Scientia Iranica A (2018) 25(5), 2480–2487

SCIENTIA IRANICA

Sharif University of Technology

Scientia Iranica

Transactions A: Civil Engineering

http://scientiairanica.sharif.edu

Optimal domain decomposition using the global sensitivity analysis-based metaheuristic algorithm

A. Kaveh[a,*] and V.R. Mahdavi[b]

a. *Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran, P.O. Box 16846-13114, Iran.*

b. *School of Civil Engineering, Iran University of Science and Technology, Narmak, Tehran, P.O. Box 16846-13114, Iran.*

Received 6 October 2016; accepted 30 May 2017

KEYWORDS

Optimal domain decomposition;
Global Sensitivity Analysis-Based (GSAB);
k-median method;
Finite-element meshes.

Abstract. In this paper, an efficient approach is presented for finding optimal domain decomposition in conjunction with k-median method. Using the clique graph, the connectivity properties of finite-element meshes are represented. In order to divide the nodes of the graph or the meshes of the finite-element model into k subdomains, k-median approach is employed. For optimal subdomaining, a recently developed metaheuristic algorithm, called Global Sensitivity Analysis-Based (GSAB), is utilized. The performance of the proposed method is investigated through three finite-element models to minimize the cost of the k-median problem. A comparison of the numerical results obtained using the proposed method with those obtained by standard Colliding Bodies Optimization (CBO) and Particle Swarm Optimization (PSO) algorithms indicates that the proposed technique is capable of obtaining more promising solutions using less computational efforts.

1. Introduction

Parallel processing has been widely applied to large-scale problems, such as the analysis of hydraulic systems, electrical networks, and finite-element meshes. The aim of a parallel algorithm is to decompose the given domain of system into subdomains and analyse each subdomain by a processor. The dependency between substructures is resolved after completing the study of individual substructures.

Some algorithms were developed for optimal domain decomposition of finite-element models [1-6], and some review papers on this topic are also available [7,8]. Finding the medians of a graph is an NP-hard combinatorial optimization problem, and the exact solution to the problem is complex and highly time consuming for graphs with a large number of nodes. Such algorithms can be found in the works of [9-12]. Therefore, many approximate algorithms are developed for finding the medians of a graph. The simplest approach to dealing with domain decomposition is referred to as the k-median method [7]. In this method, a graph is associated with the topological property of the finite-element models. Then, the optimal medians in the graph are selected, such that the sum of the distances of nodes to medians becomes optimum. Recently, methods have been developed using metaheuristics, such as Genetic algorithms [13,14], bionomic approaches [15], ant colony [16,17], particle swarm optimization, and colliding bodies optimization [18], for obtaining solutions to the k-median problem.

As a newly developed metaheuristic algorithm, the Global Sensitivity Analysis-Based (GSAB) algorithm is introduced for the design of structural problems [19]. In this method, the search space of the optimization is determined using the sensitivity indicator

*. *Corresponding author. Tel.: +98 21 44202710; Fax: +98 21 77240398*
E-mail address: alikaveh@iust.ac.ir (A. Kaveh)

doi: 10.24200/sci.2017.4508

of variables. Unlike many metaheuristic algorithms, in which all the variables are simultaneously changed in the optimization process, in this approach, the sensitive variables of the solution iteratively change more rapidly than the less sensitive ones in the search space. This algorithm utilizes simple formulation and requires no parameter tuning.

In this paper, an algorithm based on the k-median concept is presented for optimal domain decomposition of finite-element meshes with continuous variables using the GSAB algorithm. Computer programs are developed to perform this optimization, and three numerical examples with different domain shapes are presented to demonstrate the efficiency of the proposed method.

