1}^{m} \mathbb{P}\left(g_i(\mathbf{w}^*) \leq 0\right). \tag{9.28}$$

The constrained optimization infill criterion IC is given by

$$IC(\mathbf{w}^*) = \mathbb{E}[I(\mathbf{w}^*)]P_f(\mathbf{w}^*). \tag{9.29}$$

The data sample to be added to the GP training data set is obtained by evaluating the value of the exact objective and constraint functions for the value of $\mathbf{w}^*$ that maximizes the IC:

$$\mathbf{w}^* = \text{argmax}(IC(\mathbf{w})). \tag{9.30}$$

An auxiliary optimization problem solving is therefore necessary. As the computation time required to evaluate the IC is negligible with respect to the exact objective and constraint functions, classical optimization algorithms can be employed. Once the value of $\mathbf{w}^*$ that maximizes IC has been determined, the exact objective and constraint functions of the optimization problem are computed at said location and the obtained data sample is added to the GP data set. Subsequently, the surrogate models (of the objective and constraints) are trained anew to account for the additional information provided by the added data sample. This process is repeated until a user-defined stopping criterion is reached.

In mixed continuous and discrete problems, the auxiliary optimization used to maximize IC can either be carried out separately in each category of the problem by subsequently choosing the category yielding the largest value or it can be directly performed in the mixed continuous/discrete search space. In Pelamatti et al. (2018), the IC is defined in a mixed variable search space (the same as the optimization problem) and consequently most of the commonly used algorithms, such as gradient-based ones, may not be used due to the presence of multiple local minima, therefore, an evolutionary algorithm is used. IC is optimized using on a mixed continuous/discrete Genetic Algorithm (GA) similar to the one presented by Stelmack et al. (1998) and coded with the help of the python based toolbox DEAP (Fortin et al. 2012). The GA optimization routine is terminated once the maximum number of generations has been reached, or alternatively after the optimum value of the objective function has not improved over a predefined number of generations.

9.3.4 Application to Toy Case

In order to illustrate the mixed continuous/discrete variable EGO algorithm with the different kernel parameterizations, the following toy case problem is considered:

$$\begin{aligned} \min \quad & f(\mathbf{z}, \mathbf{x}) && (9.31)\\ \text{w.r.t.} \quad & \mathbf{z}, \mathbf{x} \\ \text{s.t.} \quad & g(\mathbf{z}, \mathbf{x}) \leq 0 && (9.32)\\ & \mathbf{z} \in [-5, 5]^2, \mathbf{x} \in \{0, 1\}^2, \end{aligned}$$

where $\mathbf{z} = \left[z^{(1)}, z^{(2)}\right]$, $\mathbf{x} = \left[x^{(1)}, x^{(2)}\right]$ and:

$$f(\mathbf{z}, \mathbf{x}) = \begin{cases} 200 & +{z^{(1)}}^4 - 16 * {z^{(1)}}^2 + 5 * z^{(1)} + {z^{(2)}}^4 \\ & -16 * {z^{(2)}}^2 + 5 * z^{(2)} \quad \text{if } x^{(1)} = 0 \text{ and } x^{(2)} = 0 \\ {z^{(1)}}^4 & +12 * {z^{(1)}}^2 + 8 * z^{(1)} + {z^{(2)}}^4 \\ & -16 * {z^{(2)}}^2 + 5 * z^{(2)} \quad \text{if } x^{(1)} = 0 \text{ and } x^{(2)} = 1 \\ {z^{(1)}}^4 & +16 * {z^{(1)}}^2 - 2 * z^{(1)} + {z^{(2)}}^4 \\ & +12 * {z^{(2)}}^2 - 2 * z^{(2)} \quad \text{if } x^{(1)} = 1 \text{ and } x^{(2)} = 0 \\ {z^{(1)}}^4 & +3 * {z^{(1)}}^3 + 5 * z^{(1)} + {z^{(2)}}^4 \\ & +2 * {z^{(2)}}^3 + 8 * z^{(1)} \quad \text{if } x^{(1)} = 1 \text{ and } x^{(2)} = 1 \end{cases} \tag{9.33}$$

$$g(\mathbf{z}, \mathbf{x}) = \begin{cases} -(z^{(1)} - 2)^2 + (z^{(2)} - 1)^2 + 2 & \text{if } x^{(1)} = 0 \text{ and } x^{(1)} = 0 \\ -(z^{(1)} - 1.5)^2 + (z^{(2)} - 0.7)^2 + 4 & \text{if } x^{(1)} = 0 \text{ and } x^{(2)} = 1 \\ -(z^{(1)} - 2.5)^2 + (z^{(2)} + 2)^2 + 3 & \text{if } x^{(1)} = 1 \text{ and } x^{(2)} = 0 \\ -(z^{(1)} + 1)^2 + (z^{(2)} - 1.5)^2 + 3 & \text{if } x^{(1)} = 1 \text{ and } x^{(2)} = 1. \end{cases} \tag{9.34}$$

