Distrib Parallel Databases (2007) 21:167–192
DOI 10.1007/s10619-007-7009-9

Optimizing thermal design of data center cabinets with a new multi-objective genetic algorithm

G. Li · M. Li · S. Azarm · J. Rambo · Y. Joshi

Published online: 13 March 2007

Abstract There is an ever increasing need to use optimization methods for thermal design of data centers and the hardware populating them. Airflow simulations of cabinets and data centers are computationally intensive and this problem is exacerbated when the simulation model is integrated with a design optimization method. Generally speaking, thermal design of data center hardware can be posed as a constrained multi-objective optimization problem. A popular approach for solving this kind of problem is to use Multi-Objective Genetic Algorithms (MOGAs). However, the large number of simulation evaluations needed for MOGAs has been preventing their applications to realistic engineering design problems. In this paper, details of a substantially more efficient MOGA are formulated and demonstrated through a thermal analysis simulation model of a data center cabinet. First, a reduced-order model of the cabinet problem is constructed using the Proper Orthogonal Decomposition (POD). The POD model

Recommended by: Monem Beitelmal

G. Li · M. Li · S. Azarm
Department of Mechanical Engineering, University of Maryland, College Park, MD 20742

G. Li
e-mail: ligenzi@umd.edu

M. Li
e-mail: mli6@umd.edu

S. Azarm
e-mail: azarm@umd.edu

J. Rambo
Shell Global Solutions, Houston, TX 77082
e-mail: jeff.rambo@shell.com

Y. Joshi (✉)
G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332
e-mail: yogendra.joshi@me.gatech.edu

is then used to form the objective and constraint functions of an optimization model. Next, this optimization model is integrated with the new MOGA. The new MOGA uses a "kriging" guided operation in addition to conventional genetic algorithm operations to search the design space for global optimal design solutions. This approach for optimal design is essential to handle complex multi-objective situations, where the optimal solutions may be non-obvious from simple analyses or intuition. It is shown that in optimizing the data center cabinet problem, the new MOGA outperforms a conventional MOGA by estimating the Pareto front using 50% fewer simulation calls, which makes its use very promising for complex thermal design problems.

Keywords Data center · Thermal design · Multi-objective optimization · Genetic algorithm · Meta-modeling

1 Introduction

Energy consumption for large data center facilities is currently in the 10 MW–100 MW range [27, 28]. The computer room cooling equipment accounts for as much as 50% additional energy consumption [19]. As old computing equipment is upgraded every 2–3 years to bring in higher performance cabinets, which often dissipate higher power and have higher cooling requirements, the facility operators are increasingly facing the challenge of optimizing the use of a fixed amount of cooling resources. Currently, individual racks can dissipate as much as 30 kW of power, and projections are for further increases, as dense architectures such as blades become more common [4]. With continually rising energy costs, it is crucial to arrive at the best thermal design that results in the least expenditure of cooling resources as new equipment is brought in during such retrofits of data center facilities. An optimized layout of heat generating servers within new cabinets provides an opportunity for reduction in energy consumption for cooling. Moreover, such savings when accounted for the entire facility may also result in substantial savings in utility bills for the facility operator.

Generally speaking, there are two classes of design optimization methods that can be used for optimizing thermal design problems of data centers (see, e.g., [3, 7, 26]). These two classes are gradient- and non-gradient-based methods. Gradient-based methods (see, e.g., [6]) require derivative information for the optimization functions (i.e., objective and constraint functions) and usually have an implicit assumption that these functions are "smooth" and that design variables are continuous. In general, gradient-based methods can only guarantee a local optimum design solution unless the functions used in the problem have special properties such as linearity or convexity. The smoothness assumption can be relaxed in non-gradient-based techniques. One popular and general class of non-gradient-based techniques for design optimization is Genetic Algorithms (GAs) [13, 14]. GAs were developed by John Holland [14] and are based on evolutionary concepts. These algorithms can handle non-smooth functions, a situation common in many engineering design problems. GAs can also obtain a global optimum design solution. Moreover, they can be easily extended to handle multiple

design objectives, i.e., Multi-Objective Genetic Algorithms or MOGAs [8, 10, 18, 23]. However, the main shortcoming of MOGA-based optimization methods is that they require a large number of simulation calls. Unfortunately, the large number of simulation calls for MOGAs has prohibited applications of these algorithms to realistic engineering design problems, such as thermal design of data centers, whereby even a single simulation call can require a significant amount of computational effort.

A common strategy to reduce the computational effort of optimization methods, such as MOGAs, when integrated with expensive simulation models is to use meta-modeling (see, e.g., [5, 20, 35, 36, 39]). In particular, a number of techniques incorporating meta-modeling with genetic algorithm based methods have been reported in the literature (e.g., [16, 32]). Some of these methods use meta-modeling in the genetic algorithm's fitness estimation [16]. Others use meta-modeling to guide the search in the design space [32]. While the fitness estimation methods have been reported to reduce the computational cost significantly [16], these methods have the risk of generating false optimum solutions. On the other hand, the risk can be reduced by using meta-modeling for searching the design space, especially when the approximations are not sufficiently accurate (see, e.g., [1, 2, 31, 38]). Unfortunately, all of the aforementioned methods heavily depend on the accuracy of the meta-models over the entire design space or some specific neighborhood.

This paper presents a genetic algorithm-based multi-objective optimization framework and demonstrates applicability of this framework to thermal design of data center cabinet problems. In the framework, a reduced-order model of the cabinet problem is constructed using the Proper Orthogonal Decomposition (POD). The POD model is then used to form the objective and constraint functions of an optimization model. This optimization model is integrated with the new MOGA. The new MOGA falls under the methods that use meta-modeling to guide the search in the design space, i.e., a kriging guided reproduction approach, that has several advantages over the previous techniques as follows. First, the new MOGA has the flexibility of using the meta-modeling in the design space only where the meta-model produces reasonably accurate estimates of simulation responses. Second, the new MOGA uses an objective rather than subjective measure to determine whether the simulation response for a design alternative can be estimated by a meta-model. Finally, in contrast to many meta-model assisted optimization methods in the literature, the new MOGA incorporates a global interpolation meta-modeling technique based on a newly defined kriging guided reproduction approach that can significantly reduce the number of iterations needed for convergence of the optimization procedure.

The present paper is organized as follows. Definitions and terminology are collected in Section 2. The thermal design of a data center cabinet problem and the corresponding POD model are introduced in Section 3. In Section 4, brief reviews of genetic algorithms and kriging meta-modeling are outlined, and details of the new MOGA are provided. Results and discussions for the data center cabinet problem are given in Section 5. Key conclusions of the paper are provided in Section 6.

2 Definition and terminology

A multi-objective design optimization problem is formulated as follows:

$$\begin{array}{lll} \underset{x}{\text{minimize}} & f_i(\boldsymbol{x}) & i = 1, \ldots, M \\ \text{subject to :} & g_j(\boldsymbol{x}) \leq 0 & j = 1, \ldots, J \\ & h_k(\boldsymbol{x}) = 0 & k = 1, \ldots, K \end{array} \tag{1}$$

where f_i refers to the scalar design objective function, $\boldsymbol{x} = (x_1, \ldots, x_N)^T$ is the design variable vector, the scalar functions $g_1, \ldots, g_J$ are the inequality constraints, and $h_1, \ldots, h_K$ are the scalar equality constraints. It is assumed that there are trade-offs among at least two of the M objective functions. As such, the optimization problem in Eq. (1) has more than one optimum solution. Some terminologies that are used in the paper are defined below.

Feasible design point. A design point that does not violate any constraints (i.e., g_j and h_k in Eq. (1)).

Non-dominated and dominated design point and set. A design point is said to be non-dominated if no other design point in the current set of design points is better than that point with respect to all objectives [10]. The set of all non-dominated design points in the current set of design points forms a non-dominated design set. The remaining design points in the current set form a dominated set.

Pareto optimum design points (or Pareto frontier). Pareto optimum design points refer to a set of points (or Pareto frontier) that are the solutions to the problem in Eq. (1), and are non-dominated when the entire feasible design space is considered.

Kriging based non-dominated design points. These points are an estimate of Pareto points obtained based on kriging meta-models.

3 Thermal design of a data center cabinet

The optimization of a two-dimensional representation of an enclosed data center cabinet containing 10 individual servers is selected to illustrate the new MOGA method. Cold supply air at 15°C from the data center facility is delivered to the servers through a 0.39 m cutout in the bottom of the rack. Each server contains two 0.30 m tall by 0.50 m long blocks with uniform heat generation to mimic the power dissipation from actual chips. Since a two-dimensional geometry is considered, the chips are assumed to extend infinitely in the third direction (depth) and the specified power dissipation is per unit depth. A cubic pressure-velocity relationship is prescribed at each server exhaust to model an induced draft fan and a lumped pressure drop model is included at the inlet to server, producing a nominal server flow rate of 0.180 kg/s (310 CFM). Figure 1 describes the rack dimensions, based on commercially available units, and schematically depicts the dominant airflow paths.

