Steel and Composite Structures, *Vol. 24, No. 2 (2017) 227-237*
DOI: https://doi.org/10.12989/scs.2017.24.2.227

Optimization of long span portal frames using spatially distributed surrogates

Zhifang Zhang [1a], Jingwen Pan [1b], Jiyang Fu [1c], Hemant Kumar Singh [2d],
Yong-Lin Pi [3e], Jiurong Wu [1f] and Rui Rao [*1]

[1] *Guangzhou University-Tamkang University Joint Research Center for Engineering Structure Disaster Prevention and Control, Guangzhou University, Guangzhou 510006, P.R. China*
[2] *School of Engineering and Information Technology, University of New South Wales, Canberra, Australia*
[3] *Centre for Infrastructure Engineering Safety, University of New South Wales, Sydney, Australia*

(*Received November 21, 2016, Revised March 10, 2017, Accepted March 24, 2017*)

Abstract. This paper presents optimization of a long-span portal steel frame under dynamic wind loads using a surrogate-assisted evolutionary algorithm. Long-span portal steel frames are often used in low-rise industrial and commercial buildings. The structure needs be able to resist the wind loads, and at the same time it should be as light as possible in order to be cost-effective. In this work, numerical model of a portal steel frame is constructed using structural analysis program (SAP2000), with the web-heights at five locations of I-sections of the columns and rafters as the decision variables. In order to evaluate the performance of a given design under dynamic wind loading, the equivalent static wind load (ESWL) is obtained from a database of wind pressures measured in wind tunnel tests. A modified formulation of the problem compared to the one available in the literature is also presented, considering additional design constraints for practicality. Evolutionary algorithms (EA) are often used to solve such non-linear, black-box problems, but when each design evaluation is computationally expensive (e.g., in this case a SAP2000 simulation), the time taken for optimization using EAs becomes untenable. To overcome this challenge, we employ a surrogate-assisted evolutionary algorithm (SAEA) to expedite the convergence towards the optimum design. The presented SAEA uses multiple spatially distributed surrogate models to approximate the simulations more accurately in lieu of commonly used single global surrogate models. Through rigorous numerical experiments, improvements in results and time savings obtained using SAEA over EA are demonstrated.

Keywords: structural optimization; steel portal frames; equivalent static wind loading; SAP2000; surrogate-assisted evolutionary algorithm

1. Introduction

Long-span steel frames are often used in low-rise structures of commercial and industrial buildings (Kravanja and Žula 2010, Phan *et al*. 2013b). Due to the long span of such frames and tapered sections of the columns and beams, such portal frames are quite sensitive to wind loading. The wind loading and wind-induced response are the major factors to be considered during the structural design of portal frames (Hayalioglu and Degertekin 2005, Saka 2003).

To design such frames, it is important to optimize their structures and cost under wind loading, and therefore the modeling of wind loads should be as accurate as possible. Traditionally, wind loads have been considered in design of such structures by using simplified codes and standards which include tables and plots for reduction factors. These standards are mostly based on experience (Camp *et al*. 1998).

Furthermore, although the wind environment is dynamic in real-life scenarios, the wind effects are generally considered as static loads in the structural optimization for simplicity. Compared to the static loading case, the wind resistant optimization of a portal frame under dynamic wind loading is significantly complicated and time-consuming (Kameshki and Saka 2003, Paya *et al*. 2008). As the time variable is involved in calculating the dynamic response, the objective function(s) and the constraint(s) in the optimization are time-dependent functions. Theoretically, the structural optimization process should be conducted at each time instant and the peak values of the responses should be adopted for wind resistant optimization. Besides, for any modification of frame dimensions, not only are the internal forces of the frame redistributed, but the external interaction between the wind and the portal frame will also change accordingly, resulting in different deformations of the frame (Li and Li 2004). Therefore, the displacements of portal frame at the critical locations (i.e., atop of the column and the mid-span of the rafter) under wind effects have to be calculated by re-analyzing the whole frame. At the same time, the constraints on strength, stability and flexibility have to be continually satisfied as well, which makes the

∗Corresponding author, Associate Professor,
E-mail: raorui@gzhu.edu.cn
[a] E-mail: zfzhang@gzhu.edu.cn
[b] E-mail: jingwenpan@hotmail.com
[c] E-mail: jyfu@gzhu.edu.cn
[d] E-mail: h.singh@adfa.edu.au
[e] E-mail: y.pi@unsw.edu.au
[f] E-mail: jrwuce@qq.com

http://www.techno-press.org/?journal=scs&subpage=6
ISSN: 1229-9367 (Print), 1598-6233 (Online)

optimization of portal frame structures a complex task. Though a number of efforts have been made towards achieving this (Kameshki and Saka 2001, Kravanja *et al*. 2013, Moss *et al*. 2009), there are no established methods for effectively optimizing such structures under dynamic wind loading. To address this, Wu *et al*. (2012) adopted optimality criteria (OC) method to obtain the minimum heights of cross sections for rafters and columns (therefore minimum total weight) of long-span portal frames while satisfying the stability constraints under dynamic wind environment. The weight of the portal frame was reported to have reduced from that of an initial baseline design by about 26%-28%.

This paper presents a method for structural optimization for a long-span portal rigid steel frame with tapered sections under dynamic wind loading. Application of optimization algorithms in real-life engineering structure design has been a prominent research topic (Begg and Liu 1998, Park and Adeli 1997), and the two main type of methods that are prevalent for optimization are classical and evolutionary. The classical methods (e.g. the OC method used in Wu *et al*. (2012)) are quick, but are often limited in terms of the types of functions they can handle. They often require mathematical conditions on the problem to be optimized, such as linearity, continuity, convexity, differentiability etc. Moreover, they are typically suited to find a local optimum of the problem instead of global. Unfortunately, most of the practical problems do not adhere to the above strict conditions, and are highly nonlinear and often even *black-box* (Adeli and Park 1996), i.e., the underlying function may not be explicitly formulated, or may be restricted for the user due to other reasons (e.g., trade secrets in commercial software). To solve such problems, evolutionary algorithms (EA) (also referred to in literature as genetic algorithms or GA) have been a popular choice in recent years due to their ability to deal with non-linear, black-box functions. Further, they can also deal with discrete/mixed variables, multiple objectives and attempt to locate the global optimum of the problem being solved. EAs typically *evolve* a population of designs over a number of generations using principles of natural selection (Deb 2001). Over the years, the approach has been used for optimization in a number of studies, including structural optimization. Hayalioglu and Degertekin (2005) employed a GA to obtain the minimum total cost of non-linear steel frames with semi-rigid connections and column bases. Camp *et al*. (1998) developed a design procedure incorporating a simple GA for discrete optimization of two-dimensional structures. The objective function considered was total weight (or cost) of the structure, which was minimized subject to serviceability and strength requirements. A GA based design procedure was developed as a module in the finite element analysis (FEA) program. Kameshki and Saka (2001) presented a GA based design method for nonlinear multi-story steel frames to achieve least weight while satisfying the serviceability and strength constraints. Senouci and Al-Ansari (2009) presented a GA model to perform cost optimization of composite beams based on load and resistance factor design specification. Sgambi *et al*. (2012) proposed a method based on a combined application of GA and finite element method (FEM) to conduct the serviceability assessment of a long-span suspension bridge. In the context of composite steel frames, GA was employed in Artar and Daloglu (2015a, b) to optimize the weight of the structures. Cost minimization of pre-stressed steel trusses considering shape and size variables was investigated using a GA by Aydin and Cakir (2015). Phan *et al*. (2013a) used GA for cost minimization of cold-formed steel portal frames. GA was also used for optimization of pre-cast hollow core slabs by Sgambi *et al*. (2014). Use of some other nature-inspired methods has also been reported, such as colliding bodies optimization (Kaveh and Shokohi 2015) for laterally supported castellated beams and cuckoo search (Kaveh *et al*. 2014) for multi-span composite girder bridges. While most of the studies typically consider weight or cost (which are sometimes equivalent) as the objective, some studies have also consider other objectives such as frequency (Topal 2012).