The objective function is derived from the Styblinski-Tank function (Silagadze 2007) and adapted for different categories. The four objective functions (depending on the category defined by the discrete variable vector $\mathbf{x}$) and the associated constraints are presented in Figure 9.14. Depending on the value of the discrete variable vector $\mathbf{x}$, the objective function, and the constraints are quite different but also present some similarities that could be exploited by the mixed variable adaptation of EGO. Among the four categories, the optimum is within the $\mathbf{x} = [0, 1]$ category.

In order to evaluate the efficiency of the mixed-GP approaches, a reference method named category-wise (CW) is used consisting in building a separate GP (for the objective and the constraint functions) in each category of the problem (here height GPs) and no information are exchanged. Using these GPs, EGO is run and the most promising point (based on an optimization of EI in each category and comparing the results for each category) is added in the corresponding category and the GPs are trained anew (for that particular category). For the SMBDO

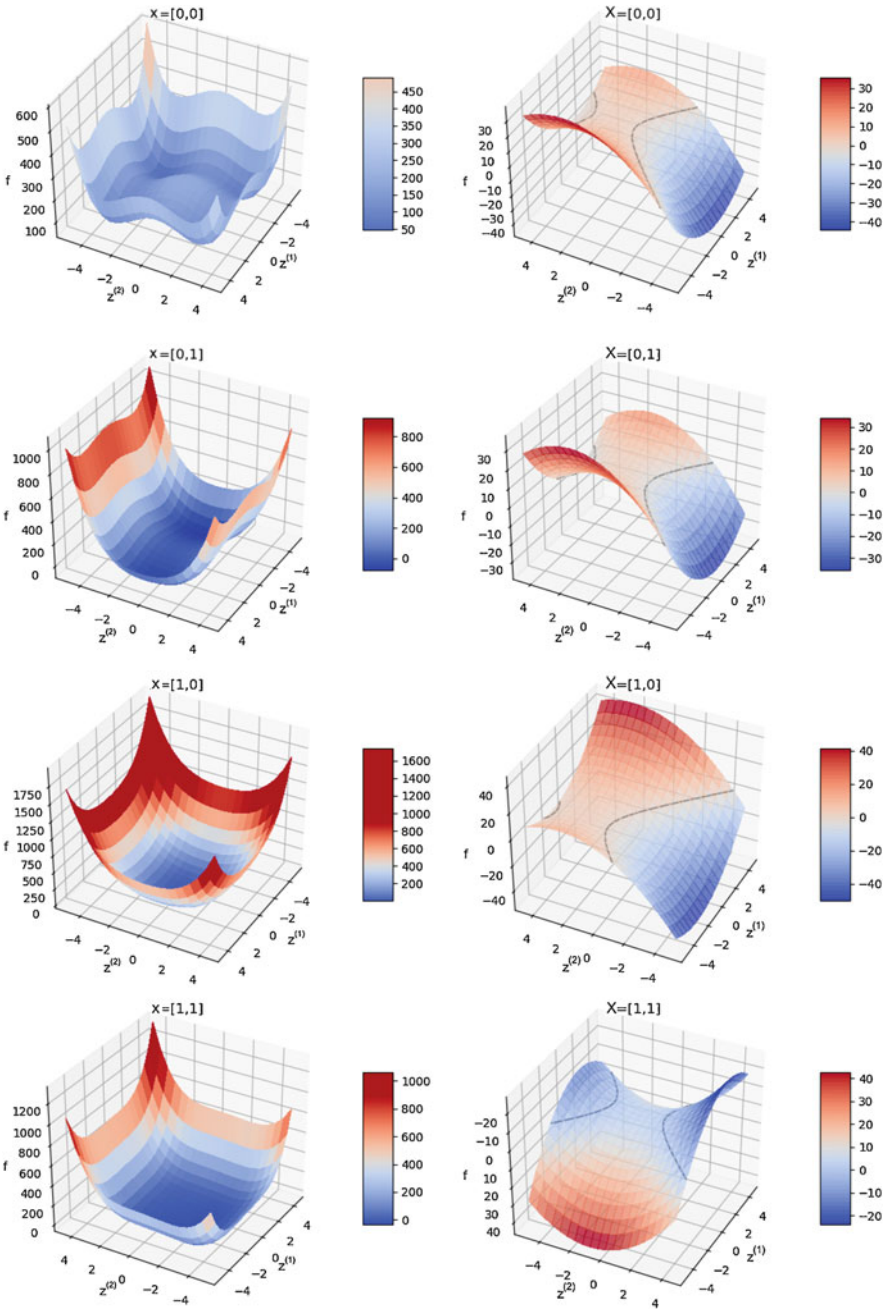


Fig. 9.14 Objective (left) and constraint (right) functions for the different value of the discrete variable vector **x**-toy case problem