The full numerical model neglects buoyancy effects [29] and solves the steady incompressible Reynolds-averaged Navier-Stokes (RANS) momentum equations using the standard 2 equation k-ε turbulence modelclosure with no body forces. The computations used second order upwinding and SIMPLEC pressure-velocity coupling with

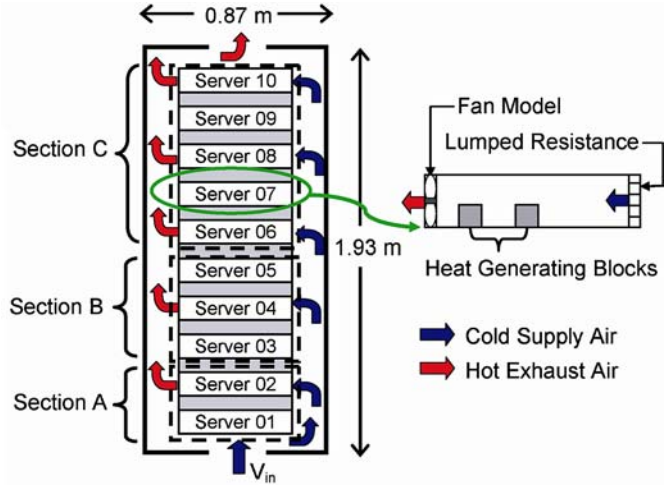


Fig. 1 Rack model geometry and dominant airflow paths

PRESTO pressure interpolation, see [12] for a description of the numerical methods employed. The final converged model contained 20,263 grid cells for a total of 121,578 degrees of freedom. Note that the sum of the server flow rates will generally be greater than the net rack flow rate. This flow rate imbalance requires a large degree of recirculation, or a server's hot exhaust being drawn through another server before exiting the rack, and a correspondingly complex flow field.

The thermal design problem associated with the rack model in Fig. 1 is to dissipate as much power as possible, while maintaining all chip temperatures below a prescribed maximum allowable value, taken to be 85°C. Increasing the rack inlet velocity causes the blowers in the data center air handing units to consume more power, but does not necessarily correlate with an overall reduction in chip temperatures. The partitioning the cabinet into sections A, B and C will be explained in subsequent sections, but note the complex geometry and resultant flow patterns within the rack cause the cooling supply air to bypass the bottom 2 servers as V_{in} increases. This is the physical mechanism that allows the power to the first 2 servers to be grouped as a single design variable. Practically, redistributing the power to the individual servers corresponds to changing the vertical arrangement of servers within the rack. The corresponding optimization problem is formulated in Section 5, and results are obtained and also discussed in that section.

3.1 Reduced-order modeling framework

A reduced-order model of the data center cabinet is constructed using the Proper Orthogonal Decomposition (POD). The POD model is used to form the objective and constraint functions of the optimization model. The POD empirically computes the basis functions forming the optimal linear subspace given an ensemble of system observations. The empirical basis functions or 'POD modes' are used to compute new

approximate solutions using the expansion theorem:

$$\boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{u}_0(\boldsymbol{x}, t) + \sum_{i=1}^{m} a_i(t)\boldsymbol{\varphi}_i(\boldsymbol{x}) \tag{2}$$

where $\boldsymbol{u}$ is a distributed variable in space ($\boldsymbol{x}$) and time (t), a_i are the time-dependent weight coefficients and $\boldsymbol{\varphi}_i$ are the spatially distributed basis functions on the domain $\Omega(\boldsymbol{x})$. The basis functions, or 'POD modes', ($\boldsymbol{\varphi}_i$) are commonly homogeneous and the $\boldsymbol{u}_0$ source function takes the inhomogeneous boundary conditions, akin to traditional spectral methods. This technique has been very successful in constructing low-dimensional dynamical models in fluid mechanics [15] and developing control schemes for distributed parameter systems (e.g., [24, 25, 33]) by projecting the governing equations onto the m-dimensional POD subspace, producing m coupled ordinary differential equations in time describing the weight coefficient (a_i) evolution. The POD has been previously limited to prototypical systems with simple geometries because inhomogeneous boundary conditions present difficulties. The resultant models are also only valid for the parameter (Reynolds or Nusselt number) value for which the system observations were made and the models typically produce very poor results outside that parameter value due to the highly nonlinear nature of the Navier-Stokes and RANS equations.

A new POD-based modeling methodology intended for parametric models of stationary turbulent flows, and especially for numerical simulations of the (RANS) equations, was developed in [30]. The method takes a data ensemble of steady solutions at different parameter values and parameterizes the POD modes as a function of the solution corresponding to the new parameter value using orthogonal complement subspaces. This is achieved by selecting the nearest observations to the new approximate solutions in terms of the parameter values. To solidify this concept, consider a turbulent flow model where the inlet mass flux is varied in a specified range. In the analysis of complex flows, the exact velocity profile on the boundary is often unknown. However, the design objectives are often based on integral conditions, such as the appropriate mass flux through a given set of surfaces, Γ, prompting the introduction of the mass flux function:

$$\boldsymbol{F}(\boldsymbol{u}) = \int_{\Gamma} \rho \boldsymbol{u} \cdot \boldsymbol{n} d\boldsymbol{x}, \qquad \Gamma \subseteq \Omega \tag{3}$$

The mass flux function $\boldsymbol{F}$ computes the mass fluxes through surfaces Γ given the fluid density ρ and the outward pointing surface normal, $\boldsymbol{n}$. Note $\boldsymbol{F}$ is generally a vector-valued function because Γ may define a set of discontinuous surfaces.

The modal weight coefficients are evaluated with a flux matching procedure that satisfies the appropriate integral form of the parameter value, eliminating the time-consuming projection step. A new approximate solution (denoted $\boldsymbol{u}^*$) is defined by an s-dimensional vector of mass flux goals ($\boldsymbol{G}$) corresponding to the desired mass flux through the set of control surfaces $\Gamma = \{\Gamma_1, \Gamma_2, \ldots \Gamma_s\}$, such that $\boldsymbol{G} = \boldsymbol{F}(\boldsymbol{u}^*)$. The solution procedure becomes an optimization problem to find the set of weight

coefficients that minimizes the mass flux error on the control surfaces Γ:

$$\min\left\{\left\|\boldsymbol{G}' - \sum_{i=1}^{m} a_i \boldsymbol{F}(\boldsymbol{\varphi}_i)\right\|\right\} \quad \text{where} \quad \boldsymbol{G}' = \boldsymbol{G} - \boldsymbol{F}(\boldsymbol{u}_0) \tag{4}$$

Equation (4) can be solved as $\boldsymbol{a} = \boldsymbol{F}^{+}(\boldsymbol{\varphi}_i)\boldsymbol{G}'(\boldsymbol{u}^*)$, where $\boldsymbol{F}(\boldsymbol{\varphi})$ is the $s \times m$ dimensional matrix obtained by operating Eq. (3) on the s control surfaces of the m POD modes and $()^{+}$ is a suitably defined generalized matrix inverse.

3.2 Rack reduced-order model

The rack observation database was created by varying the inlet velocity between 0.0 and 2.0 m/s on 0.25 m/s increments, $V_{\text{obs}} = \{0.0,\ 0.25, \ldots, 2.0\}$ m/s for a total of nine observations. Based on inlet hydraulic diameter and first nonzero observation $\boldsymbol{V}_{\text{obs}} = 0.25$ m/s, the Reynolds number of the observations ranges from $6136 \leq \text{Re} \leq 70{,}629$. To demonstrate the method, an approximate solution to $V_{\text{in}} = 0.33$ m/s is computed. This scenario corresponds to a rack placed in an existing data center facility with a fixed rack flow rate due to facility-level flow constraints imposed by the raised floor plenum and other data processing racks.

Figure 2 plots the POD modal streamline patterns because the flow is too complex to plot the velocity vectors to discern the flow patterns for the entire domain. The POD modes are solutions to the governing equations and each successive POD mode resolves higher order characteristics of the flow. Figure 3 plots the relative L^2 error over the entire domain for the test case, showing initially sharp error attenuation and final velocity field approximation error of less than 4%. Examining the absolute error field over the entire domain shows a region in the lower right hand corner of the domain where the only significant errors occur. Figure 4 plots the local velocity vectors for the full numerical model and the POD-based approximation in the region of maximum error to illustrate the approximation accuracy.