2. Mathematical formulation of the median problem

The aim of the k-median problem is to cut a node set, N, into k-node $N_k \in N$, such that the sum of the distances of nodes to the median nodes becomes minimum. The problem of k-median can be stated as optimizing a function which decomposes domain G into k subdomains, $G_1, G_2, \cdots, G_k$, where k is the number of subdomains [20]. The objective function, which must be minimized, is formulated as follows:

$$\sigma_0(N_k) = \sum_{j \in N} v_j d(N_k, j), \qquad (1)$$

where $\sigma_0(N_k)$ is called out-transmission of nodes N_k; N_k is the median node number; v_j is the weight of the node j; and $d(N_k, j)$ is defined as follows:

$$d(N_k, j) = \min[d(i', j)] : (i' \in N_k). \qquad (2)$$

Let i' be the node of N_k which corresponds to the minimum value of Eq. (1), and then node j is allocated to i'. A shortest-route tree is rooted in each node for obtaining the shortest distance between nodes [5].

In order to find the nodal numbers of the medians of a graph, the coordinates of the medians are considered as the variables of the optimization process. Then, the nearest nodes from this coordinate are selected as the medians of the graph. Otherwise, if nodal numbers are considered as optimization variables, due to the high number of meshes in finite-element models, the search space becomes very large; secondly, discrete variables should be used in the optimization process. Therefore, in this work, the proposed optimization algorithm is considered to use the continuous variables.

3. A global sensitivity analysis-based algorithm

This section introduces a Global Sensitivity Analysis-Based (GSAB) optimization algorithm, which is a single solution search method. The proposed algorithm is named as "Global Sensitivity Analysis (GSA)" because of determining the Sensitivity Indicator (SI) of decision variables for guiding the search boundaries of the algorithm.

The samples/populations of GSAB algorithm are used for two purposes: estimating the SI of decision variables and finding the single-solution of the algorithm. Since these samples are not updated iteratively, the proposed GSBA is studied within the single-solution metaheuristic category. The feasibility space of samples in the GSAB algorithm is updated for searching the optimal solution over several iterations. In each iteration, the feasibility space is updated using two values consisting of the sensitivity indicators and the global best sample. It is assumed that the problem is a minimization problem in R^D. The notations used are as follows:

S^t	The sample matrix in the tth iteration, $S^t = [X_i^t \mid i = 1, 2, \cdots, N]$;
X_i^t	The position of sample vector i in the tth iteration, $X_i^t = \{x_{ij}^t \mid j = 1, 2, \cdots, D\}$;
$X_{\min}$	The minimum allowable values vector of variables, $X_{\min} = \{x_{\min_j} \mid j = 1, 2, \cdots, D\}$;
$X_{\max}$	The maximum allowable values vector of variables, $X_{\max} = \{x_{\max_j} \mid j = 1, 2, \cdots, D\}$;
$f(X_i)$	The fitness of vector i;
UB^t	The upper search boundary vector of variables in the tth iteration, $UB^t = \{ub_j^t \mid j = 1, 2, \cdots, D\}$;
LB^t	The lower search boundary vector of variables in the tth iteration, $LB^t = \{lb_j^t \mid j = 1, 2, \cdots, D\}$;
BW^t	The bandwidth of search space of variables in the tth iteration, $BW^t = \{bw_j^t \mid j = 1, 2, \cdots, D\}$;
SF^t	The scale factor of bandwidth of search space in the tth iteration, $SF^t = \{sf_j^t \mid j = 1, 2, \cdots, D\}$;
$Sbest$	The global best sample (i.e., with lower fitness), $Sbest = \{sbest_j \mid j = 1, 2, \cdots, D\}$;
R	A random vector within [0,1].

3.1. Methodology

The following steps outline the main procedure to implement the GSAB.

Step 1. *Initialization:* The initial positions of samples are determined with random initialization in the search space:

$$X_i^0 = X_{\min} + R(X_{\max} - X_{\min}), \quad i = 1, 2, \cdots, N, \tag{3}$$

where X_i^0 determines the initial value vector of the ith sample, and N is the number of samples;

Step 2. *Calculation of the sensitivity indices of variables:* In this step, the outputs (the objective function of the optimization problem) are calculated first. The sensitivity analysis is conducted next for the generated samples, and the Sensitivity Indicators (SIs) of variables are calculated.