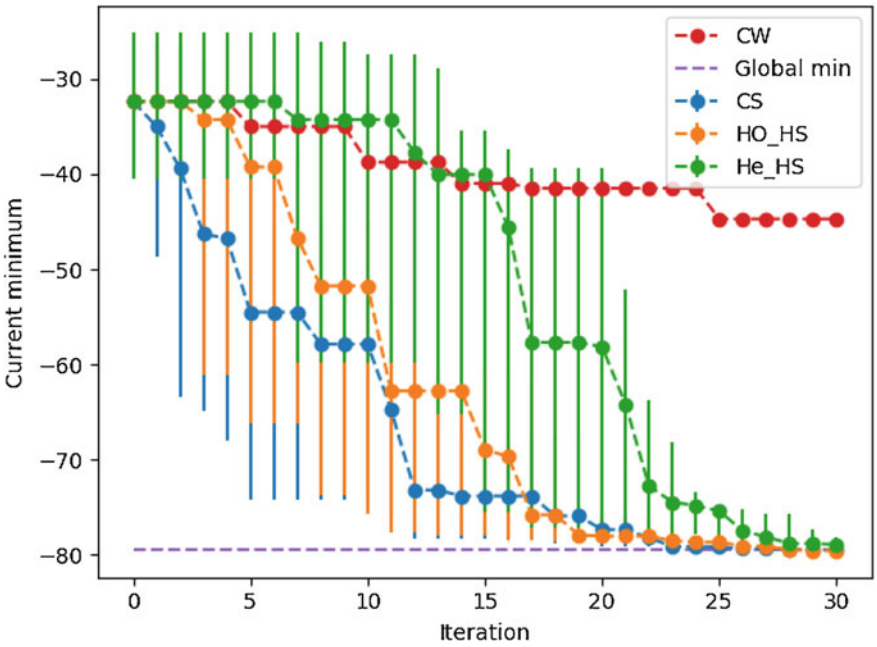


Fig. 9.15 Convergence of mixed EGO and reference approaches over 10 repetitions. The used optimization techniques rely on heteroscedastic dimension-wise decomposition (He_HS), homoscedastic dimension-wise decomposition (Ho_HS), compound symmetry decomposition (CS), category-wise separate surrogate modeling (CW). The median value and the first and last quartile are represented on the convergence graph

techniques, an initial training data set of 24 samples is used (i.e., six samples for each independent EGO in the category-wise (CW) case) and subsequently 30 additional data points are infilled during the optimization process. The results obtained for this test case over 10 repetitions are presented in Figures 9.15 and 9.16.

The mixed variables adaptations of EGO provide better convergence results and robustness to initial DoE than the standard category-wise EGO. Figure 9.15 presents the convergence curves (with the median value and the first and last quartile) for the EGO-based techniques and the three mixed variable EGO adaptations outperform the category-wise EGO which would require more iterations to converge. The robustness to the initial DoE is illustrated in Figure 9.16 where the obtained minimum for the ten repetitions are represented and the three mixed variable EGO adaptations are clearly more robust than the category-wise EGO.

Due to the simplicity of this analytical test case, no noticeable difference in the performance between the proposed mixed variable EGO adaptations can be noticed. For more complex test cases, see (Pelamatti et al. 2018). The data samples infilled by the presented methods during one of the repetitions are presented in Figure 9.17. It can be seen that an infill sample has been added in the category $\mathbf{x} = [0, 1]$ for $\mathbf{z} = [1.8, -1.2]$ close to the constraint and the global minimum.