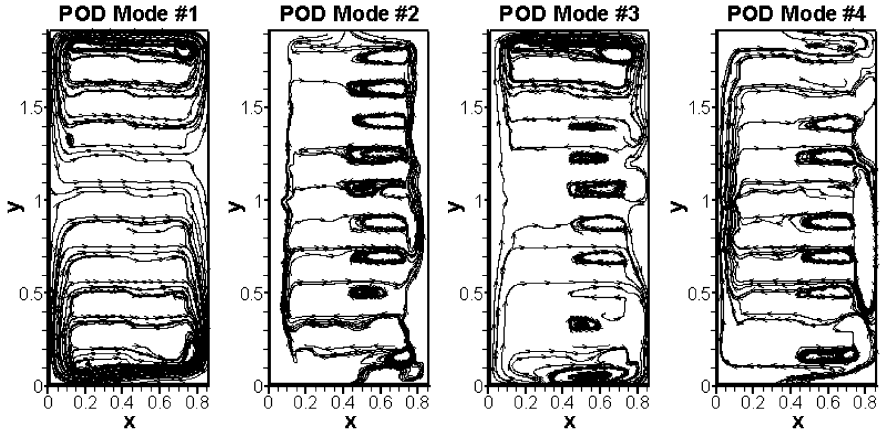


Fig. 2 POD modal streamlines

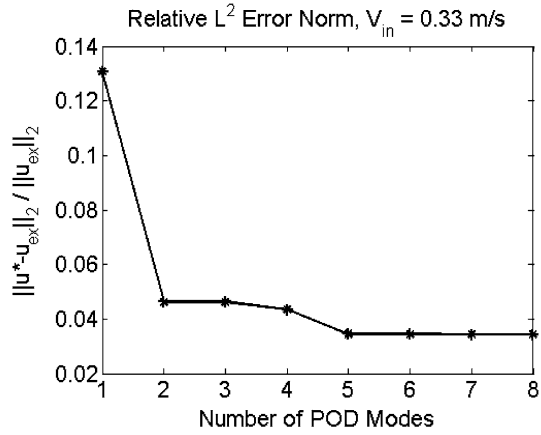


Fig. 3 Approximate solution relative L^2 error norm over the entire domain

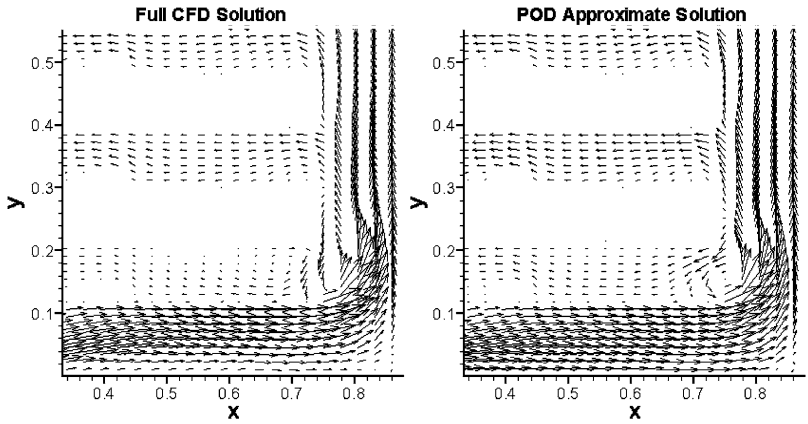


Fig. 4 Local exact solution from full CFD model and POD-based approximation projected to a coarser grid than the actual model for illustrative purposes

The model reduction procedure was performed on the velocity field only because the solution of the RANS momentum equations consumes significantly longer time relative to solving the energy equation in forced convection. A RANS energy equation solver was developed to compute the temperature field given the reduced-order velocity solution.

In validating the temperature solution solver against full numerical solutions, the maximum chip temperature responses showed the servers could be grouped into three sections (cf. Fig. 1) with the lower servers forming section 'A', servers 3 through 5 forming section 'B' and the top servers 6 though 10 forming section 'C'. The quantities Q_A, Q_B, and Q_C denote the heat generation of each processor in the respective cabinet section. This partitioning is not a required step, but it is chosen to reduce the number of control variables from 10 to 3 in this illustrative example. The model post-processes the temperature field to obtain the maximum chip temperature for each server, ultimately producing a one-dimensional output vector. The reduced-order model program flow and computational times for each subroutine are shown in Fig. 5. Note the full

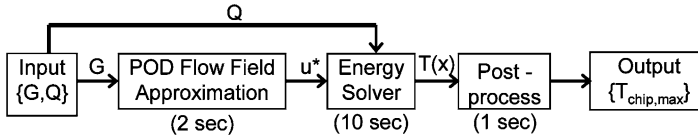


Fig. 5 Reduced-order model program flow and computational time for a single function call

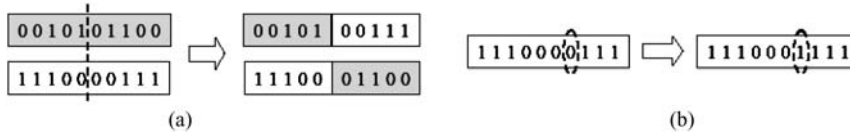


Fig. 6 (a) Crossover, and (b) mutation operations

numerical model requires approximately 1200 s to converge on a 2.8 GHz Xeon processor workstation with 2 GB of memory whereas the POD-based approximate solution requires 13 seconds.

4 Background and proposed optimization approach

4.1 Background on single- and multi-objective genetic algorithms

Genetic algorithms (GAs) were originally developed for global optimization of unconstrained single-objective optimization problems. GAs are stochastic and population based optimization methods whereby the design space is searched through selected design points using random genetic operations. In a GA, each point in the design space is coded as a binary string and assigned a fitness value that reflects its goodness in terms of design objective function value. GA creates a new generation of design points through two important operations: crossover and mutation. In a crossover operation, as shown in Fig. 6(a), two design points are selected as parents and then portions of their chromosomes (representing design alternatives) are swapped to produce offspring design points. For diversity, as shown in Fig. 6(b), a mutation operation is performed in that a bit is randomly selected and changed from 0 to 1 or vice versa.

In GAs, design points are also "inherited" in that a subset of design points that have good fitness values are chosen to migrate into the next generation. Together crossover, mutation, and inheritance provide a powerful search mechanism for finding a global optimum design point [10]. As shown in Fig. 7, a GA starts with an initial population of design points that are often randomly generated. Next, design points in the population are evaluated and their fitness assigned. After that, a stopping criterion is checked to see whether the fitness of design points can be further improved or whether a predefined number of iterations has been reached. If the stopping criterion is met, then the GA is stopped; otherwise, crossover, mutation, and inheritance operations are applied to produce a new generation of design points and the procedure is repeated. For convenience, in this paper, the process of producing new design alternatives is referred to as "reproduction."

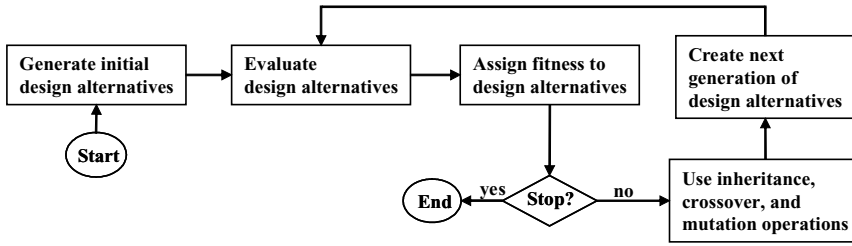
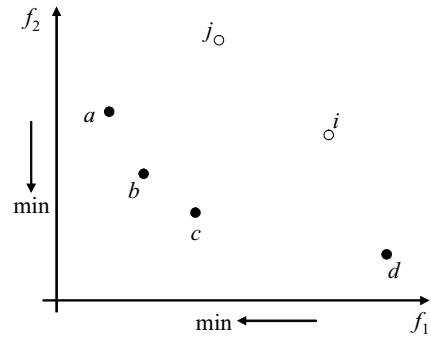


Fig. 7 Steps for a genetic algorithm

Fig. 8 Domination relation in MOGA



GAs can be easily modified to handle multi-objective optimization problems. Multi-objective optimization version of the GAs are referred to as Multi-Objective Genetic Algorithms (MOGAs) (e.g., [10, 18, 23]). MOGAs are capable of handling multiple objectives by defining fitness in a multi-objective sense. For instance, Deb [10] used a non-dominated sorting scheme, NSGA, in which the domination relations among design points are established pair-wise for each design point.

Figure 8 shows an example, whereby both objectives are being minimized (as illustrated by the two arrows on the two axes), with a population of six design points. Four of these design points $\{a, b, c, d\}$ are non-dominated. That is, each of these design points is strictly dominant (or better) in terms of at least one objective function value and at the same time is not inferior (or not worse) in terms of the other objective function over any other design point. So, in the current generation, design points $\{a, b, c, d\}$ form the non-dominated set, and the design points $\{i, j\}$ form the dominated set. MOGAs can also handle multi-objective optimization problems with constraints. A common approach for handling constraints is the penalty method (e.g., [9, 18]) whereby the constraints are combined with the objective functions for fitness assignment.

The MOGA used in this paper is implemented based on the non-dominated sorting algorithm [10] together with an elitism strategy similar to SPEA [41], and is considered as a conventional MOGA in comparison to the proposed new MOGA approach in this paper. In the conventional MOGA, the fitness value of each point is assigned by a non-dominated sorting technique [10]. Then the non-dominated points in the current generation are directly copied into the next generation as elite points. A diversity measure is used to choose a subset of the non-dominated points if the number of those

points is more than 70% of total number of design points in the current population. This elite-preservation strategy is similar to SPEA [41]. According to the results from [21], this conventional MOGA can converge to the true Pareto frontier for multi-objective optimization problems, and its performance is comparable to NSGA-II [11] or SPEA [41].