Though EAs are powerful tools for optimization, they typically require large numbers of design evaluations to converge near the optimum solution(s). If evaluation of each design is done using a computationally expensive simulation (e.g., in this study a SAP2000 simulation, or FEA in some of the above mentioned studies), then the overall time for optimization becomes impractical, especially if the optimization needs to be executed multiple times during design development. This remains the key drawback of EAs. In order to alleviate this difficulty, the use of “surrogate models” has been proposed in the literature. Surrogate models (also known as meta-models or approximation models) build an approximate function representing the true simulations based on available data. Thereafter, the responses (objectives/constraints) could be predicted instead of truly evaluated in order to guide the search. For an overview of some of the prominent approaches that use surrogate modeling, the interested readers are referred to the survey papers (Jin 2005, Wang and Shan 2007).

Although the use of surrogates in engineering design is not new, to the authors’ knowledge, there are no reports of using surrogate-assisted optimization on long-span portal frames. In order to address this gap, we present a surrogate-assisted evolutionary algorithm (SAEA) in this paper for the optimization of weight (correspondingly, the cost) of a portal frame, which satisfies strength and stability criteria to resist in-plane buckling under dynamic wind loads. In contrast with most of the surrogate-assisted approaches available in literature, which tend to use single global surrogate model, the presented algorithm uses multiple spatially distributed surrogates. This means that various areas of the search space are approximated locally using different types of surrogate models, which are also updated over time as more data becomes available. Such an approach has been demonstrated to offer more flexibility and accuracy on a number of benchmark optimization problems by Isaacs *et al*. (2009), Bhattacharjee *et al*. (2016). In this paper, we present the advantages of using SAEA over traditional EA in terms of the outputs as well as time taken for the portal frame optimization problem. Furthermore, we also present and solve a modified formulation of the problem to incorporate certain practical constraints. To deal with dynamic wind loads, equivalent static wind loads (ESLWs) (Sun *et al*. 2015) are used, incorporating an aerodynamic database of wind tunnel tests to generate equivalent static wind effects.

Remainder of this paper is organized as follows. Mathematical formulation of the optimization problem is discussed in Section 2, followed by the proposed approach for solving it in Section 3. Numerical experiments are presented in Section 4. Summary and potential future research directions are presented in Section 5.

2. Mathematical formulation

2.1 Formulation of the optimization problem

To formulate an optimization problem, the design variables (attributes that represent a design), objectives (functions that need to be minimized/maximized) and constraints (relations among variables that must be satisfied) need to be defined. The columns and beams of typical steel portal frames usually have I-sections and the flanges of the I-section typically have uniform width *wF i* and thickness *tF i*. The web of the I-section has a uniform thickness *tw i* and linearly varying height as shown in Fig. 1. The 5 web heights $\{h_i;\ i = 1, 2, \ldots, 5\}$ of sections 1-5 shown in Fig. 1 are considered as the design variables. h_1-h_5 are discrete variables since the web heights (in mm) should be integers in a realistic scenarios.

The beams and columns are divided to elements, and the objective function is defined as minimization of the total weight (W) of all the elements in the steel frame, calculated as show in Eq. (1).

$$W = \rho \sum_{i=1}^{N} L_i \left(2 w_i^F t_i^F + \frac{t_i^W}{2} \sum_{j=1}^{2} D_{ij}^W \right) \tag{1}$$

In the above expression, ρ is the density of the steel, L_i is the length of the i^{th} element ($i = 1, 2, \ldots, N$). For example, in the structure shown in Fig. 1(a), $N = 12$ elements are marked along the frame, and *DW i*1 and *DW i*2 are the web heights at the two ends of the tapered i^{th} element.

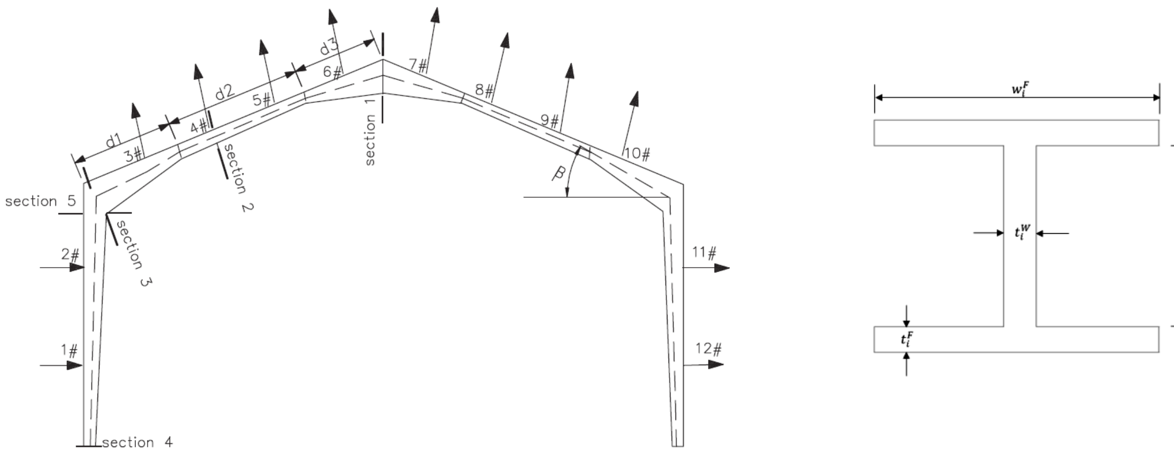
The constraints are defined by limiting two key wind-induced displacements, the vertical displacement (δ_1) at the mid-span of rafter and the horizontal displacement (δ_2) at the top of the column, to comply with the technical specification for steel structure of light-weight building with gabled frame (China Association for Engineering Construction Standardization 2012). According to the specification, the wind-induced displacements δ_1 and δ_2 should be less than $\frac{B}{180}$ (B is the span of the rafter) and $\frac{H}{60}$ (H is the height of the column), respectively. The two resulting constraints are shown in Eq. (2).

$$g_1 \equiv \delta_1 - B/180 \le 0\ , \quad g_2 \equiv \delta_2 - H/60 \le 0 \tag{2}$$

The two wind-induced displacement constraints can be obtained by the virtual work principle (Chan *et al*. 1995) as shown in Eq. (3)

$$\begin{aligned} \delta_k = \sum_{i=1}^{N} \int_0^{L_i} \Bigg(& \frac{F_{Xk} f_X}{EA} + \frac{F_{Yk} f_Y}{GA_Y} + \frac{F_{Zk} f_Z}{GA_Z} \cdots \\ & \cdots + \frac{M_{Xk} m_X}{GI_X} + \frac{M_{Yk} m_Y}{EI_X} + \frac{M_{Zk} m_Z}{EI_Z} \Bigg) dx; \\ & (k = 1, 2) \end{aligned} \tag{3}$$

where N is the number of total structural elements in the structure, E and G are the elastic and shear modulus of steel respectively, A, A_Y and A_Z are the areas of cross section and the shear areas about two major axes (X and Y) of cross section, respectively. GI_X, EI_Y, and EI_Z are torsional stiffness and flexural stiffness of the design section, F_{Xk}, F_{Yk}, F_{Zk}, M_{Xk}, M_{Yk} and M_{Zk} are the internal forces resulting from the applied unit force in the direction at the specified displacement, while f_X, f_Y, f_Z, m_X, m_Y and m_Z are the internal force (axial force, shear force and bending moment in along the three major axes) induced from external wind loading.