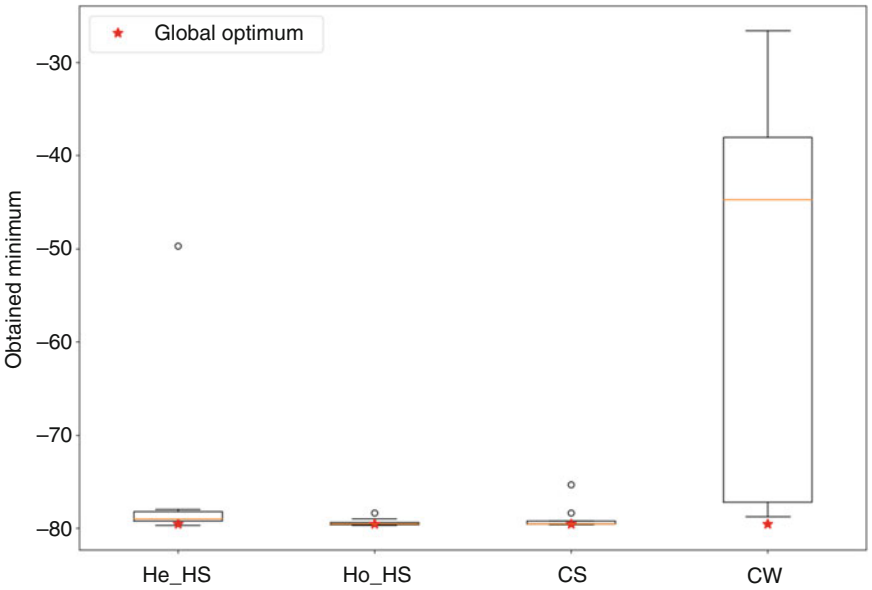


Fig. 9.16 Optimization results obtained over 10 repetitions. From left to right, the used optimization techniques rely on heteroscedastic dimension-wise decomposition (He_HS), homoscedastic dimension-wise decomposition (Ho_HS), compound symmetry decomposition (CS), category-wise separate surrogate modeling (CW)

This toy case problem involves heteroscedastic properties. Theoretically, the He_HS discrete kernel parameterization is supposed to be the most appropriate choice, as it provides a more accurate modeling of heteroscedastic functions. This aspect is discussed by Roustant et al. (2018), where several relevant examples are provided. However, the variance of a given category depends on the trend over the entirety of the search space, while for the SMBDO techniques, the relevant metric would be the variance of the problem functions in the areas of interest of the search space, as that is where the modeling accuracy is most important in order to determine the most suitable location in which to infill data samples. Indeed, in SMBDO, the purpose is not to accurately model the objective and constraint functions over the entire search space, but to identify the problem optimum while managing surrogate model uncertainty. In general, unless problem specific knowledge is provided, there is usually no way of knowing the optimum neighborhood variance a priori. Moreover, as previously explained, the He_HS relies on a larger number of hyperparameters, the optimal values of which are difficult to determine when limited amounts of data are provided. For these two reasons, the a priori knowledge of the heteroscedasticity or homoscedasticity of a given optimization problem is not sufficient in order to determine the most suitable discrete kernel parameterizations for the SMBDO algorithm.

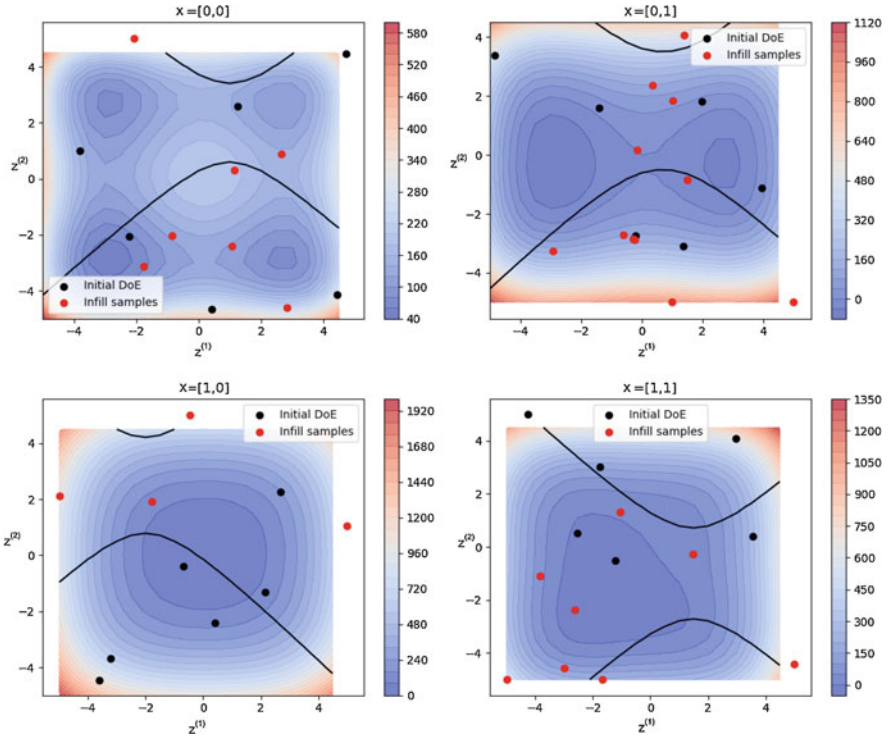


Fig. 9.17 Contour plot of the objective function and infill samples with CS parameterization for one of the 10 repetitions. The contour plot of $g(\mathbf{z}, \mathbf{x}) = 0$ is represented with the black lines (feasible regions are outside)

This toy case illustrates the interest of using mixed variable EGO adaptations compared to classical optimization algorithms to converge efficiently to the global minimum with respect to the continuous and discrete design variables with a limited number of function evaluations.