4.2 Background on kriging-based meta-modeling

Over the past several years kriging (e.g., [17, 37]) has become popular for meta-modeling in the field of engineering design because of its general predictive modeling ability. Kriging is a linear predictor estimating unknown response values based on all the observed response values. That is, the predicted value at any design point is a linear combination of all observed points. Often, the input points for observed (or true) responses are referred to as experiments.

To simplify the description of kriging, we consider a single-input (x) single-response (y) simulation model. A kriging method treats the deterministic response y of the simulation as a realization of a stochastic process Y:

$$Y = \mu + Z(x) \tag{5}$$

where Y is the approximation of a true response y, μ is a constant representing the known prior information (i.e., mean of the responses based on the current meta-model), x is the input, and Z is assumed to be a stochastic process with zero mean and non-zero variance σ^2.

It is assumed that the function (or simulation) to be approximated is continuous and thus the responses will not change significantly for two adjacent design points in the input space. In practice, this assumption can be somewhat relaxed and replaced with a weaker assumption, i.e., for two input vectors x_i and x_j, the responses $y(x_i)$ and $y(x_j)$ should be close to each other if the distance between x_i and x_j is small. In other words, one can state that the random variables $Y(x_i)$ and $Y(x_j)$ tend to be highly correlated if the distance between x_i and x_j, $\| x_i - x_j \|$, is small. The correlation "*Corr*" between any two random variables $Y(x_i)$ and $Y(x_j)$ can be calculated by a variety of correlation functions [22]. The normal correlation function [37] is the most commonly used:

$$Corr[Y(x_i), Y(x_j)] = \exp(-\theta \|x_i - x_j\|) \tag{6}$$

where θ is a parameter determined by the degree of the correlation among responses. The larger the value of θ the weaker the correlation and vice versa. In the rest of this section, we assume the normal correlation function, i.e., all equations are based on this assumption.

Suppose now we have n experiments: $x_1, x_2, \ldots, x_n$, and $\bar{y}$ is the set of n corresponding observed true responses, R is a $n \times n$ correlation matrix with the (i, j) element given by Eq. (6), r is a $n \times 1$ correlation matrix for $\bar{y}$ and an un-observed x_0, and J as a $n \times 1$ vector of ones. Then the kriging estimate $Y(x_0)$ and the corresponding kriging variance $s^2(x_0)$ can be obtained using the following equations:

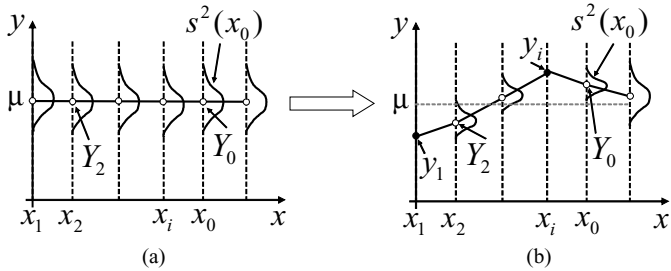


Fig. 9 Kriging meta-modeling: (a) before any experiment is observed, and (b) after two experiments (x_1 and x_i) are observed

$$Y(x_0) = \mu + r^T R^{-1}(\bar{y} - J\mu) \tag{7}$$

$$s^2(x_0) = \sigma^2 \left[1 - r^T R^{-1} r + \frac{(1 - r^T R^{-1} r)^2}{J^T R^{-1} J} \right] \tag{8}$$

The kriging meta-modeling is also illustrated in Fig. 9. Figure 9(a) shows the prior distributions of Y's with the given mean of the stochastic process and without any observation of the response. On the other hand, Fig. 9(b) shows the posterior distributions based on the observing the responses for two experiments (x_1 and x_i). The kriging method updates the estimates of other responses, i.e., it revises the prior distribution of Y's (notice the difference between Y_2 in Fig. 9(a) and Y_2 in Fig. 9(b)). The change of kriging variance $s^2(x_0)$ is also illustrated in Fig. 9.

The kriging variance $s^2(x_0)$ can serve as a measure of uncertainty for an estimated response. Clearly, for observed experiments (e.g., x_1 and x_i of Fig. 9), the kriging variance is zero. On the other hand, there is a non-zero kriging variance for any unobserved experiment (e.g., x_0, see Fig. 9(b)). The further away is the design point x_0 from current experiments, the higher the value of the kriging variance. When Eq. (5) is assumed to be a normal process, the kriging variance is a conditional normal distribution with zero mean and $s^2(x_0)$ as the variance [40]. Thus, the kriging variance can be converted to a "prediction interval" with a 95% confidence level and the range of the interval equals to $4s(x_0)$ (or $\pm 2s(x_0)$). This prediction interval is used to measure the uncertainty of a kriging meta-model estimate for the POD model response in the proposed optimization approach.

4.3 Proposed optimization approach

In the proposed optimization approach, some design points in a new generation (or new population) are produced using kriging meta-model of the simulation while the rest are produced by conventional MOGA operations, i.e., cross-over, mutation and inheritance operations. The proposed approach has the following two main features. First, the approach uses a criterion that measures the accuracy of kriging meta-models for a simulation. Second, the proposed approach uses a kriging-assisted design point selection strategy based on the accuracy of the kriging meta-models. In other

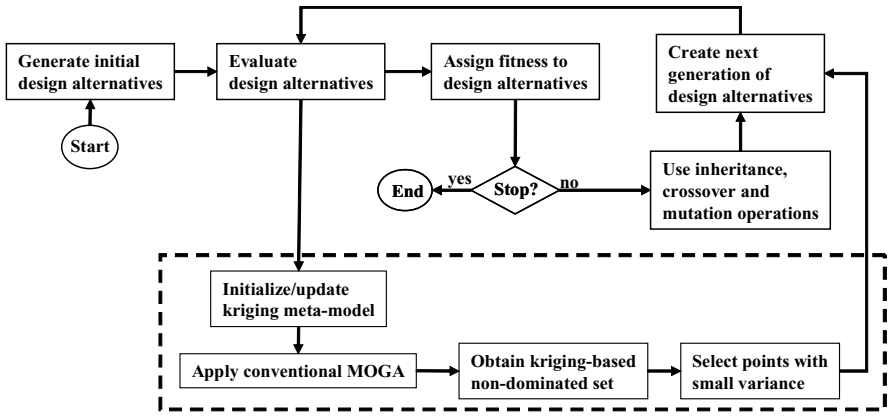


Fig. 10 Main steps for the proposed approach

words, a kriging meta-model is used to estimate the Pareto frontier and then select design points from this so called kriging based non-dominated design points. These selected design points are considered to have a higher probability to be eventually on the Pareto frontier compared to those points not in the set of kriging based non-dominated design points. This is because the estimated response for these points has a reasonable accuracy. The remaining design points in the population are produced using conventional MOGA operations. Since the new population is produced in such a hybrid manner, the computational cost of converging to the Pareto frontier can be significantly reduced compared to a conventional MOGA. The main steps of the proposed approach are shown in Fig. 10.

Figure 10 is similar to Fig. 7 except that in Fig. 10 some design points in a new generation are produced based on kriging meta-models (steps that are inside the dashed box in Fig. 10). Also, the two output arrows from the block: "Evaluate design alternative" in Fig 10 implies that the true responses from the simulation are used for both: (i) updating the kriging meta-model, and (ii) assigning fitness to design alternatives. As shown in the dashed box, for each generation, kriging meta-models are constructed for the original simulation (i.e., for the objective and/or constraint functions). Next, a conventional MOGA is applied to the kriging-based meta-models of the optimization problem to obtain a kriging-based non-dominated set. Among the design points in this set, those with a small kriging variance (or small prediction interval) are selected for the next generation of design points.

A criterion determining how small the interval should be is defined in Section 4.3.2 by deriving a relationship between the prediction interval and a measure of domination relation in objective space, which is defined by *Minimum of Minimum Distances* (*MMD*) (initially introduced in [21]) and is discussed in Section 4.3.1. Finally, a new kriging guided reproduction approach using the relation between *MMD* and prediction interval is proposed in Section 4.3.3.