The most well-known methods for calculating the variance-based sensitivity indicators are the Monte Carlo simulations; however, these do not make full use of each output model evaluation. In order to calculate the variance-based sensitivity indicators from the given data, the scatter plot partitioning method can be utilized [21]. For this method, a single set of samples suffices to estimate all the sensitivity indicators. For estimating the variance-based sensitivity indices, suppose that there are N points/samples $\{X^1, \cdots, X^N\}$ and N model output samples $\{y^1, \cdots, y^N\}$ obtained using model $y = g(X)$. The variance of Y can be calculated by sample variance $V(y)$. For the sample bounds of X_i as $[b1, b2]$, let it be decomposed into s successive, equal-probability and non-overlapping subintervals $A_k = [a_{k-1}, a_k)$, with $k = 1, \cdots, s$, $b_1 = a_0 < a_1 < \cdots < a_k < \cdots < a_s = b_2$, and $\Pr(A_k) = 1/s$. Decompose the output samples $\{y^1, \cdots, y^M\}$ into s subsets according to the decomposition of X_i, where:

$$B_k = \{y^j | x_i^j \in A_k\}, \qquad k = 1, \cdots, s.$$

The conditional variance $V(Y|x_i \in A_k)$ can then be evaluated by the following:

$$V(Y|x_i \in A_k) = V(B_k). \tag{4}$$

The expected conditional variance, $E_{x_i}(V(Y|x_i))$, can now be evaluated approximately using the following relationship:

$$E_{x_i}(V(Y|x_i)) \approx \frac{1}{s}\sum_{k=1}^{s} V(B_k). \tag{5}$$

Ultimately, the sensitivity indicator of the ith variable, SI_i, is calculated as follows:

$$\mathrm{SI}_i = \frac{V(Y) - E_{X_i}(V(Y|X_i))}{V(Y)} = 1 - \frac{E_{X_i}(V(Y|X_i))}{V(Y)}. \tag{6}$$

In sensitivity analysis, SI_i varies between 0 and 1. The lower value of SI_i corresponds to the less influential X_i; the higher value of SI_i corresponds to the much influential X_i; for $\mathrm{SI}_i = 0$, X_i will have no influence on Y;

Step 3. *Definition of the search boundaries:* In the GSAB algorithm, the search boundaries are moved around to the global best sample (which is updated and memorized in each iteration), *Sbest*, to push the samples into the feasible search space. The search boundaries are also decreased based on the values of sensitivity variables, which are evaluated in the previous step. Hence, the upper and lower boundaries of the search space of variables in the $t+1$th iteration can be computed by the following:

$$\begin{aligned} \mathrm{UB}^{t+1} &= Sbest + \mathrm{BW}^t \times \mathrm{SF}^t \leq X_{\max}, \\ \mathrm{LB}^{t+1} &= Sbest - \mathrm{BW}^t \times \mathrm{SF}^t \geq X_{\min}, \end{aligned} \tag{7}$$

where BW^t and SF^t are the bandwidth and scale factor of boundaries in the tth iteration, respectively. Eq. (7) ensures that the current search space is moved around *Sbest* with bandwidth BW^t in the D-dimensional space. Vector BW^t can be calculated as follows:

$$\mathrm{BW}^t = \max(Sbest - \mathrm{LB}^t, \mathrm{UB}^t - Sbest). \tag{8}$$

For the algorithm to converge to a near-optimal solution, further exploitation (strong locality) is required to move the current solution towards the optimal one. In the proposed GSAB algorithm, this is achieved by using a scale factor, SF. For this purpose, once SI values of variables are calculated, the most sensitive variable, i.e., variable with high SI value, is identified for reducing the bandwidth; then, SF is calculated as follows:

$$\mathrm{SF}_j = \begin{cases} 1 - si_j & \text{if} \quad si_j = \max(\mathrm{SI}) \\ 1 & \text{Otherwise} \end{cases}$$

$$\forall j = 1, \cdots, D. \tag{9}$$

This equation shows that the bandwidth of the most sensitive variable decreases, while other bandwidths are constant in the tth iteration;

Step 4. *Replacement of the current samples:* In this step, the samples must be ensured to be inside the new search boundaries. For this purpose, the samples that exceed the boundaries are regenerated randomly in the new search boundaries as follows:

$$X_i^{t+1} = \begin{cases} X_i^t, & LB^{t+1} \leq X_i^t \leq UB^{t+1}, \\ LB^{t+1} + R(UB^{T+1} - LB^{t+1}) & \text{Otherwise}, \end{cases} \tag{10}$$

where $i = 1, 2, \cdots, N$, and t represents the iteration index.