(a) (b)

Fig. 1 Portal frame and the I shape cross-section: (a) distribution of 5 sections and 12 elements (figure taken from Wu *et al*. (2012)); (b) dimensions of I type cross-section

All these internal forces of the elements could be obtained from numerical simulations using Structural Analysis Program (SAP2000) (Computers and Structures 1997). Since the geometric properties could be expressed as a function of the design variables, the integration and derivation of the constraint conditions in Eq. (3) with respect to the design variables could be carried out using numerical methods, such as those available in Matlab 2014a (MathWorks 2014). The dynamic wind loads are incorporated using the equivalent static wind loads based on the aerodynamic database of wind tunnel tests, as described in the next subsection.

2.2 Calculation of equivalent static wind loads

For a long-span portal steel frame, the wind-induced displacements under dynamic wind loading are mainly caused by the background response. In the cases where the resonant response is negligible, the Load Response Correlation (LRC) method (Holmes 2002) can be used to obtain the equivalent static wind loads (ESWLs) from the mean and background components of the wind load as shown in Eq. (4).

$$\left[p(z)\right]_{\hat{r}} = \bar{p}(z) + p_B(z) = \bar{p}(z) + g_B \rho_{pr}(z)\sigma_P(z) \tag{4}$$

where $\bar{p}(z)$ and $p_B(z)$ are the mean and background wind loads respectively; g_B is the peak factor, $\sigma_P(z)$ is the standard deviation of the fluctuating wind loads, $\rho_{pr}(z)$ is the correlation coefficient between the fluctuating wind load $p'(z)$ a and the specified wind-induced response (r), which could be expressed as shown in Eq. (5).

$$\rho_{pr}(z) = \frac{\sigma_{pr}}{\sigma_p(z)\sigma_{r,B}} = \frac{\overline{p'(z,t)\int_0^L p'(z_1,t)I_r(z_1)dz_1}}{\sigma_p(z)\sigma_{r,B}}$$

where

$$\sigma_{r,B} = \left[\int_0^L\int_0^L \overline{p'(z_1,t)\,p'(z_2,t)}I_r(z_1)I_r(z_2)dz_1dz_2\right]^{1/2} \tag{5}$$

Here the subscript B denotes the background response, L denotes the length of structural element, $I_r(z)$ is the influence coefficient, i.e., the value of wind-induced response when a unit load is applied at the position z. If the portal steel frame is divided into discrete elements, the expressions in Eq. (5) can be rewritten as shown in Eq. (6).

$$\rho_{pj,r} = \frac{\sum_{i=1}^{N}\overline{p_i'(t)\,p_j'(t)}\beta_i}{\sigma_{pj}\sigma_{r,B}}$$

$$\sigma_{r,B} = \sum_{i=1}^{N}\sum_{j=1}^{N}\overline{p_i'(t)\,p_j'(t)}\beta_i\beta_j \tag{6}$$

Subsequently, the ESWLs defined in Eq. (4) can also be written in the finite summation form as shown in Eq. (7)

$$P_{B,j} = g_B\rho_{pj}\sigma_{pj} = \frac{g_B\sum_{i=1}^{N}\overline{p_i'(t)\,p_j'(t)}\beta_i}{\sigma_{r,B}} \tag{7}$$

$$= \frac{g_B\sum_{i=1}^{N}\overline{p_i'(t)\,p_j'(t)}\beta_i}{\sum_{i=1}^{N}\sum_{k=1}^{N}\overline{p_i'(t)\,p_k'(t)}\beta_i\beta_k}$$

where i and j are element numbers, N is total number of elements and β_i is the influence coefficient. Substitution of the aerodynamic databases from wind tunnel tests into Eq. (7) is used to obtain the equivalent static wind loads, which are then applied in the SAP2000 model to compute the internal forces, moments, and the wind-induced displacements.

3. Surrogate-assisted evolutionary algorithm

Evolutionary algorithms have been a popular choice for solving design optimization problems. This is attributed to the fact that they do not require specific mathematical properties in the objective functions/constraints and can deal with discrete and black-box functions, such as those involved in the problem studied here. EAs work by *evolving* a *population* of designs towards the optimum through iterative process of selection, recombination, mutation and reduction. The *fitter* designs at each generation (those with better objective values) are more likely to survive in the ranking process and generate new *child* solutions for the next generation. The process is similar to natural selection in nature, and is expected to produce better (and eventually optimum) designs as the generations progress.

However, EAs, though powerful, typically require large number of function evaluations to converge to the optimum, as a population of solutions are evolved over a number of generations. For the cases where a true evaluation requires a computationally expensive simulation (e.g., Computational Fluid Dynamics (CFD), Finite Element Modeling (FEM) etc.), the overall cost of optimization can thus become prohibitive. For the problem considered in this paper, the calculation of wind-induced displacements (δ_1, δ_2) requires an expensive simulation using SAP. In such cases, one of the approaches in the literature is to use a *surrogate* function in lieu of the true evaluation wherever possible to guide the search. Such functions are referred to by various terms, such as approximation models, metamodels, surrogate models or simply surrogates. Optimization methods that use these models to reduce the overall optimization time are referred to as surrogate-assisted optimization (SAO) methods.

One of the key considerations in surrogate-assisted optimization is the choice of the model itself. It is reasonably well established in the literature that there isn't a single type of metamodel that can approximate all types of data (Bhattacharjee *et al*. 2016). Furthermore, a surrogate model that is the best for an objective may not be the best for a constraint of the given problem, or a model that is best for one of the constraints may not be the best for another constraint, and so on. In fact, the same surrogate model may not be the best for a given function in all locations. Lastly, if/when more truly evaluated solutions become available during the search, the best model(s) to approximate a particular function may itself change over generations.

Considering the above factors, we apply a surrogate-assisted evolutionary algorithm (SAEA) based on multiple spatially distributed surrogates (Isaacs 2009, Isaacs *et al.* 2009) to solve this problem. The reasons for choosing this approach are straightforward: (a) the approach should be able to handle non-linear and black-box functions (given the nature of the problem); and (b) it should use small number of true function evaluations to obtain a good solution (ideally the optimum solution), given the computational complexity of calculating the wind-induced displacements.

3.1 SAEA framework

The main SAEA framework used in this study has been developed by Isaacs (2009), and summarized in Algorithm 1. The underlying baseline algorithm in the framework is the real-valued evolutionary algorithm presented in Deb *et al.* (2002), which is enhanced through surrogates resulting in SAEA. The key components of the approach are outlined next.

3.1.1 Initialization

The algorithm begins by initializing a population of solutions randomly within the given variable bounds. These solutions are then evaluated (true evaluations through SAP2000), and added to the archive $\mathcal{A}$ of truly evaluated solutions.