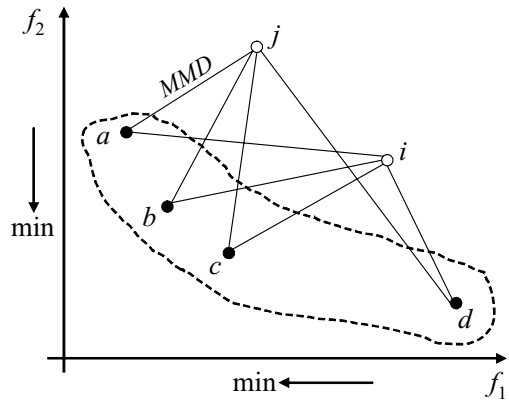


Fig. 11 Minimum of minimum distances

4.3.1 Minimum of minimum distances

In each generation (after the initial population), the kriging meta-models can be used to obtain the predicted response for each design point. There is a kriging variance for each prediction. For an un-observed design point, this variance is non-zero and a corresponding prediction interval can be obtained; while the kriging variance is zero for an observed design point. Based on the predicted objective function values, the domination relation can be determined for a set of observed and/or unobserved design points, as shown in Fig. 11. Domination relation implies that the population is partitioned into two sets: dominated set and non-dominated set. Notice that this partitioning is based on predicted values, that is, no original function calls are used (for unobserved design points) so far in the current generation.

A quantitative measure of domination is defined in the objective space. This measure is referred to as *minimum of minimum distance* (*MMD*). The quantity *MMD* refers to the minimum distance of all distances between any non-dominated design point (e.g., design point "*a*" in Fig. 11) and any dominated design point (e.g., design point "*j*" in Fig. 11). For instance, in Fig. 11 *MMD* is obtained as follows:

(1) Calculate the minimum distance ("*dis*") from each non-dominated design point to all dominated design points:

$$\begin{aligned} \min(dis_{ai}, dis_{aj}) &= dis_{aj} \\ \min(dis_{bi}, dis_{bj}) &= dis_{bi} \\ \min(dis_{ci}, dis_{cj}) &= dis_{cj} \\ \min(dis_{di}, dis_{dj}) &= dis_{dj} \end{aligned} \tag{9}$$

(2) Obtain *MMD* as the minimum of the above minimum distances:

$$MMD = \min(dis_{aj}, dis_{bi}, d_{cj}, d_{dj}) = d_{aj} \tag{10}$$

The value of *MMD* determines the lower bound of distance between any two design points whereby one of the points is in the non-dominated set while the other is not.

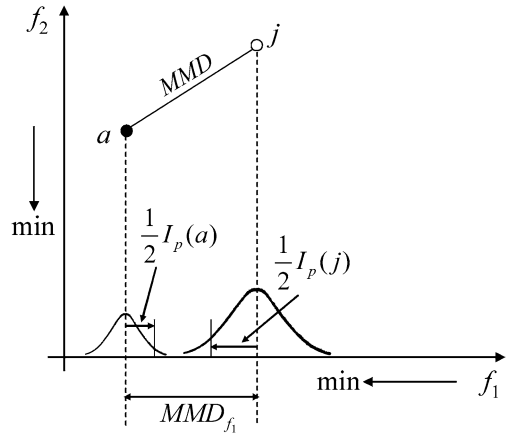


Fig. 12 Relation between *MMD* and *prediction interval*

Since a meta-model is constructed separately for each objective function, *MMD* is projected to each dimension in objective space as: $MMDf_m$ $(m = 1, \ldots, M)$ so that the *MMD*'s projected values can be used in the corresponding meta-models.

4.3.2 Relation between prediction interval and MMD

As mentioned before, the estimated response from a kriging model has a variance which can be converted to a prediction interval. As long as the values of prediction interval do not affect the domination relation (which is represented as a distance quantitatively) of two design points, it is considered that the kriging meta-model outputs are accurate estimates for the true responses for these two design points. Hence, the criterion that relates *MMD* and prediction interval can be defined by checking whether or not the domination relation is changed when kriging estimates are used. If the domination relation is changed, then the design points that contribute to this change is evaluated using the original simulations. Those design points that do not contribute to the change of the domination relation can be estimated using kriging meta-models. An implementation of the aforementioned idea is shown in Fig. 12.

To obtain the relation between *MMD* and prediction interval, for simplicity, we examine the situation in one dimension in the objective space (e.g., along objective function f_1). Because the prediction interval is symmetrical as described in Section 4.2, we only need to examine one half of the interval for each design point involved in the evaluation. In the worst case, both half-intervals diminish the *MMD* projection, as shown in Fig. 12. We define the prediction interval for design point j as $I_P(j)$ (note that $I_P(j) = 4s(j)$ for a 95% confidence level). It can be observed from Fig. 12 that as long as the sum of half prediction intervals for design points a and j does not exceed the corresponding *MMD* projection, the domination relation of a and j along objective f_1 will not be changed even if the kriging estimates for both design points are used. This relation between *MMD* and prediction interval can also be expressed by the following condition:

$$I_p(a) + I_p(j) \leq 2MMD_{f_1} \tag{11}$$

It can be seen from Eq. (11) that if the value of the half of the prediction interval of a design point is greater than *MMD*, this design point must be evaluated using the simulation. Note that this relation should hold for any design point that can be estimated using kriging meta-models so that the domination relations for all design points in the current population are not changed. For those design points that can be estimated using kriging meta-models, the kriging-based estimated responses are considered to be accurate enough to ensure a correct domination relation in the current population. The aforementioned condition, Eq. (11), is achieved in a three-stage procedure including shrinking, sorting, and selecting stages. This procedure is applied in each generation of a MOGA to determine the number of design points that can be estimated using kriging meta-models. A description of the three-stage procedure follows. (To simplify the exposition, we consider a single-input (x) case. The procedure can be readily extended to multi-input case, as demonstrated by the data center cabinet example.)

In the first stage (i.e., the shrinking stage), design points with zero prediction interval I_p and design points with a large value of the prediction interval (i.e., design points with prediction interval greater than twice the value of *MMD*, recall Eq. (11)) are removed from the population. Note that the points with zero prediction have already been evaluated while those with a large value of the prediction interval are to be evaluated. After the removal of these two kinds of design points, the size of the population becomes smaller than the original population. We refer to this reduced size population as a "shrunk population". We define the population size as n and the shrunk population size as n_s. Since some design points in the current population are inherited, the number of design points with a zero prediction interval is always greater than zero. That is, $n_s < n$, and we will work on a set with fewer design points than the original population.

In the second stage (i.e., sorting stage), the design points in the shrunk population are sorted based on their prediction interval values, from large to small. In other words, the first design point x_1 in the set has the largest value of the prediction interval $I_p(x_1)$ while the last design point x_{n_s} has the smallest value of the prediction interval $I_p(x_{n_s})$, as shown in Eq. (12).

$$2MMD_{f_1} \geq I_p(x_1) \geq \cdots \geq I_p(x_{n_s}) > 0 \tag{12}$$

where x_i is a design point in the sorted shrunk population, $i = 1, \ldots, n_s$.

In the last stage (i.e., selecting stage), the design points in the sorted shrunk population are examined, beginning from the first design point x_1, to see whether or not it satisfies the following condition:

$$I_p(x_i) + I_p(x_{i+1}) \leq 2MMD_{f_1} \tag{13}$$

If the condition in Eq. (13) is not satisfied for a design point x_i, then that design point is removed from the sorted shrunk population and evaluated using the simulation. On

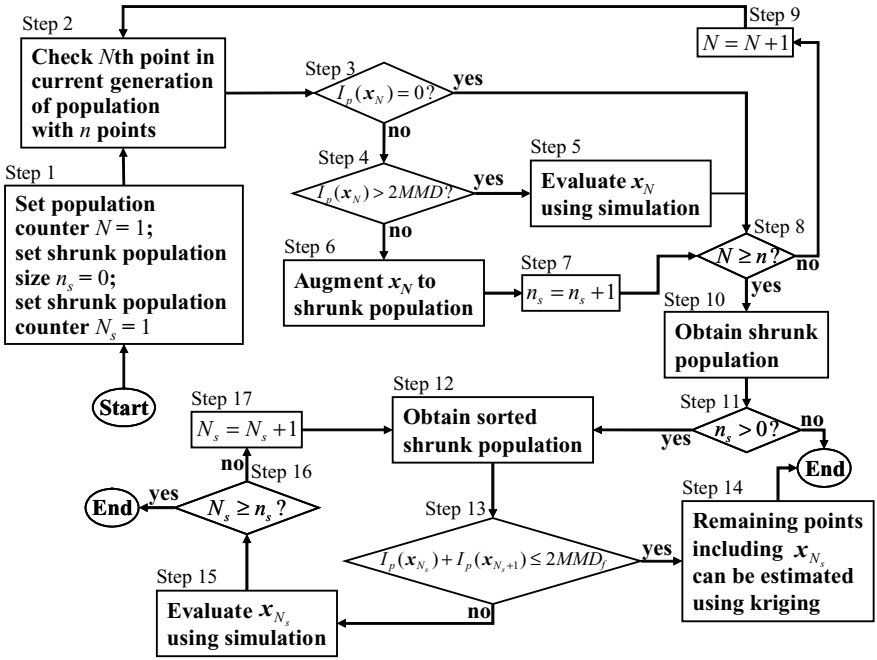


Fig. 13 Shrinking, sorting, and selecting algorithm for one generation of design alternatives

the other hand, if the condition in Eq. (13) is satisfied, then following Eq. (12) we have:

$$2MMD \geq I_p(x_i) + I_p(x_{i+1}) \geq I_p(x_i) + I_p(x_{i+2}) \geq \cdots \geq I_p(x_i) + I_p(x_{i+n_s-1}) \tag{14}$$

which implies that the design point x_i as well as all remaining design points in the sorted shrunk population are to be estimated using the kriging metal-model for the simulation.