Step 5. *Termination:* The optimization process is

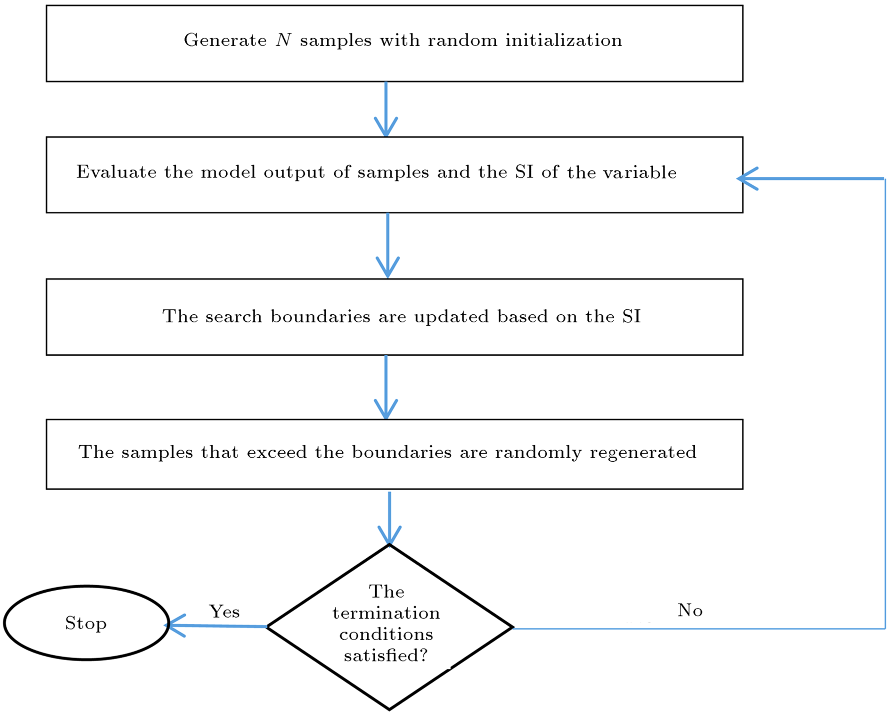


Figure 1. Flowchart of the GSAB.

repeated from Step 2 until a termination criterion, such as maximum iteration number or no improvement of the best sample, is satisfied. In the GSAB algorithm, if the maximum bandwidth of the search space, $\max(W)$, becomes smaller than 0.000001, the optimization process will be terminated. This is because the GSAB cannot change the search space of the agents. For the sake of clarity, the flowchart of the optimization procedure using the proposed GSAB is shown in Figure 1.

4. Examples

In this section, three numerical examples are studied. The topological properties of the finite-element models are transferred to the connectivity properties of graphs by the clique graphs [4]. A clique graph G of a FE mesh has its nodes in a one-to-one correspondence with the elements of the considered FE mesh, and two nodes of G are connected by an edge if the corresponding elements have at least one common node. In all of these examples, the weights of all the edges and the demands of all nodes are taken as unity, and the four-node rectangular meshes are considered for the FEMs.

In order to compare available metaheuristic algorithms, all of the examples are also solved using the Particle Swarm Optimization (PSO) and Colliding Bodies Optimization (CBO) [18]. In these examples, the number of agents is set to 20 individuals. Comparisons are made through the cost of k-median problem and the number of function evaluations as well as the convergence curves.