9.4 Summary

This chapter presents topics related to MDO in order to solve complex aerospace vehicle design problem. The two discussed research fields are multi-objective and the mixed continuous/discrete optimization. In order to tackle these challenges, recent works focused on the use of Gaussian processes (Emmerich and Klinkenberg 2008; Pelamatti et al. 2018; Brevault et al. 2019; Hebbal et al. 2019) and have been introduced in this chapter. Each of the presented approach have been illustrated on a toy case problem highlighting the interest of suited adaptation of GP for such complex problems. GP have interesting properties in terms of prediction and

estimation of the associated prediction errors enabling efficient adaptive strategies for multi-objective or mixed continuous/discrete variable problems with a limited number of computationally intensive simulations. Some of the presented techniques are still at the beginning of their development (for instance the Deep Gaussian processes for optimization) and further works are necessary to become a useful tool for MDO and MDO under uncertainty.

Acknowledgements The work on EGO and DGP was co-funded by ONERA-The French Aerospace Lab and Université de Lille, in the context of a joint PhD thesis. In addition, experiments presented in this chapter were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER, and several Universities as well as other organizations. The work on EGO for mixed variable problem was co-funded by ONERA—The French Aerospace Lab and CNES.

References

Agresti, A. (1996). *An introduction to categorical data analysis*. Wiley.

Aliakbargolkar, A., Crawley, E. F., Wicht, A. C., Battat, J. A., and Calandrelli, E. D. (2013). Systems architecting methodology for space transportation infrastructure. *Journal of Spacecraft and Rockets*, 50(3):579–590.

Arias-Montano, A., Coello, C. A. C., and Mezura-Montes, E. (2012). Multiobjective evolutionary algorithms in aeronautical and aerospace engineering. *IEEE Transactions on Evolutionary Computation*, 16(5):662–694.

Armstrong, M., De Tenorio, C., Mavris, D., and Garcia, E. (2008). Function based architecture design space definition and exploration. In *26th International Congress of the Aeronautical Sciences, Anchorage, AK, USA*.

Audet, C., Booker, A. J., Dennis Jr, J. E., Frank, P. D., and Moore, D. W. (2000). A surrogate-model-based method for constrained optimization. In *8th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA, USA*.

Audet, C. and Dennis Jr, J. E. (2006). Mesh adaptive direct search algorithms for constrained optimization. *SIAM Journal on optimization*, 17(1):188–217.

Bader, J. and Zitzler, E. (2011). Hype: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary computation*, 19(1):45–76.

Bajer, L. and Holena, M. (2013). Surrogate model for mixed-variables evolutionary optimization based on GLM and RBF networks. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 7741 LNCS, pages 481–490. Springer, Berlin, Heidelberg.

Barjhoux, P.-J., Diouane, Y., Grihon, S., Bettebghor, D., and Morlier, J. (2018). A bilevel methodology for solving a structural optimization problem with both continuous and categorical variables. In *2018 Multidisciplinary Analysis and Optimization Conference, Atlanta, GA, USA*.

Beauthier, C., Mahajan, A., Sainvitu, C., Hendrick, P., Sharifzadeh, S., and Verstraete, D. (2014). Hypersonic cryogenic tank design using mixed-variable surrogate-based optimization. In *4th International Conference on Engineering Optimization, Lisbon, Portugal*.

Beume, N., Naujoks, B., and Emmerich, M. (2007). SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669.

Brevault, L., Balesdent, M., Hebbal, A., and Patureau De Mirand, A. (2019). Surrogate model-based multi-objective MDO approach for partially Reusable Launch Vehicle design. In *AIAA Scitech Forum 2019, San Diego, CA, USA*.

Bui, T., Hernández-Lobato, D., Hernandez-Lobato, J., Li, Y., and Turner, R. (2016). Deep Gaussian Processes for regression using approximate expectation propagation. In *International Conference on Machine Learning, New-York, NY, USA*.

Castellini, F. and Lavagna, M. (2012). Comparative analysis of global techniques for performance and design optimization of launchers. *Journal of Spacecraft and Rockets*, 49(2):274–285.

Chepko, A., De Weck, O., Linne, D., Santiago-Maldonado, E., and Crossley, W. (2008). Architecture modeling of in-situ oxygen production and its impacts on lunar campaigns. In *AIAA SPACE 2008 conference & exposition, San Diego, CA, USA*.