This three-stage procedure is presented in detail in a flowchart in Fig. 13, N is the counter for the original population ($N = 1, \ldots, n$), and N_s is the counter for the shrunk population ($N_s = 1, \ldots, n_s$). A step-by-step description of the procedure is also given as follows.

Step 1: Initialize counters: $N = 1,\ n_s = 0,\ N_s = 1$.
Step 2: Examine the value of prediction interval $I_p(x_N)$ of the Nth design point in current generation.
Step 3: If the value of prediction interval $I_p(x_N)$ for the Nth design point is zero (i.e., this design point has been evaluated using simulation), go to Step 8.
Step 4: If the value of prediction interval $I_p(x_N)$ for the Nth design point is greater than $2MMD$ (i.e., this design point has a large variance and needs to be evaluated using simulation), go to Step 5.
Step 5: Evaluate this design point using simulation, and go to Step 8.

Step 6: If conditions in Steps 3 and 4 are not met, augment this design point into the shrunk population.
Step 7: Set $n_s = n_s + 1$, go to step 8.
Step 8: Check whether all design points in current population are examined: if yes, go to Step 10; otherwise, go to Step 9.
Step 9: Set $N = N + 1$, and repeat Steps 3 to 7 for the Nth design point.
Step 10: Form the shrunk population with design points obtained from the previous steps.
Step 11: Examine whether there is any design point in the shrunk population: if yes, go to Step 12; otherwise, end the procedure (i.e., all the design points with non-zero prediction interval are to be evaluated using simulation).
Step 12: Sort the design points in the shrunk population by the value of prediction interval (from large to small) to obtain a sorted shrunk population.
Step 13: Examine the N_sth design point with the criterion described in Eq. (13): if satisfied, it means this design point as well as remaining design points in the sorted shrunk population can be estimated using kriging meta-models; otherwise, go to Step 15.
Step 14: Form the set of design points that can be estimated using kriging meta-models.
Step 15: Evaluate the N_sth design point using simulation.
Step 16: Check whether all the design points in the sorted shrunk population have been examined: if yes, end the procedure; otherwise, go to Step 17.
Step 17: Set $N_s = N_s + 1$, and go back to Step 12.

This procedure is repeated for all objective functions in the simulation. (Recall from Section 4.1 that the constraint functions can be embedded into the objective functions using a penalty approach.) If a design point is selected to be evaluated using kriging for all objective functions, then that point is estimated using kriging; otherwise, that design point is evaluated using the simulation. Following the above procedure, the design points in the current population can be divided into three groups in terms of the value of their prediction interval. The first group of design points refers to those that are selected by the above mentioned three-stage procedure. The second group of design points refers to those design points that have non-zero prediction interval and are not selected by the above mentioned procedure. Finally, the third group refers to those design points that have a zero prediction interval and also not selected by the above mentioned procedure. Note that the third group has already been evaluated using simulation and their true responses have already been used to construct kriging meta-models.

We define the ratio of the number of design points that should be evaluated using kriging (first group) over the number of design points with non-zero prediction interval (first and second groups), in the current population, as $ratio_k$. This ratio can be used as a good estimate of the meta-model accuracy for the current kriging meta-models. This is because the number of design points that can be evaluated by the kriging meta-model is directly dependent on the meta-model accuracy which is proportional to the value of $ratio_k$. The quantity $ratio_k$ is used in the proposed approach as a percentage at which some design points from the set of kriging based non-dominated design points are selected for new population.

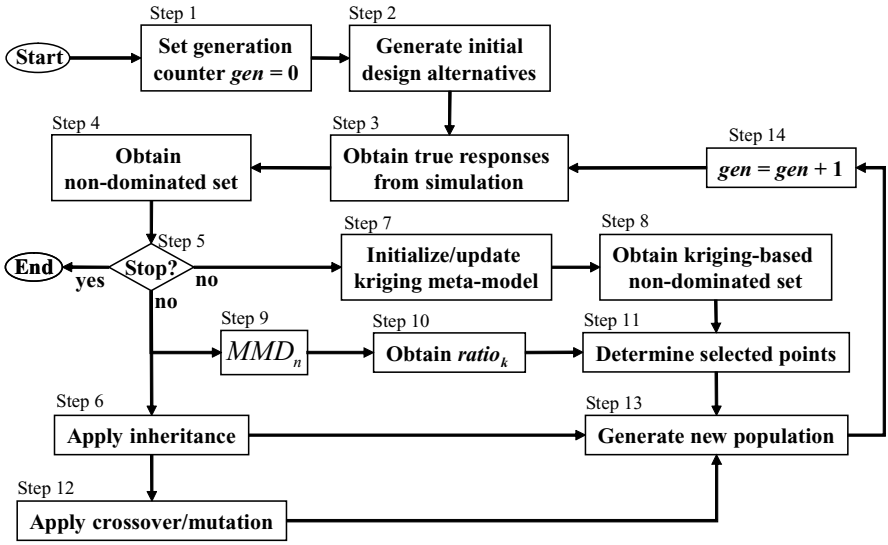


Fig. 14 Kriging guided reproduction approach

4.3.3 Kriging guided reproduction approach

The proposed kriging guided reproduction approach has four operations. These include the three conventional genetic algorithm operations (i.e., cross-over, mutation and inheritance) and a newly defined kriging guided operation. The kriging guided operation has two major components. The first component involves obtaining a kriging-based non-dominated set. The second component involves selecting some design points from the obtained set according to $ratio_k$. The stopping criterion used in this paper is that the procedure stops when the number of non-dominated points exceeds a pre-specified threshold (e.g., 85%) of the total population for several consecutive generations.

A step-by-step description of this approach is given as follows, and also presented as a flowchart in Fig. 14.

Step 1: Set generation counter to $gen = 0$.
Step 2: Randomly generate an initial set of design alternatives.
Step 3: For design alternatives, use simulation to obtain true responses.
Step 4: Based on the true responses, obtain a non-dominated set.
Step 5: Check stopping criterion. If satisfied, stop. Otherwise, continue to the next step.
Step 6: Apply inheritance, i.e., migrate the non-dominate solutions from the last generation. (It is recommended that at most 85% of design points in the current generation is obtained from inheritance. Steps 7 to 11 are applied only if this percentage is less than 85%; otherwise, the procedure continues to Step 12.)
Step 7: Initialize (for the initial iteration) or update (for the remaining iterations) the kriging meta-models using simulation responses.

Step 8: Obtain kriging-based non-dominated design points. (As mentioned at the beginning of Section 4.3, this step involves an inner optimization wherein MOGA is applied to the current kriging based meta-models.). Use the three-stage procedure (recall Section 4.3.2) to obtain an ordered set of kriging-based non-dominated design points.
Step 9: Obtain *MMD* for the current generation.
Step 10: Calculate $ratio_k$. (It is recommended to set $ratio_k$ to 100% for $gen \leq 4$.)
Step 11: Use $ratio_k$ to select a number of design points with small value of variance from the ordered set of kriging-based non-dominated design points.
Step 12: Augment the current generation by creating design points using the cross-over and mutation operations. (It is recommended that at least 15% of the design points in the current generation are created by these operations.)
Step 13: Combine the design points obtained from inheritance, cross-over, mutation, and kriging meta-models.
Step 14: Reset the generation counter: $gen = gen + 1$, and go to Step 3.

Note that the proposed approach is different from the one introduced in [21], although both of which use the concept of *MMD*. The emphasis in the proposed approach is on searching the design space efficiently, while the approach in [21] focuses on estimating design alternatives in each generation.

In the next section, the proposed approach is applied to the data center cabinet thermal design problem introduced in Section 3, which is formulated as a constrained multi-objective optimization problem. The results from the proposed approach are compared with those from a conventional MOGA (e.g., [10, 18]), and discussions follow to explain the obtained results.

5 Results and discussions for the cabinet case study

In this case study, we consider the optimal thermal design for a cabinet where the emphasis is on optimizing the layout of heat generating components within the cabinet. Based on the simulation model for the cabinet thermal design, we formed a two-objective design optimization problem. In this thermal optimization model, inlet air velocity V_{in} (in m/s) and chip heat generation rates (in W) Q_j, ($j = 1, \ldots, 10$) are continuous design variables, where $Q_1 = Q_2, Q_3 = Q_4 = Q_5$ and $Q_6 = \cdots = Q_{10}$ (as shown in Fig. 1). The output of the simulation model is a 10-element vector of temperatures, one for the maximum temperature in each server. The objectives are to: (1) minimize the maximum of chip temperatures (in °C), and (2) maximize the sum of total heat generation rates. In this regard, the two design objectives are correlated and conflicting. That is, reducing the maximum temperature (which is preferred) requires reducing the total heat generated by the cabinet servers (which is not preferred). Note that the maximum temperatures (T_j) optimized here can be used to represent the server inlet temperatures. In other words, when T_j goes up, the corresponding server inlet temperature also goes up. Also, Max(T_j), $j = 1, \ldots, 10$, is used in the constraints, which requires that maximum chip temperature for each server to be less than 85°C.