Algorithm 1 Surrogate-assisted Evolutionary Algorithm (SAEA)

Require: N : Population size
N_G : Number of Generations
$FE_{\max}$: Maximum number of true function evaluations
$I_{train} > 1$: Periodic Surrogate Training Interval
$N_{RC} > 1$: Number of truly evaluated solutions retained

Set $i = 1$ {Generation counter}
Set $FE = 0$ {True function evaluation counter}
Initialize(pop_i)
Evaluate(pop_i) {True evaluations of initial population}
Rank(pop_i)
Set $\mathcal{A} = pop_i$ {Archive to store all truly evaluated solutions}
for $i = 2$ **to** N_G **do**
  if $i > I_{train}$ and modulo(i, I_{train}) = = 0 **then**
    do_training = 1
  end if
  $cpop_i$ = Evolve($pop_i - 1$)
  if *do_training* = 1 **then**
    Evaluate($cpop_i$)
  else
    EvaluateSurrogate($cpop_i$,S)
    Rank($cpop_i$)
    for $j = 1$ to N_{RC} **do**
      if $Soln_{rank=j}$ ($cpop_i$) is better than $Soln_{rank=j}(pop_{i-1})$ **then**
        Evaluate($Soln_{rank}=j$ ($cpop_i$))
      end if
    end for
  end if
  $\mathcal{A} = \mathcal{A} \cup cpop_i$
  Rank($pop_{i-1} \cup cpop_i$)
  pop_i = Reduce($pop_{i-1} \cup cpop_i$)
if *do_training* = = 1 **then**
    S = BuildSurrogate($\mathcal{A}$)
end if
  Update FE
  if $FE \geq FE_{max}$ **then**
    return;
  end if
end for

3.1.2 Ranking

For a given set of solutions, the ranking is done using the *feasibility first* principles. The solutions are first separated into those that do not violate any constraints ("feasible") and those that do violate the constraints ("infeasible"). The feasible solutions are ranked based on their objective value. Thus, for a minimization problem (like the one considered here), a solution with lower objective value gets a better rank. The infeasible solutions are ranked among themselves based on the amount of maximum constraint violation (CV). Thus a solution that has smaller value of CV compared to another will have a better rank. Thereafter, the feasible solutions are placed above the infeasible solutions in the ranked list.

3.1.3 Parent selection and evolution

In order to generate the child population from a parent population, fitter parents need to be selected and then evolution operators need to be applied on them. In SAEA, fitter parents are selected using tournament selection. This simply involves comparing a ranked population to a shuffled population. A pairwise comparison is done between each pair and the better ranked solution is chosen as a parent. For a population size of N this will result in N parents. From these, $N/2$ pairs are created for crossover. Thereafter, Simulated Binary Crossover (Deb *et al.* 2002) for crossover is performed to generate child solutions. Each pair of parent solutions generates a pair of child solutions in the process, and thus a total of N child solutions are generated. Thereafter, Polynomial Mutation (Deb *et al.* 2002) is further applied to mutate some of the child solutions. The crossover and mutation are controlled using the parameters *probability of crossover* (usually set close to 1) and *probability of mutation* (usually set close to 0), respectively.

3.1.4 Surrogate training

This is the key step of the SAEA. After every I_{train} generations, a set of different surrogate models are trained based on the available archive $\mathcal{A}$ of truly evaluated solutions. As mentioned above, it is known that a single surrogate may not be able to accurately approximate the whole objective space. Therefore a k-means clustering of $\mathcal{A}$ is performed in order to identify different spatial sets of solutions within $\mathcal{A}$. The solutions in each cluster are divided into training (80%) and validation (20%) sets. For each cluster, multiple surrogates are built using the training set, and the one that has the least prediction error on the validation set is used for the final approximation of that cluster. In this study, we use three different type of surrogate models (RSM, RBF and Kriging), which are briefly discussed below. It is possible within the framework to use other type of surrogate models as well.

(1) *Response Surface Method* (*RSM*): Response Surface Method is a linear or polynomial regression. RSM uses first or second degree polynomials to fit the data (Myers and Montgomery 1995). A generic second order quadratic polynomial model, with m input variables $\{x_1, x_2,..., x_m\}$ can be written as shown in Eq. (8)

$$y(\mathbf{x}) = \beta_0 + \sum_{i=1}^{m} \beta_i x_i + \sum_{i=1}^{m} \beta_{ii} x_i^2 + \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \beta_{ij} x_i x_j \tag{8}$$

where, $\beta_0, \beta_i, \beta_{ii}, \beta_{ij}$ are the unknown parameters of the model that are determined from the given data. In vector form, this can be written as $y(\mathbf{x}) = \mathbf{f}^T \mathbf{b}$. The vector $\mathbf{f}$ contains all the terms of $x_1, x_2,..., x_m$ and vector $\mathbf{b}$ contains all the unknown coefficients. The values of the unknown coefficients are determined using least squares method. The least squares estimate of $\mathbf{b}$ is given by Eq. (9).

$$\hat{\mathbf{b}} = \left(\mathbf{F}^{\mathrm{T}}\mathbf{F}\right)^{-1} \mathbf{F}^{\mathrm{T}}\mathbf{Y} \tag{9}$$

where $\mathbf{F}$ is a matrix containing N rows, each row is a vector $\mathbf{f}^{\mathrm{T}}$ evaluated at an observation and $\mathbf{Y}$ are the observed responses.

(2) *Radial Basis Function* (*RBF*): Radial Basis Function is a type of Artificial Neural Network (ANN), which are known to be effective in modeling nonlinear relationships. A model for an RBF response is given by Eq. (10).

$$y(\mathbf{x}) = \sum_{i=1}^{k} w_i \phi(\mathbf{x} - \mathbf{x_i}) \tag{10}$$

where $\phi(.)$ denote the radial basis functions, ‖.‖ is the Euclidean norm and w_i are the unknown weights. An RBF is symmetric around its associated center, in this case $\mathbf{x_i}$. A common RBF is the Gaussian Function with Euclidean norm shown in Eq. (11).

$$\phi(\mathbf{x} - \mathbf{x_i}) = e^{-r^2/\sigma^2} \tag{11}$$

where r is the Euclidean distance between $\mathbf{x}$ and $\mathbf{x_i}$, and σ is the scale or width parameters. In the generalized RBF network, the number of centers (k) are usually less than the number of observations N. The unknown weights w_i are determined using least squares estimates.

(3) *Kriging*: Kriging, also known as Design and Analysis of Computer Experiments (DACE) (Sacks *et al*. 1989, Queipo *et al*. 2002) is among the most popular methods to approximate nonlinear functions. It attempts to approximate the function as a combination of a global regression model and a deviation term (with zero mean) as shown in Eq. (12).

$$y(\mathbf{x}) = \mu(\mathbf{x}) + \epsilon(\mathbf{x}) \tag{12}$$

The regression model is typically polynomial, for example the one shown in Eq. (8). The covariance function is given using the Eq. (13).

$$cov\left(\epsilon(\mathbf{x_i}), \epsilon(\mathbf{x_j})\right) = \sigma^2 R(\mathbf{x}_i, \mathbf{x}_j) \tag{13}$$

Here, σ^2 is the process variance and $R(\mathbf{x_i}, \mathbf{x_j})$ is a spatial correlation function, typically modeled as shown in Eq. (14).