4.1. Example 1

A rectangular FEM for a plate, shown in Figure 2, is considered as the first example. The numbers of medians in this example are set to $k = \{3, 4, 5$, and $6\}$. As can be seen in Figure 2, the number of meshes is $2601(51 \times 51)$. The performance of the algorithms is tested on this model, and the results are depicted in Table 1. Figure 3 shows the optimal subdomains of FE meshes with different colors for different values of k. The evolution processes of the best fitness value obtained by three algorithms for $k = 6$ are also shown in Figure 4.

4.2. Example 2

The FEM of a rectangular plate with four openings consisting of 760 meshes, as shown in Figure 5, is considered. The number of medians considered in this example is set as $k = \{5$ and $10\}$. The performance of the proposed algorithm is tested on this model, and the results are depicted in Table 1. Figure 6 shows the optimal subdomains obtained using the GSAB algorithm for different values of k, and Figure 7 illustrates the convergence curves of the best results obtained for this example.

4.3. Example 3

In the last example, a circular plate with 2400 elements having one opening is considered (Figure 8). The

Table 1. The optimal cost and CPU time obtained using the CBO and PSO algorithms for the examples.

	The number of medians	PSO		CBO		Proposed method	
		Cost	Number of function evaluations	Cost	Number of function evaluations	Cost	Number of function evaluations
Example 1	$k = 3$	28122	4000	28097	4000	28099	1494
	$k = 4$	22186	4000	22113	4000	22113	2235
	$k = 5$	20589	4000	20505	4000	20507	3746
	$k = 6$	19219	4000	18924	4000	18927	2537
Example 2	$k = 5$	3809	6000	3787	6000	3798	1765
	$k = 10$	2876	6000	2589	6000	2563	3954
Example 3	$k = 5$	21745	8000	21508	8000	21400	3288
	$k = 10$	15742	8000	15071	8000	14759	4172

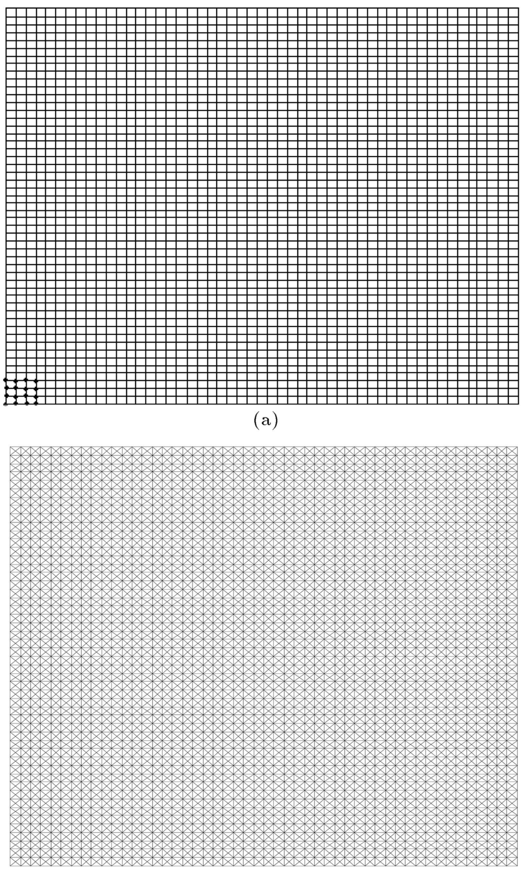


Figure 2. (a) A 2601 rectangular FEM for a rectangular plate. (b) The associated clique graph.

number of medians considered in this example is $k = \{5 \text{ and } 10\}$. Similar to the previous examples, the performance of all the algorithms is tested on this model, and the results are depicted in Table 1. Figure 9 shows also the optimal subdomains obtained using the GSAB algorithm for different values of k, and Figure 10