The optimization problem is formulated as follows:

$$\underset{V_{in},Q_{a,b,c}}{\text{Minimize}}\ \text{Max}(T_j) \quad j = 1,\ldots,10$$

$$\underset{V_{in},Q_{a,b,c}}{\text{Maximize}}\ \text{Sum}(Q_j) \quad j = 1,\ldots,10$$

subject to

$$\text{Max}(T_j)/85 - 1 \leq 0 \qquad j = 1,\ldots,10 \tag{15}$$

where

$$0.2 \leq V_{in} \leq 2$$

$$2 \leq Q_j \leq 200$$

An alternative data center optimization problem is to fixed the net power dissipation and vary the server-level power dissipation and flow rate in order to minimize the chip temperatures, T_j. Such an investigation was performed using robust design methods [34].

The proposed MOGA approach and a conventional MOGA are used to obtain the solutions for the problem in Eq. (15). Figure 15 illustrates the Pareto optimal solutions from the conventional MOGA and the proposed approach. We used 30 design points per generation for both approaches. According to the stopping criterion described in Section 4.3.3, the new approach requires about 20 generations to obtain Pareto optimum design solutions, while the conventional MOGA needs about 50 generations to obtain a comparable Pareto optimum design solutions. As shown in Fig. 15, the results from the proposed approach are in good agreement with those from the

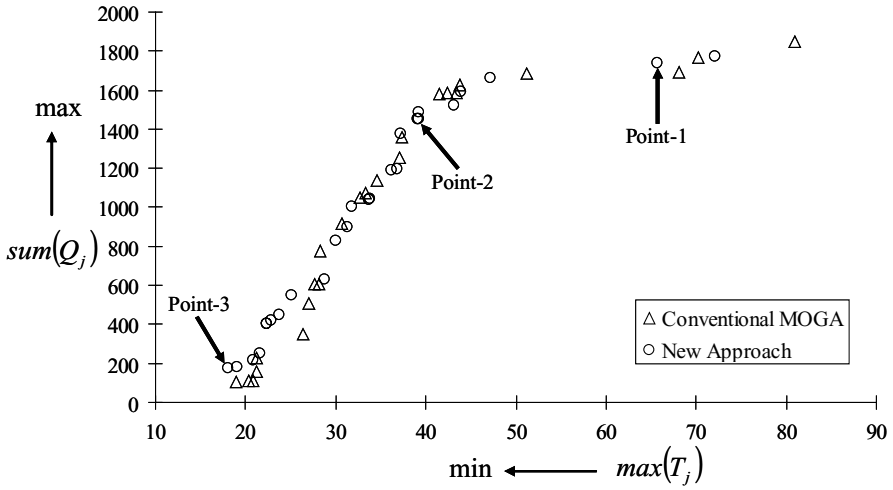


Fig. 15 Pareto front from conventional MOGA and new approach

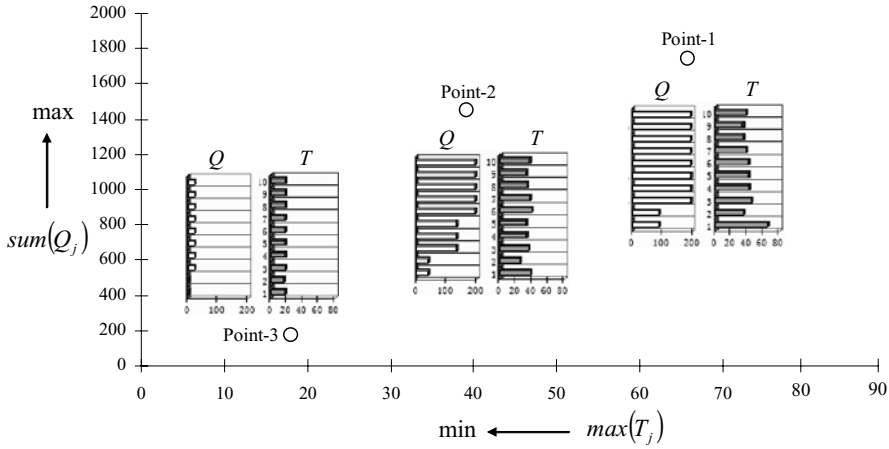


Fig. 16 Design objective function values for selected Pareto design points from new approach

conventional MOGA. In the Pareto design points obtained from the new approach, three optimized design alternatives are selected and their corresponding design objective function values are shown in Fig. 16.

The results in Figs. 15 and 16 show several interesting and unexpected features. First of all, due to the airflow bypass patterns created near the lower servers, their maximum power dissipations are significantly lower than the upper servers as shown in Fig. 16. Of course, it would be possible to re-design the flow patterns by placement of baffles. Secondly, it is also seen that the maximum total power dissipation with this example air-cooled geometry is limited to $\sim$1,800 W (as shown in Fig. 15), beyond which the maximum temperature specification cannot be met. It is extremely useful to deduce such limits for various cooling solutions, so that one could determine the overall thermal management approach for a given total cabinet power dissipation. Finally, due to the rather flat shape of the Pareto Front in Fig. 15 it is seen that proper placement of servers at the higher end of total powers is very important in reducing temperatures.

Note that the corresponding temperatures for some extreme inputs of the model are not necessarily optimal or even feasible. For example, as shown in Fig. 17, when

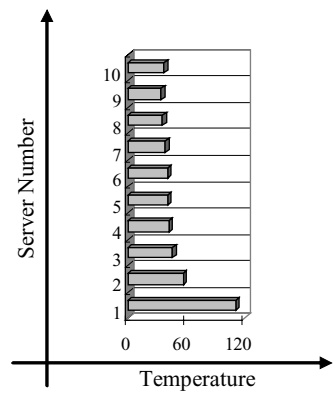


Fig. 17 Temperature profile for an extreme (infeasible) design

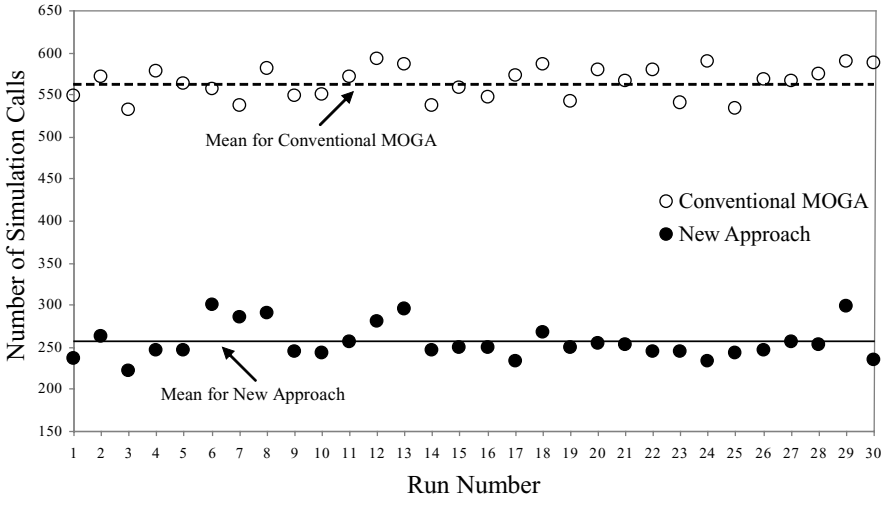


Fig. 18 Number of simulation calls in 30 runs for conventional MOGA and new approach

$V_{in} = 2.0$ m/s and $Q_i = 200$ W, the corresponding temperature for server 1 violates the temperature constraint (i.e., junction temperature for server 1 exceeds 85°C) as given in Eq. (15), thus the obtained solution is infeasible. This demonstrates that the optimal solutions are far from being trivial or obvious, which further shows the necessity for using an optimization technique such as the one proposed in this paper.

To account for stochastic nature of MOGA, we ran the new MOGA and the conventional MOGA for this problem 30 times with identical initial population of design points. The obtained results are depicted in Fig. 18. It can be observed that the number of simulation calls has been significantly reduced when the new approach is used. The mean for the number of simulation calls are obtained to be 255 for the new approach and 565 for the conventional MOGA. Hence, for this example, on the average 55% (i.e., $1 - 255/565 = 0.55$) simulation calls are saved when the new approach is used compared with the conventional MOGA. Also, the results from the new approach have a similar standard deviation (i.e., 20.7) compared to that from the conventional MOGA (i.e., 18.8). Since these two values are close to each other, the performance of the new approach can be considered to be almost as robust as the conventional MOGA.