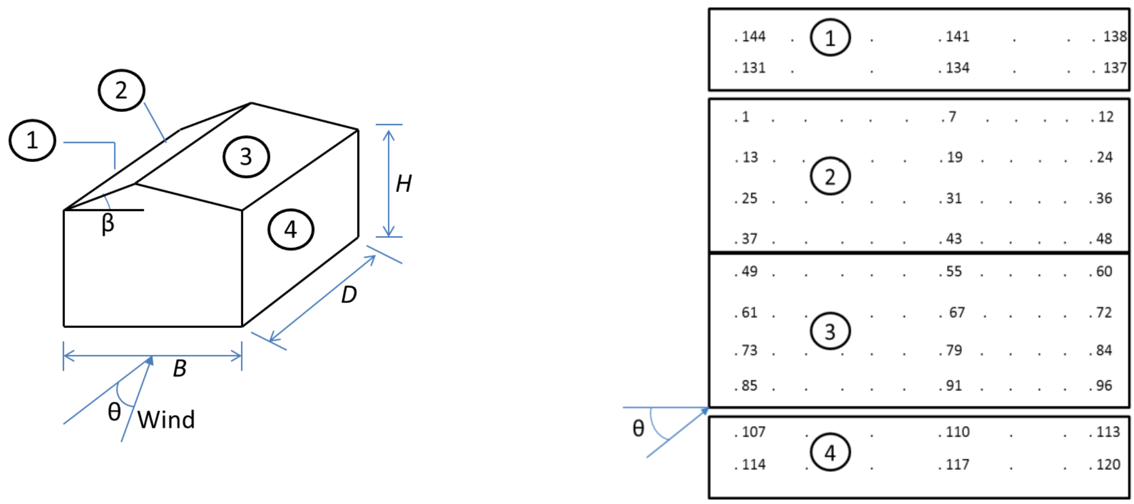
$$R(\mathbf{x_i}, \mathbf{x_j}) = \exp\left(-\sum_{k=1}^{m} \theta_k \left|x_k^i - x_k^j\right|^p\right) \tag{14}$$

Here, θ_k and p are hyper-parameters that are determined using the maximum likelihood estimation in order to obtain the Kriging model. For more detailed description of the model, the readers are referred to Sacks *et al*. (1989).

Once the surrogate model S (which is in fact a set of spatially distributed surrogates) has been updated, the algorithm evaluates the objectives and constraints using S (thereby bypassing the time consuming true function evaluations) until the next update.

4. Numerical experiments: Case study

Case study on a long-span low-rise building (shown in Fig. 2(a)) is presented, which has dimensions of 24 m (depth D) × 16 m (span B) × 4 m (height H) with an incline angle β of the rafter equal to 9.4°. There are total 5 portal frames for constructing the low-rise building, with the frame-to-frame distance along the building depth being 6 m. The middle long-span portal frame (located in the middle of the building depth) is selected for the case study presented here. The length of the rafter varies along its span with a ratio of $d_1 : d_2 : d_3 = 0.2 : 0.6 : 0.2$ (refer to Fig. 1 for d_1–d_3). For I shaped cross-section rafters, the width and thickness of flange are 0.25 m and 0.01 m, respectively, and the thickness of its web is 0.006 m. For the I shaped columns, the corresponding dimensions are 0.28 m, 0.01 m and 0.008 m respectively. The dynamic wind loading for this case was extracted from the aerodynamic database of wind tunnel tests for a low-rise building of the same dimensions. The database was compiled by the Tokyo Polytechnic University, Japan and is accessible online (Tokyo Polytechnic University 2011). In the database, the mean wind speed is 28 m/s at a height of 10 m and the measurement of wind pressure was performed at multiple points as shown in Fig. 2(b). The time history of pressure taken points (points no. 141, 134, 7,…, 117 along the middle part of the building shown in Fig. 2(b)) under the wind angle of 45° were extracted from the aerodynamic database. By multiplying the load covering areas, the time history of wind loading acting on the element nodes (No. 1# -12# shown in Fig. 1(a)) can be obtained. With the corresponding mean, background, and standard deviation components of wind loads at these nodes, the equivalent



(a) (b)

Fig. 2 (a) The building configuration, (b) distribution of pressure taken points in the wind tunnel test. Figures taken from Wu *et al*. (2012)

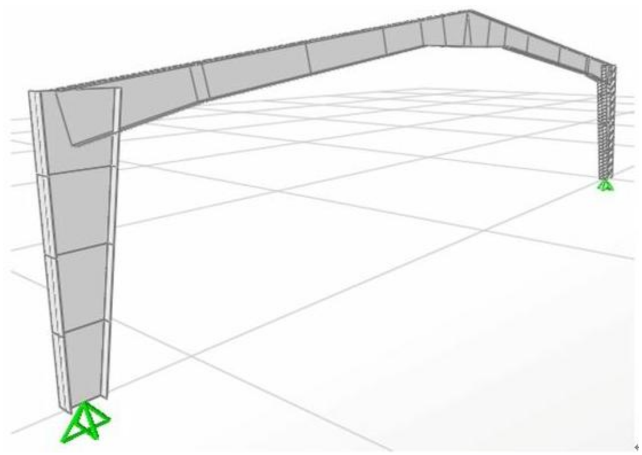


Fig. 3 SAP2000 model of long-span portal frame

static wind loads can be obtained using Eqs. (4)-(7).

ESWLs were then applied on numerical model built in SAP2000, a commercial software developed by American Computers and Structures Inc. for structural design and analysis (http://www.csiamerica.com/products/sap2000), to generate the internal forces and moments in portal frame (Fig. 3). The internal force and moment values are input into Eq. (3) to obtain the two wind-induced displacements (δ_1, δ_2) by virtual work principle.

4.1 Numerical experiment setup

For all numerical experiments in the section, we maintain a consistent framework and settings for EA and SAEA so that the effect of using surrogate models on the search behavior could be clearly observed. Both algorithms use a population size of N = 12. In the context of solving computationally expensive problems, the total number of true evaluations (SAP2000 simulations) are restricted to a small number $FE_{\max}$ = 240. The crossover and mutation probabilities (p_c and p_m) are set to 0.9 and 0.1 respectively, while the index of crossover and mutation (η_c and η_m) are set to 10 and 20 respectively. For the SAEA, the frequency of updating surrogates (I_{train}) is set to 5 generations, and retain count N_{RC} is set to 1. Since the EAs are stochastic in nature, it is important to observe the performance across multiple instantiations in order to establish the reliability of the algorithm. Therefore, multiple (11) independent runs are done for each algorithm and the resulting statistics are reported.

In the experiments that follow, we present two studies. First is based on the formulation presented in Wu *et al*. (2012), whereas the second one is on a proposed modified formulation that includes additional design constraints for practicality.

4.2 Formulation 1

In the first experiment, the formulation of the problem given in Wu *et al*. (2012) is studied. In this formulation (referred to as formulation 1), the lower and upper limits of the variables are considered as shown in Table 1. In Wu *et al*. (2012), a classical technique, known as the optimality criteria method, was applied to solve the problem. Starting from a base design of (0.88, 0.53, 0.83, 0.38, 0.88) with a weight of 1.766 t, the optimization resulted in a final design weighing 1.2715 t.

The results obtained using EA and SAEA are shown in Tables 2-3. It can be seen that SAEA is able to obtain a feasible design with each of the variables at their lower bound (300,200,300,100,300). Since the weight monotonically increases with an increase in each of these

Table 1 Lower and upper bounds of design variables for formulation 1

Web height(mm)	h_1	h_2	h_3	h_4	h_5
Lower bound	300	200	300	100	300
Upper bound	1500	1100	1400	800	1500

Table 2 Results (weight in tonnes) obtained across multiple (11) runs for formulation 1

Algorithm	Mean	Median	Best	Worst	St. dev.
EA	1.3336	1.3092	1.2656	1.5391	0.0813
SAEA	1.2603	1.2593	1.2593	1.2648	0.0018

Table 3 Designs obtained by EA and SAEA for formulation 1. All web-heights are in mm

	h_1	h_2	h_3	h_4	h_5
		EA			
Median	533	225	350	127	320
Best	325	203	329	101	300
Worst	427	414	306	564	388
		SAEA			
Median	300	200	300	100	300
Best	300	200	300	100	300
Worst	309	201	344	101	302

variables (heights of the sections), it is evident that the lowest bound of each variable being feasible is the true optimum of the problem. The median weight obtained using SAEA is 1.2593 t, which corresponds to the true optimum. At the same time, it can be observed that EA was not able to converge to the true values within the given budget of 240 evaluations in any of the runs, since the best value obtained by it is 1.2656 t. This clearly shows the efficacy of SAEA in delivering superior results on a limited computational budget. It is also worth noting that in the optimized solution reported in Wu *et al*. (2012), the variables are not at their lower bound, and thus the OC method appears to have converged to a local optimum instead of global. Thus, given the nature of the problem, it is beneficial to use evolutionary approach to solve it.