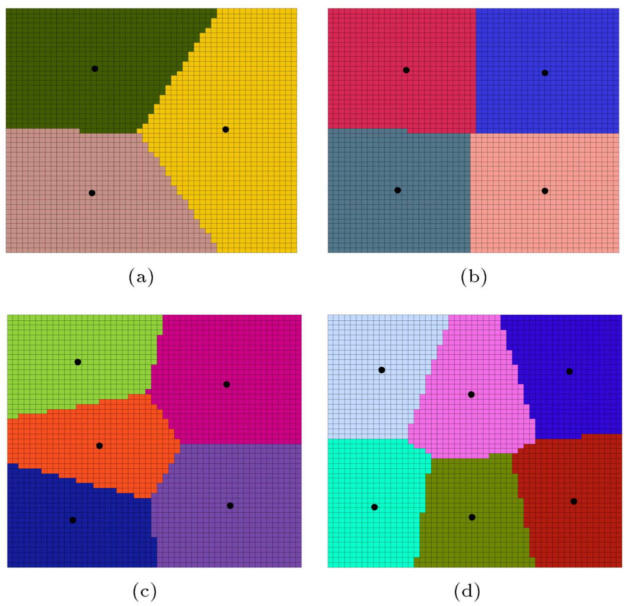


Figure 3. A FEM divided into k subdomains using the GSAB algorithm: (a) $k = 3$, (b) $k = 4$, (c) $k = 5$, and (d) $k = 6$.

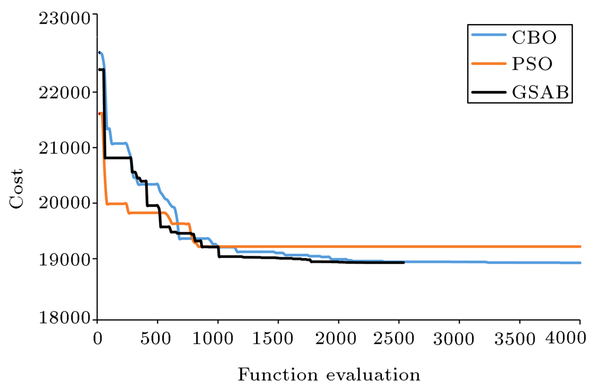


Figure 4. The convergence curves for $k = 6$ using three algorithms.

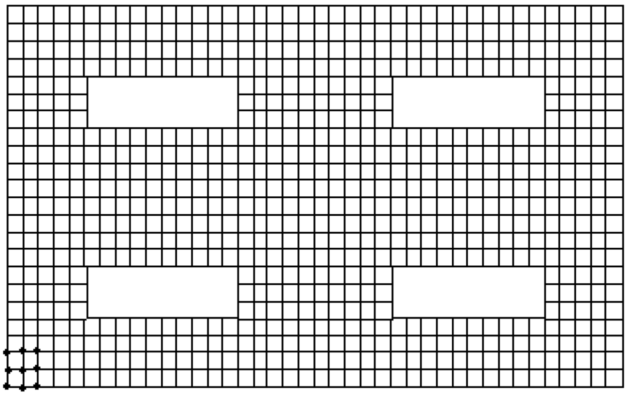
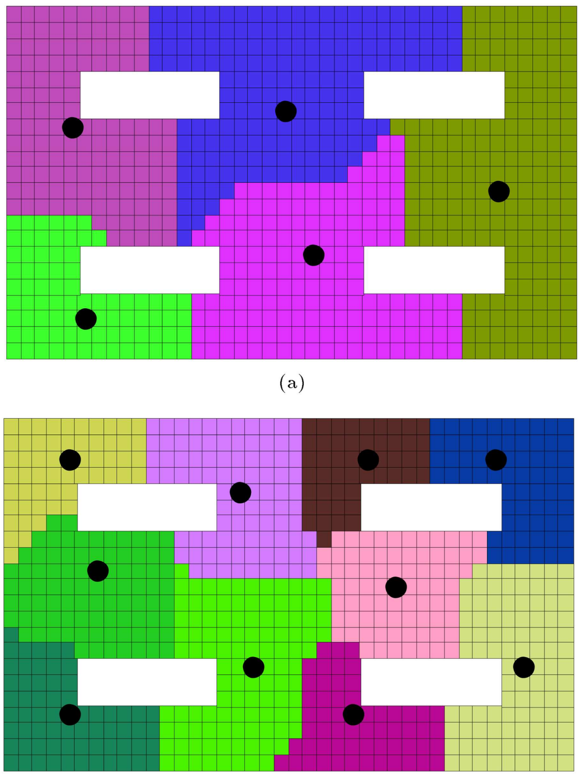


Figure 5. A plate FE mesh with 760 elements.