Chepko, A. (2009). Technology selection and architecture optimization of in-situ resource utilization systems. Master's thesis, Massachusetts Institute of Technology.

Coello Coello, C. and Lechuga, M. (2002). MOPSO: a proposal for multiple objective particle swarm optimization. In *2002 IEEE Congress on Evolutionary Computation (CEC), Honolulu, HI, USA*.

Dai, Z., Damianou, A., González, J., and Lawrence, N. (2015). Variational auto-encoded deep Gaussian processes. *arXiv preprint arXiv:1511.06455*.

Damianou, A. and Lawrence, N. (2013). Deep Gaussian Processes. In *16th International Conference on Artificial Intelligence and Statistics, Scottsdale, AZ, USA*.

Davis, C. (1962). The norm of the Schur product operation. *Numerische Mathematik*, 4(1):343–344.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*, volume 16. John Wiley & Sons.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *International Conference on Parallel Problem Solving From Nature, Paris, France*.

Deb, K., Pratap, A., and Meyarivan, T. (2001). Constrained test problems for multi-objective evolutionary optimization. In *International conference on evolutionary multi-criterion optimization, Zurich, Switzerland*.

Donahue, B. (2001). Architecture selection-the key decision for human mars mission planning. In *37th Joint Propulsion Conference and Exhibit, Sault Lake City, UT, USA*.

Emmerich, M. and Klinkenberg, J.-W. (2008). The computation of the expected improvement in dominated hypervolume of Pareto front approximations. *Technical report, Leiden University*, 34.

Emmerich, M. T., Giannakoglou, K. C., and Naujoks, B. (2006). Single-and multiobjective evolutionary optimization assisted by Gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation*, 10(4):421–439.

Fazeley, H., Taei, H., Naseh, H., and Mirshams, M. (2016). A multi-objective, multidisciplinary design optimization methodology for the conceptual design of a spacecraft bi-propellant propulsion system. *Structural and Multidisciplinary Optimization*, 53(1):145–160.

Ferreira, C. (2001). Gene expression programming: A new adaptive algorithm for solving problems. *Complex System*, 13(2):87–129.

Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A., Parizeau, M., and Gagné, C. (2012). {DEAP}: Evolutionary Algorithms Made Easy. *Journal of Machine Learning Research*, 13:2171–2175.

Frank, C., Marlier, R., Pinon-Fischer, O. J., and Mavris, D. N. (2016). An evolutionary multi-architecture multi-objective optimization algorithm for design space exploration. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, San Diego, CA, USA*.

Frank, C. P. (2016). *A design space exploration methodology to support decisions under evolving uncertainty in requirements and its application to advanced vehicles*. PhD thesis, Georgia Institute of Technology.

Fujikawa, T., Tsuchiya, T., and Tomioka, S. (2013). Multi-Objective, High-Accuracy Trajectory Optimization Using SQP and Sampling on the Manifold. In *2013 Asia-Pacific International Symposium on Aerospace Technology, Takamatsu, Japan*.

Fujikawa, T., Tsuchiya, T., and Tomioka, S. (2015). Multi-objective, multidisciplinary design optimization of TSTO space planes with RBCC engines. In *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Kissimmee, FL, USA*.

Gower, J. C. (1971). A General Coefficient of Similarity and Some of Its Properties. *Biometrics*, 27(4):857.

Gunawan, S., Azarm, S., Wu, J., and Boyars, A. (2003). Quality-assisted multi-objective multidisciplinary genetic algorithms. *AIAA Journal*, 41(9):1752–1762.

Haas, T. C. (1990). Kriging and automated variogram modeling within a moving window. *Atmospheric Environment. Part A. General Topics*, 24(7):1759–1769.

Haftka, R. T. and Watson, L. T. (2006). Decomposition theory for multidisciplinary design optimization problems with mixed integer quasiseparable subsystems. *Optimization and Engineering*, 7(2):135–149.

Halstrup, M. (2016). *Black-box optimization of mixed discrete-continuous optimization problems*. PhD thesis, TU Dortmund.

Hebbal, A., Brevault, L., Balesdent, M., Taibi, E.-G., and Melab, N. (2018). Efficient Global Optimization using Deep Gaussian Processes. In *2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, Brazil*.

Hebbal, A., Brevault, L., Balesdent, M., Talbi, E.-G., and Melab, N. (2019). Multi-objective optimization using Deep Gaussian Processes: Application to aerospace vehicle design. In *AIAA Scitech Forum 2019, San Diego, CA, USA*.

Higdon, D., Swall, J., and Kern, J. (1999). Non-stationary spatial modeling. *Bayesian statistics*, 6(1):761–768.

Holmström, K., Quttineh, N. H., and Edvall, M. M. (2008). An adaptive radial basis algorithm (ARBF) for expensive black-box mixed-integer constrained global optimization. *Optimization and Engineering*, 9:311–339.