The performance is further evidenced through a comparison of convergence rates (all, mean and median runs) using EA and SAEA in Fig. 4. It can be clearly seen from the figures that SAEA is able to converge much faster than EA, and is able to achieve the true optimum in most of the runs (which is also reflected in its low standard deviation values). From Fig. 4(b), it can also be seen that to

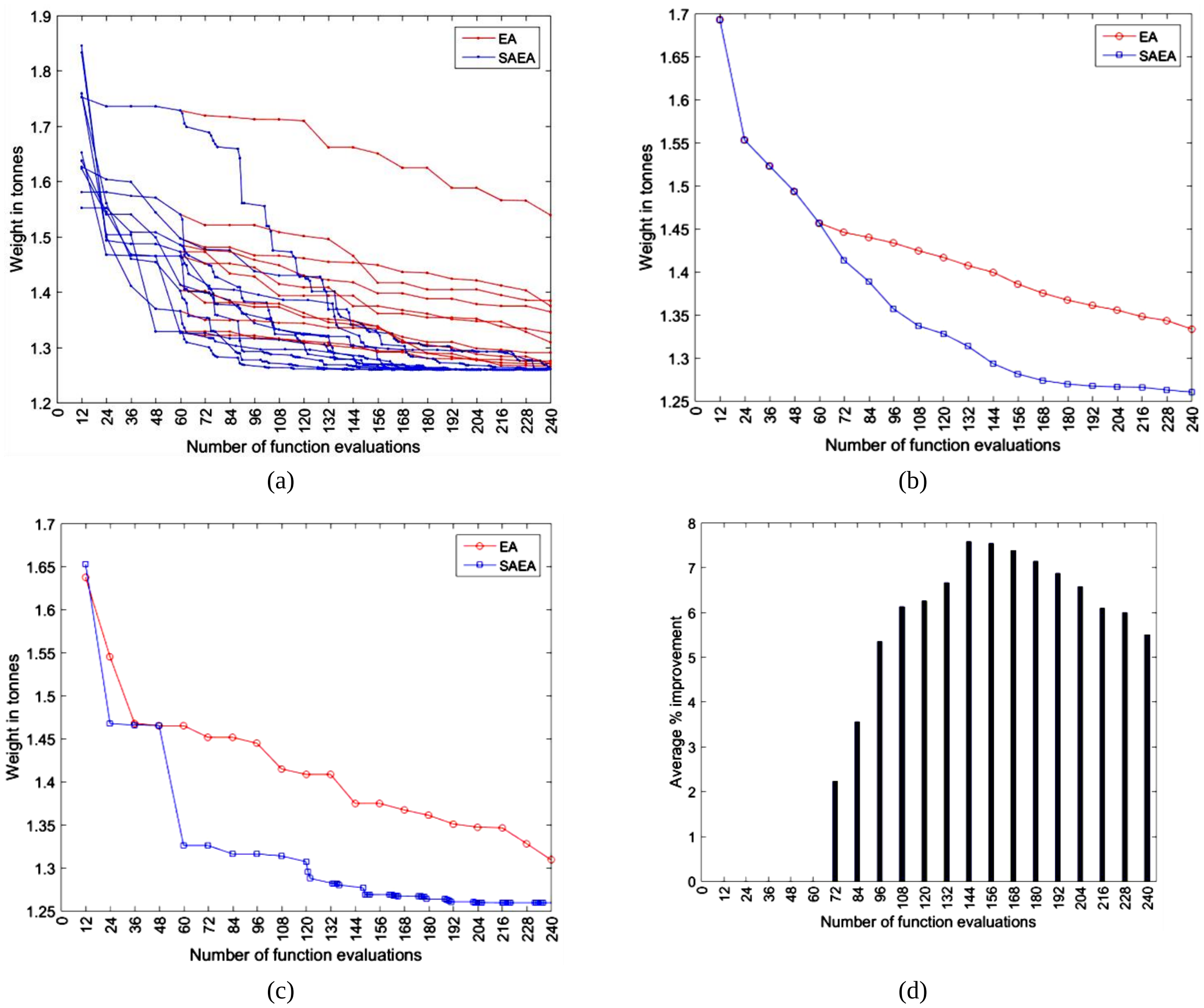


Fig. 4 Convergence for original formulation over 11 runs: (a) all; (b) average; (c) median; (d) average improvement in weight value obtained by SAEA over EA

Table 4 Lower and upper bounds of design variables for formulation 2

Web height(mm)	h_1	h_2	h_3	h_4	h_5
Lower bound	100	100	100	100	100
Upper bound	1500	1100	1400	800	1500

Table 5 Results (weight in tonnes) obtained across multiple (11) runs for the modified formulation

Algorithm	Mean	Median	Best	Worst	St.dev.
EA	1.2324	1.1816	1.1363	1.4458	0.0975
SAEA	1.1484	1.1397	1.1227	1.1911	0.0265

obtain the same average result as EA (1.336 t), SAEA required only about half the evaluations, which amounts in a saving of about 50% computational time. Note that the costs of surrogate building and other algorithm operators are negligible compared to the true function evaluations, and hence computationally effort is largely determined by the number of true function evaluations.

The average improvement obtained by SAEA compared to EA over different number of evaluations is shown in Fig. 4(d). This is calculated as $(f_{EA} - f_{SAEA}) / f_{EA} \times 100$, where f_{EA} and f_{SAEA} are average weight values (across multiple runs) obtained by EA and SAEA respectively, as shown in Fig. 4(b). The difference is 0 until 60 evaluations since the surrogate is first invoked after 5 generations, i.e., $N \times I_{train}$ = 60 evaluations. Significant improvements can be observed thereafter.

4.3 Formulation 2

Next, a modified formulation of the study presented above, referred to as formulation 2, is proposed and studied.

Given that the optimum design for the formulation 1 occurs at the lower bound of each variables, it indicates that there is a further scope for the reduction of weight. Therefore, we shift the lower bounds to the lowest value of practically allowable height in Table 1, i.e., 100 mm. The resulting bounds are presented in Table 4. This extension in the range of variables would typically make the problem harder to solve since the search space has now expanded.

Further, two constraints were added to the problem for the practicality of the design:

- First additional constraint results from the fact that the joint connecting Sections 3 and 5 is constructed sing a plate. For manufacturability of the design of the joint, the two heights should be equal. Thus, $h_3 = h_5$.
- The second constraint results from the fact that for effective load bearing, the height of Section 1 must be equal or larger than that of Section 2. Thus, a constraint $h_1 \geq h_2$ is added to reflect this.

The results using EA and SAEA are presented in Tables 5-6. Once again, it is seen that SAEA is able to achieve much lighter designs compared to EA within the given computational budget of 240 evaluations. On an average, the weight obtained using SAEA was 6.82% lower (1.1484 t vs. 1.2324 t) than that obtained using EA. The consistency of SAEA is also reflected in its lower standard deviation values. All reported designs satisfy the wind-induced constraints mentioned in Section 2.

The convergence rates corresponding to the multiple runs are shown in Fig. 5. It is evident that SAEA is able to converge towards a better solution faster than EA. From Fig. 5(b), it can also be seen that to obtain the same average result as the EA (1.816 t), SAEA required only about half the evaluations, which amounts in a saving of 50% computational time.