(a)

(b)

Figure 6. An FE mesh divided into k subdomains using the GSAB algorithm: (a) $k = 5$ and (b) $k = 10$.

illustrates the convergence curves of the best result obtained for this example.

4.4. Discussions

As can be seen in Figures 3, 6, and 9, the problem of finding the median of the considered FEMs is achieved using the proposed method. The optimal subdomains contain approximately equal numbers of meshes to balance the computational load between processors and, also, have good aspect ratios.

Table 1 shows the comparison of the results obtained using the GSAB algorithm and those of the CBO and PSO algorithms for all the examples. It can be seen in this table that the best costs obtained by the presented algorithm are better than those of the PSO and CBO algorithms, except for some cases with few number of function evaluations.

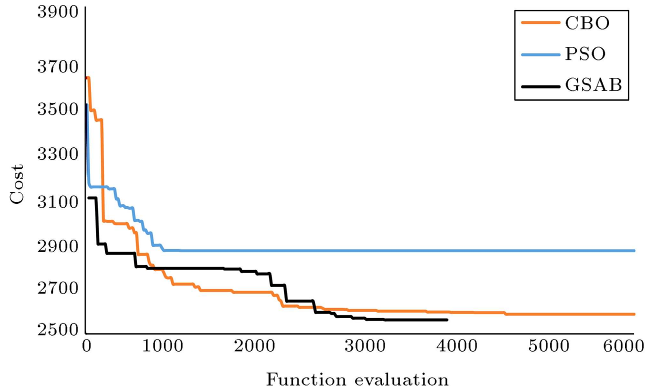


Figure 7. The convergence curves for $k = 10$ using three algorithms.

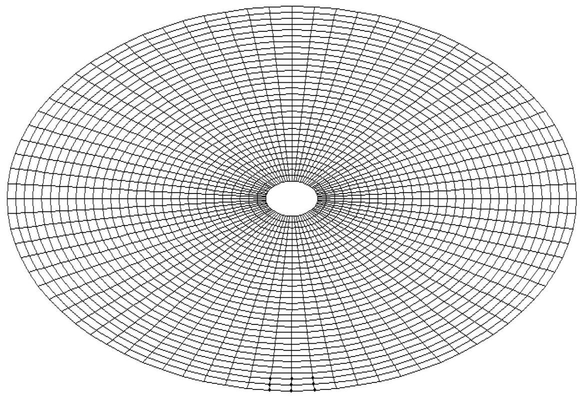
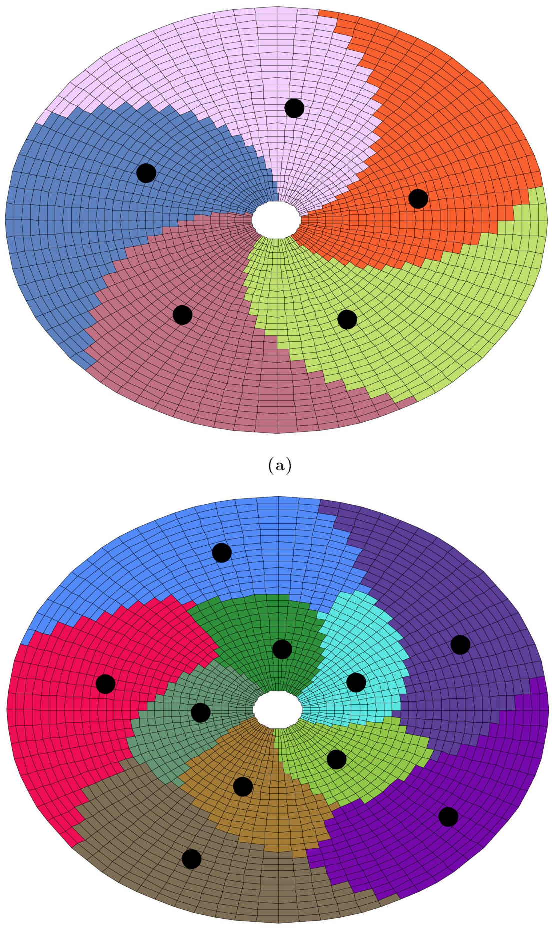