6 Concluding remarks

Multi-objective design optimization model of an enclosed data center cabinet has been formulated and solved. For this model, the design objectives are: (i) minimizing the maximum chip temperature, and (ii) maximizing the total heat dissipations, over all servers in a modeled air-cooled cabinet. Appropriate bounds on the temperature and design variables (V_{in} and Q_j) form the constraints for this constrained multi-objective design optimization problem. To facilitate the optimization task, a POD-based reduced-order model of the data center cabinet is constructed first. The POD

model is used to substitute the original simulation model for the cabinet problem and forms objective and constraint functions of the optimization model. Next, this optimization model is integrated and solved with the new MOGA. The new MOGA uses a kriging guided operation in addition to the conventional genetic algorithm operations such as inheritance, crossover, and mutation to efficiently search the design space for optimal design solutions.

It is observed that the application of the new MOGA to thermal design of a data center cabinet obtains Pareto optimal design solutions that are very close to those obtained from a conventional MOGA. Furthermore, the number of simulation calls needed in the new MOGA approach to obtain the optimal solutions is about 55% fewer than that in the conventional MOGA. This makes the new approach attractive in handing multiple cabinets that interact with each other and are housed in a data center or similar facility. Such problems are currently outside the demonstrated capability of optimal design, due to the large computational requirements.

Finally, while in this paper the new MOGA is only demonstrated with a data center cabinet thermal design problem, the approach is general and can be applied to other multi-objective engineering design optimization problems.

Acknowledgments The authors acknowledge the support for this research in part through the Office of Naval Research: G. Li, M. Li and S. Azarm under contract number N00014-05-1-0529, J. Rambo and Y. Joshi under contract number N00014-04-1-0335, both monitored by Dr. Mark Spector.

References

1. K. Abboud and M. Schoenauer, "Surrogate deterministic mutation: Preliminary results," Lecture Notes in Computer Science, vol. 2310, pp. 104–116, 2001.
2. K.S. Anderson and Y. Hsu, "Genetic crossover strategy using an approximation concept," in Proc. of IEEE Congress on Evolutionary Computation, Washington DC IEEE, 1999, pp. 527–533.
3. J.S. Arora, Introduction to Optimum Design, 2nd eds., Elsevier, New York, USA, 2004.
4. ASHRAE, Datacom Equipment Power Trends and Cooling Applications, American Society of Heating, Refrigeration and Air-Conditioning Engineers, Technical Committee TC9.9, Atlanta, GA, 2005.
5. J.-F.M. Barthelemy and R.T. Haftka, "Approximation concepts for optimum structural design—A review," Structural Optimization, vol. 5, pp. 129–144, 1993.
6. M.S. Bazaraa, H.D. Sherali, and C.M. Shetty, Nonlinear Programming: Theory and Algorithms, John Wiley & Sons, New York, USA, 1993.
7. A.D. Belegundu and T.R. Chandrupatla, Optimization Concepts and Applications in Engineering, Prentice Hall, New Jersey, USA,1999.
8. C.A. Coello Coello, D.A. Van Veldhuizen, and G.B. Lamont, Evolutionary Algorithms for Solving Multi-Objective Problems, Kluwer Academic Publishers, Boston, USA, 2002.
9. D.W. Coit, A.E. Smith, and D.M. Tate, "Adaptive penalty methods for genetic optimization of constrained combinatorial problems," INFORMS Journal on Computing, vol. 8, pp. 173–182, 1996.
10. K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, New York, USA, 2001.
11. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA—II," IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182–197, 2002.
12. J.H. Ferziger and M. Peric, Computational Methods for Fluid Dynamics, 3rd edn., Springer, New York, 2002.
13. D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, Mass., USA, 1989.
14. J. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, Michigan, USA, 1975.

15. P. Holmes, J.L. Lumley, and G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge University Press, Great Britain, 1996.
16. Y. Jin, "A comprehensive survey of fitness approximation in evolutionary computation," Soft Computing, vol. 9, pp. 3–12, 2005.
17. D.R. Jones, "A taxonomy of global optimization methods based on response surfaces," Journal of Global Optimization, vol. 21, pp. 345–383, 2001.
18. A. Kurpati, S. Azarm, and J. Wu, "Constraint handling improvements for multi-objective genetic algorithms," Structural and Multidisciplinary Optimization, vol. 23, no. 3, pp. 204–213, 2002.
19. Lawrence Berkeley National Laboratory and Rumsey Engineers, Inc., "Data center energy benchmarking case study," 2003, available at: http://datacenters.lbl.gov/, accessed: March 24, 2006.
20. G. Li, S. Azarm, A. Farhang-Mehr, and A. Diaz, "Approximation of multi-response engineering simulations: a dependent meta-modeling approach," Structural and Multidisciplinary Optimization, vol. 31, pp. 260–269, 2006.
21. M. Li, G. Li, and S. Azarm, "A kriging meta-model assisted multi-objective genetic algorithm for design optimization," in Proc. of the ASME International Design Engineering Technical Conferences, Paper No. DETC2006-99316, Philadelphia, PA, Sept . 10–13. 2006.
22. T.J. Mitchell and M.D. Morris, "The spatial correlation function approach to response surface estimation," in Proc. of the 24th Winter Simulation Conference, Arlington, VA, USA, 1992, pp. 565–571.
23. S. Narayanan and S. Azarm, "On improving multiobjective genetic algorithms for design optimization," Structural and Multidisciplinary Optimization, vol. 18, pp. 146–155, 1999.
24. H.M. Park and O.Y. Kim, "Reduction of modes for the control of viscous flows," International Journal of Engineering Science, vol. 39, pp. 177–200, 2001.
25. H.M. Park and W.J. Li, "Boundary optimal control of natural convection by means of mode reduction," Journal of Dynamic Systems, Measurement and Control, vol. 124, pp. 47–54, 2002.
26. P.Y. Papalambros and D.J. Wilde, Principles of Optimal Design: Modeling and Computation, 2nd ed, Cambridge University Press, New York, USA, 2000.
27. C.D. Patel, C.E. Bash, C. Belady, L. Stahl, and D. Sullivan, "Computational fluid dynamics modeling of high compute density data centers to assure system inlet air specifications," in Proc. of IPACK'01—The Pacific Rim/ASME International Electronics Packaging Technical Conference and Exhibition Kauai, HI, 2001.
28. C.D. Patel, R. Sharma, C.E. Bash, and A. Beitelmal, "Thermal considerations in cooling of large scale high compute density data centers," in Proc. of ITHERM 2002—Eight Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems, San Diego, CA, 2002.
29. J. Rambo and Y. Joshi, "Thermal Performance Metrics for Arranging Forced Air Cooled Servers in a Data Processing Cabinet," ASME Journal of Electronic Packaging, vol. 127, pp. 452–459, 2005.
30. J. Rambo, "Reduced-order modeling of multiscale turbulent convection: application to data center thermal management," Ph.D. Dissertation, Department of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA, 2006.
31. K. Rasheed, "Informed operators: Speeding up genetic-algorithm-based design optimization using reduced models," in Proc. of the Genetic and Evolutionary Computation Conference, Las Vegas, NV, USA, 2000, pp. 628–635.
32. K. Rasheed, X. Ni, and S. Vattam, "Comparison of methods for developing dynamic reduced models for design optimization," Soft Computing, vol. 9, pp. 29–37, 2005.
33. S.S. Ravindran, "Control of flow separation over a forward-facing step by model reduction," Computer Methods in Applied Mechanics and Engineering, vol. 191, pp. 1924–1942, 2002.
34. N. Rolander, J. Rambo, Y. Joshi, and F. Mistree, "Towards Sustainable Design of Data Centers: Addressing the Lifecycle Mismatch Problem," presented at IPACK'05—International Electronic Packaging Technical Conference and Exhibition, San Francisco, CA, 2005.
35. W.J. Roux, N. Stander, and R.T. Haftka, "Response surface approximations for structural optimization," International Journal for Numerical Methods in Engineering, vol. 42, pp. 517–534, 1998.
36. S. Ruzika and M.M. Wiecek, "A survey of approximation methods in multiobjective programming," Journal of Optimization Theory and Applications, vol. 126, no. 3, pp. 473–501, 2003.
37. J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn, "Design and analysis of computer experiments," Statistical Science, vol. 4, no. 4, pp. 409–435, 1989.
38. S. Shan and G.G. Wang, "An efficient Pareto set identification approach for multiobjective optimization on black-box functions," Transaction on ASME, Journal of Mechanical Design, vol. 127, pp. 866–874, 2005.

39. T.W. Simpson, A.J. Booker, D. Ghosh, A.A. Giunta, P.N. Koch, and R.-J. Yang, "Approximation methods in multidisciplinary analysis and optimization: a panel discussion," Structural and Multidisciplinary Optimization, vol. 27, no. 5, pp. 302–313, 2004.
40. US Army Corps of Engineers Internet Publishing Group: "Engineering and design—Practical aspects of applying geostatistics at hazardous, toxic, and radioactive waste sites," Technical Letter, ETL 1110-1-175, 1997.
41. E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the Strength Pareto approach," IEEE Transactions on Evolutionary Computation, vol. 3, no. 4, pp. 257–271, 1999.