Table 6 Designs obtained by EA and SAEA for modified formulation. All web-heights are in mm

	$h1$	$h2$	$h3$	$h4$	$h5$
			EA		
Median	531	140	103	118	103
Best	335	100	102	101	102
Worst	886	101	588	492	588
			SAEA		
Median	387	100	100	100	100
Best	106	100	100	122	100
Worst	683	101	100	209	100

The average improvement obtained by SAEA compared EA over different number of evaluations (different stages of the search) is shown in Fig. 5(d). The difference is 0 until 60 evaluations since the surrogate is first invoked after 5 generations, i.e., $N \times I_{train}$ = 60 evaluations.

Overall, the above numerical experiments show the advantage of the surrogate-assisted evolutionary algorithm for the structural optimization of long-span portal frame. For both the formulations, SAEA was able to show significant reduction in weight values for the same amount of computational budget. At the same time, it was able to match the performance of EA with merely half the number of evaluations.

5. Summary and future work

A surrogate-assisted evolutionary algorithm was proposed to optimize the design of a long-span portal-rigid frame for weight, subject to the stability and strength requirements under dynamic wind loading. The web heights of I shape cross-sections were treated as discrete variables in the study. The computation of wind-induced displacements through simulation on SAP2000 is expensive, and hence to reduce the optimization time, surrogate modeling was utilized. Based on recent developments in the field of surrogate based techniques, multiple spatially distributed surrogate-assisted evolutionary algorithm (SAEA) was employed to solve the problems. Consequently, the SAEA uses not one but multiple (in this case three) different models that locally approximate the objective and constraint functions to the best accuracy in different regions of the

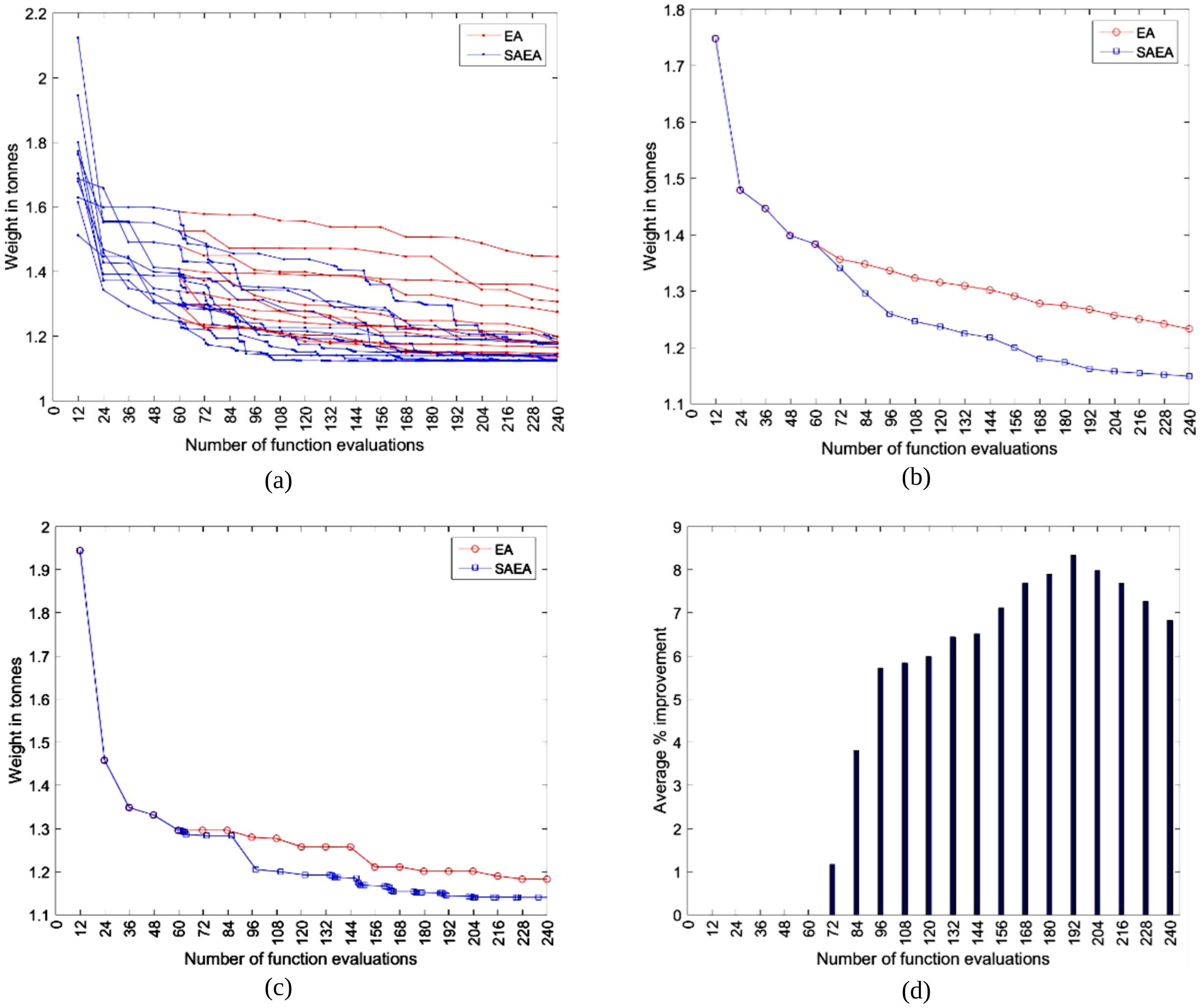


Fig. 5 Convergence for modified formulation over 11 runs: (a) all; (b) average; (c) median; (d) average improvement in weight value obtained by SAEA over EA

design space. In recent studies, such techniques have been demonstrated to offer more flexibility and accuracy in representation of complex functions. Numerical experiments were performed on two different formulations of the problem; the second one comprising different variable bounds and two additional constraints for practical design compared to the first. The results indicate that the SAEA was able to find designs with significantly reduced weight compared to EA within the fixed computational budget. At the same time, it reduced the time required to obtain the performance similar to EA by almost half. The standard deviations of SAEA results were also very low, reflecting on its consistency, and hence the confidence on it while using it for practical structural optimization problems. Thus, the proposed approach shows significant potential for application in optimization of structures for realistic scenarios.

A number of future directions can be identified from the study. While the study used a surrogate based EA, similar mechanisms could be implemented in other forms of population based algorithms, such as differential evolution, particle swarm optimization, etc. A combination of global and local search techniques could also be investigated to expedite the convergence further. Robustness considerations could be further added as objectives/constraints in the formulation to make the design more resistant to uncertainties in design and operating environment. One more direction to greatly enhance the application confidence of these optimization algorithms in real structures will be doing experiment to validate those numerical calculations. These are currently being investi-gated by the authors.

Acknowledgments

The research has been supported by National Natural Science Funds of China (Grant No. 51508118), Natural Science Foundation of Guangdong Province, China (Grant No.2016A030310261), Science and Technology Planning Project of Guangdong Province, China (Grant No. 2016B050501004), and Guangzhou Science and Technology Development Funds for Municipal Universities (Grant No. 1201431041). The authors would also like to thank Dr. Amitay Isaacs for providing the SAEA codes.

References

Adeli, H. and Park, H.S. (1996), "Hybrid CPN–neural dynamics model for discrete optimization of steel structures", *Comput.-Aided Civil Infra. Eng.*, **11**(5), 355-366.

Artar, M. and Daloglu, A.T. (2015a), "Optimum design of composite steel frames with semi-rigid connections and column bases via genetic algorithm", *Steel Compos. Struct., Int. J.*, **19**(4), 1035– 1053.

Artar, M. and Daloglu, A.T. (2015b), "Optimum design of steel space frames with composite beams using genetic algorithm", *Steel Compos. Struct., Int. J.*, **19**(2), 503-519.