Huang, C.-H. (2003). *Development of multi-objective concurrent subspace optimization and visualization methods for multidisciplinary design*. PhD thesis, State University of New York at Buffalo.

Huang, C.-H., Galuski, J., and Bloebaum, C. L. (2007). Multi-objective pareto concurrent subspace optimization for multidisciplinary design. *AIAA Journal*, 45(8):1894–1906.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492.

Kang, N., Kokkolaras, M., and Papalambros, P. Y. (2014). Solving multiobjective optimization problems using quasi-separable MDO formulations and analytical target cascading. *Structural and Multidisciplinary Optimization*, 50(5):849–859.

Knowles, J. (2006). ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66.

Knowles, J. D. and Corne, D. W. (2000). Approximating the nondominated front using the pareto archived evolution strategy. *Evolutionary computation*, 8(2):149–172.

Kosugi, Y., Oyama, A., Fujii, K., and Kanazaki, M. (2011). Multidisciplinary and multi-objective design exploration methodology for conceptual design of a hybrid rocket. In *Infotech@ Aerospace 2011, St Louis, MO, USA*.

Kothari, A. and Webber, D. (2010). Potential demand for orbital space tourism opportunities made available via reusable rocket and hypersonic architectures. In *AIAA SPACE 2010 Conference & Exposition, Anaheim, CA, USA*.

Kurapati, A. and Azarm, S. (2000). Immune network simulation with multiobjective genetic algorithms for multidisciplinary design optimization. *Engineering Optimization*, 33(2):245–260.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436.

Mavris, D. N. and Kirby, M. R. (1999). Technology identification, evaluation, and selection for commercial transport aircraft. In *58th Annual Conference Of Society of Allied Weight Engineers, San Jose, CA, USA*.

McAllister, C. D., Simpson, T. W., Hacker, K., Lewis, K., and Messac, A. (2005). Integrating linear physical programming within collaborative optimization for multiobjective multidisciplinary design optimization. *Structural and Multidisciplinary Optimization*, 29(3):178–189.

Müller, J., Shoemaker, C. A., and Piché, R. (2013). SO-MI: A surrogate model algorithm for computationally expensive nonlinear mixed-integer black-box global optimization problems. *Computers and Operations Research*, 40(5):1383–1400.

Nebro, A. J., Durillo, J. J., Garcia-Nieto, J., Coello, C. C., Luna, F., and Alba, E. (2009). SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In *IEEE Symposium on Computational intelligence in multi-criteria decision-making, Nashville, TN, USA*.

Paciorek, C. J. and Schervish, M. J. (2006). Spatial modelling using a new class of nonstationary covariance functions. *Environmetrics*, 17(5):483–506.

Parashar, S. and Bloebaum, C. (2006). Multi-objective genetic algorithm concurrent subspace optimization (MOGACSSO) for multidisciplinary design. In *47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Newport, RI, USA*.

Pelamatti, J., Brevault, L., Balesdent, M., Talbi, E.-G., and Guerin, Y. (2019). Overview and comparison of Gaussian process-based surrogate models for mixed continuous and discrete variables, application on aerospace design problems. *High-performance simulation based optimization, Springer series on Computational Intelligence, Springer*.

Pelamatti, J., Brevault, L., Balesdent, M., Talbi, E.-G., and Guerrin, Y. (2018). Efficient global optimization of constrained mixed variable problems. *Journal of Global Optimization*, 73(3):583–613.

Pinheiro, J. and Bates, D. (2009). Mixed-Effects Models in S and S-PLUS. *Statistics and Computing, Springer-Verlag*.

Pinheiro, J. and Bates, D. M. (1996). Unconstrained parametrizations for variance-covariance matrices. *Statistics and Computing*, 6(3):289–296.

Prasadh, N., Moss, R., Collett, K., Nelessen, A., Edwards, S., and Mavris, D. (2014). A systematic method for SME-driven space system architecture down-selection. In *AIAA SPACE 2014 conference and exposition, San Diego, CA, USA*.

Qian, P. Z. G., Wu, H., and Wu, C. F. J. (2008). Gaussian Process Models for Computer Experiments With Qualitative and Quantitative Factors. *Technometrics*, 50(3):383–396.

Qin, A. K., Huang, V. L., and Suganthan, P. N. (2009). Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE transactions on Evolutionary Computation*, 13(2):398–417.

Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Kevin Tucker, P. (2005). Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41(1):1–28.

Rashid, K., Ambani, S., and Cetinkaya, E. (2008). An adaptive multiquadric radial basis function method for expensive black-box mixed-integer nonlinear constrained optimization. *Engineering Optimization*, 45(2):185–206.

Rasmussen, C. E. and Ghahramani, Z. (2002). Infinite mixtures of Gaussian process experts. In *Advances in neural information processing systems*, pages 881–888.

Rebonato, R. and Jaeckel, P. (2011). The Most General Methodology to Create a Valid Correlation Matrix for Risk Management and Option Pricing Purposes. *SSRN Electronic Journal*.

Regis, R. G. (2014). Evolutionary Programming for High-Dimensional Constrained Expensive Black-Box Optimization Using Radial Basis Functions. *IEEE Transactions on Evolutionary Computation*, 18(3):326–347.

Roustant, O., Padonou, E., Deville, Y., Clément, A., Perrin, G., Giorla, J., and Wynn, H. (2018). Group kernels for Gaussian process metamodels with categorical inputs. *arXiv:1802.02368*.

Roy, S., Moore, K., Hwang, J. T., Gray, J. S., Crossley, W. A., and Martins, J. (2017). A mixed integer efficient global optimization algorithm for the simultaneous aircraft allocation-mission-design problem. In *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Grapevine TX, USA*.

Salimbeni, H. and Deisenroth, M. (2017). Doubly stochastic variational inference for deep Gaussian processes. In *31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA*.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. Springer New York.

Schonlau, M., Welch, W. J., and Jones, D. R. (1998). Global versus local search in constrained optimization of computer models. *Lecture Notes-Monograph Series*, pages 11–25.

Shawe-Taylor, J. and Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. Cambridge University Press.

Silagadze, Z. (2007). Finding two-dimensional peaks. *Physics of Particles and Nuclei Letters*, 4(1):73–80.

Soban, D. S. and Mavris, D. N. (2013). Assessing the impact of technology on aircraft systems using technology impact forecasting. *Journal of Aircraft*, 50(5):1380–1393.

Stelmack, M., Nakashima, N., and Batill, S. (1998). Genetic algorithms for mixed discrete/continuous optimization in multidisciplinary design. In *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, USA*.

Svenson, J. D. and Santner, T. J. (2010). Multiobjective optimization of expensive black-box functions via expected maximin improvement. *The Ohio State University, Columbus, Ohio*, 32.

Swiler, L. P., Hough, P. D., Qian, P., Xu, X., Storlie, C., and Lee, H. (2014). *Surrogate Models for Mixed Discrete-Continuous Variables*. Springer International Publishing, Cham.

Tappeta, R. and Renaud, J. (1997). Multiobjective collaborative optimization. *Journal of Mechanical Design*, 119(3):403–411.

Titsias, M. (2009). Variational learning of inducing variables in sparse Gaussian processes. In *12th International Conference on Artificial Intelligence and Statistics, Clearwater Beach, FL, USA*.

Titsias, M. and Lawrence, N. D. (2010). Bayesian Gaussian process latent variable model. In *13th International Conference on Artificial Intelligence and Statistics, Sardinia, Italy*.

Villeneuve, F. and Mavris, D. (2005). A new method of architecture selection for launch vehicles. In *AIAA/CIRA 13th International Space Planes and Hypersonics Systems and Technologies Conference, Capua, Italy*.

Wagner, T., Emmerich, M., Deutz, A., and Ponweiser, W. (2010). On expected-improvement criteria for model-based multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature, Krakow, Poland*.

Walton, M. A. and Hastings, D. (2004). Applications of uncertainty analysis applied to architecture selection of satellite systems. *Journal of Spacecraft and Rockets*, 41(1):75–84.

Xiao, M., Shao, X., Gao, L., and Luo, Z. (2015). A new methodology for multi-objective multidisciplinary design optimization problems based on game theory. *Expert Systems with Applications*, 42(3):1602–1612.

Xiong, Y., Chen, W., Apley, D., and Ding, X. (2007). A non-stationary covariance-based kriging method for metamodelling in engineering design. *International Journal for Numerical Methods in Engineering*, 71(6):733–756.

Zhang, K.-S., Han, Z.-H., Li, W.-J., and Song, W.-P. (2008). Bilevel adaptive weighted sum method for multidisciplinary multi-objective optimization. *AIAA Journal*, 46(10):2611–2622.

Zhang, Q., Liu, W., Tsang, E., and Virginas, B. (2010). Expensive multiobjective optimization by MOEA/d with Gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14(3):456–474.

Zhang, Y. and Notz, W. I. (2015). Computer experiments with qualitative and quantitative variables: A review and reexamination. *Quality Engineering*, 27(1):2–13.

Zhou, Q., Qian, P. Z. G., and Zhou, S. (2011). A Simple Approach to Emulation for Computer Models With Qualitative and Quantitative Factors. *Technometrics*, 53(3):266–273.