Aydin, Z. and Cakir, E. (2015), "Cost minimization of prestressed steel trusses considering shape and size variables", *Steel Compos. Struct., Int. J.*, **19**(1), 43-58.

Begg, D.W. and Liu, X. (1998), "Algorithms for optimal design of smart structural systems", *Comput.-Aided Civil Infra. Eng.*, **13**(6), 415-423.

Bhattacharjee, K.S., Singh, H.K. and Ray, T. (2016), "Multi-objective optimization with multiple spatially distributed surrogates", *J. Mech. Des.*, **138**(9), 091401.

Camp, C., Pezeshk, S. and Cao, G. (1998), "Optimized design of two-dimensional structures using a genetic algorithm", *J. Struct. Eng.*, **124**(5), 551-559.

Chan, C.M., Grierson, D.E. and Sherbourne, A.N. (1995), "Automatic optimal design of tall steel building frameworks", *J. Struct. Eng.*, **121**(5), 838-847.

China Association for Engineering Construction Standardization (2012), Technical specification for steel structure of light-weight building with gabled frame (CECS102:2012).

Computers and Structures, Inc. Berkeley, C. (1997), SAP2000; Integrated Finite Element Analysis and Design of Structures. URL: https://www.csiamerica.com/products/sap2000

Deb, K. (2001), *Multi-Objective Optimization using Evolutionary Algorithms*, Volume 16, John Wiley & Sons.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002), "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Trans. Evol. Computat.*, **6**(2), 182-197.

Hayalioglu, M. and Degertekin, S. (2005), "Minimum cost design of steel frames with semi-rigid connections and column bases via genetic optimization", *Comput. Struct.*, **83**(21), 1849-1863.

Holmes, J. (2002), "Effective static load distributions in wind engineering", *J. Wind Eng. Ind. Aerodyn.*, **90**(2), 91-109.

Isaacs, A. (2009), "Development of optimization methods to solve computationally expensive problems", Ph.D. Thesis; The University of New South Wales, Australian Defence Force Academy, Canberra, Australia.

Isaacs, A., Ray, T. and Smith, W. (2009), "Multi-objective design optimisation using multiple adap- tive spatially distributed surrogates", *Int. J. Product Develop.*, **9**(1-3), 188-217.

Jin, Y. (2005), "A comprehensive survey of fitness approximation in evolutionary computation", *Soft Comput.-A Fusion Found. Methodol. Appl.*, **9**(1), 3-12.

Kameshki, E. and Saka, M. (2001), "Optimum design of nonlinear steel frames with semi-rigid connections using a genetic algorithm", *Comput. Struct.*, **79**(17), 1593-1604.

Kameshki, E. and Saka, M. (2003), "Genetic algorithm based optimum design of nonlinear planar steel frames with various semi-rigid connections", *J. Constr. Steel Res.*, **59**(1), 109-134.

Kaveh, A. and Shokohi, F. (2015), "Optimum design of laterally-supported castellated beams using CBO algorithm", *Steel Compos. Struct., Int. J.*, **18**(2), 305-324.

Kaveh, A., Bakhshpoori, T. and Barkhori, M. (2014), "Optimum design of multi-span composite box girder bridges using Cuckoo Search algorithm", *Steel Compos. Struct., Int. J.*, **17**(5), 705-719.

Kravanja, S. and Žula, T. (2010), "Cost optimization of industrial steel building structures", *Adv. Eng. Software*, **41**(3), 442-450.

Kravanja, S., Turkalj, G., Šilih, S. and Žula, T. (2013), "Optimal design of single-story steel building structures based on parametric MINLP optimization", *J. Constr. Steel Res.*, **81**, 86-103.

Li, J.J. and Li, G.Q. (2004), "Reliability-based integrated design of steel portal frames with tapered members", *Struct. Safety*, **26**(2), 221-239.

MathWorks (2014), Matlab 2014a. URL: www.mathworks.com

Moss, P., Dhakal, R., Bong, M. and Buchanan, A. (2009), "Design of steel portal frame buildings for fire safety", *J. Constr. Steel Res.*, **65**(5), 1216-1224.

Myers, R.H. and Montgomery, D.C. (1995), *Response Surface Methodology: Process and Product in Optimzation using Designed Experiments*, John Wiley & Sons, Inc., NY, USA.

Park, H.S. and Adeli, H. (1997), "Data parallel neural dynamics model for integrated design of large steel structures", *Comput.-Aided Civil Infra. Eng.*, **12**(5), 311-326.

Paya, I., Yepes, V., Gonázlez-Vidosa, F. and Hospitaler, A. (2008), "Multiobjective optimization of concrete frames by simulated annealing", *Comput.- Aided Civil Infra. Eng.*, **23**(8), 596-610.

Phan, D.T., Lim, J.B., Sha, W., Siew, C.Y., Tanyimboh, T.T., Issa, H.K. and Mohammad, F.A. (2013a), "Design optimization of cold-formed steel portal frames taking into account the effect of building topology", *Eng. Optimiz.*, **45**(4), 415-433.

Phan, D.T., Lim, J.B., Tanyimboh, T.T. and Sha, W. (2013b), "An efficient genetic algorithm for the design optimization of cold-formed steel portal frame buildings", *Steel Compos. Struct., Int. J.*, **15**(5), 519-538.

Queipo, N.V., Goicochea, J.V. and Pintos, S. (2002), "Surrogate modeling-based optimization of SAGD processes", *J. Petrol. Sci. Eng.*, **35**(1), 83-93.

Sacks, J., Welch, W.J., Mitchell, T.J. and Wynn, H.P. (1989), "Design and analysis of computer experiments", *Statist. Sci.*, 409-423.

Saka, M. (2003), "Optimum design of pitched roof steel frames with haunched rafters by genetic algorithm", *Comput. Struct.*, **81**(18), 1967-1978.

Senouci, A.B. and Al-Ansari, M.S. (2009), "Cost optimization of composite beams using genetic algorithms", *Adv. Eng. Software*, **40**(11), 1112-1118.

Sgambi, L., Gkoumas, K. and Bontempi, F. (2012), "Genetic Algorithms for the Dependability Assurance in the Design of a Long-Span Suspension Bridge", *Comput.- Aided Civil Infra. Eng.*, **27**(9), 655-675.

Sgambi, L., Gkoumas, K. and Bontempi, F. (2014), "Genetic algorithm optimization of precast hollow core slabs", *Comput. Concrete, Int. J.*, **13**(3), 389-409.

Sun, W., Gu, M. and Zhou, X. (2015), "Universal Equivalent Static Wind Loads of Fluctuating Wind Loads on Large-Span Roofs Based on POD Compensation", *Adv. Struct. Eng.*, **18**(9), 1443-1459.

Tokyo Polytechnic University (2011), "Aerodynamic database of low-rise buildings", URL: http://www.wind.arch.t-kougei.ac.jp/ info center/windpres sure/lowrise/mainpage.html

Topal, U. (2012), "Frequency optimization for laminated composite plates using extended layerwise approach", *Steel Compos. Struct., Int. J.*, **12**(6), 541–548.

Wang, G.G. and Shan, S. (2007), "Review of metamodeling techniques in support of engineering design optimization", *J. Mech. Des.*, **129**(4), 370-380. URL: http://link.aip.org/link/?JMD/129/370/1

Wu, J., Donga, C., Xua, A. and Fua, J. (2012), "Structural optimization of long span portal-rigid frames under wind action", *Proceedings of the 7th International Colloquium on Bluff Body Aerodynamics and Applications* (*BBAA7*), Shanghai, China, September, pp. 1584-1593.

CC