Figure 8. The FEM of a circular plate with one opening.

According to Figures 4, 7, and 10, although CBO and PSO are considerably faster in the early optimization iterations, GSAB converges to a significantly better decomposition without being trapped in local optima.

5. Concluding remarks

This paper proposed an optimal subdomain decomposition method for finite-element meshes based on the Global Sensitivity Analysis-Based (GSAB) algorithm and k-median method. In this method, the search space of the optimization was determined using the sensitivity indicator of variables, and the sensitive variables of solution were iteratively changed more rapidly than the less sensitive ones in the search space. In order to find the optimal subdomains, a clique graph was used to transform the connectivity properties of FE meshes into those of graphs. Then, the medians of the graph were selected based on optimization algorithm with the continuous variables.



(a) (b)

Figure 9. An FEM divided into k subdomains using the GSAB algorithm: (a) $k = 5$ and (b) $k = 10$.

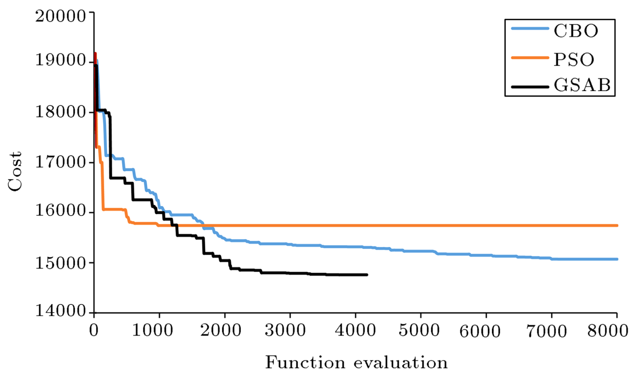


Figure 10. The convergence curves for $k = 10$ using three algorithms.

The validity and efficiency of the proposed method were illustrated using three test problems. The proposed algorithm solutions were compared with the best-known standard particle swarm optimization and colliding bodies optimization algorithms. The outcome was that the GSAB algorithm clearly outperformed the PSO and CBO algorithms with the few number of function evaluations.

References

1. Farhat, C. "A simple and efficient automatic FE mesh domain decomposer", *Comput. Struct.*, **28**, pp. 579-602 (1988).
2. Farhat, C. and Lesoinne, M. "Automatic partitioning of unstructured meshes for the parallel solution of problems in computational mechanics", *Int. J. Numer Methods Eng.*, **36**, pp. 745-764 (1993).
3. Khan, A.I, and Topping, B.H.V. "Sub-domain generation for parallel finite element analysis", *Proc. Civil Comp.*, B.H.V. Topping, Ed., Edinburgh, UK (1993).
4. Kaveh, A., *Optimal Structural Analysis*, John Wiley (Research Studies Press), 2nd Edn., Chichester, UK (2006).
5. Kaveh, A., *Computational Structural Analysis and Finite Element Methods*, Springer Press, Switzerland (2014).
6. Kaveh, A. and Roosta, G.R. "Domain decomposition for finite element analysis", *Communs Numer Methods Eng.*, **13**, pp. 61-71 (1997).
7. Kaveh, A. and Shojaee, S. "Optimal domain decomposition via p-median methodology using ACO and hybrid ACGA", *Finite Elem Anal Des.*, **44**, pp. 505-512 (2008).
8. Kaveh, A. and Sharafi, P. "Ant colony optimization for finding medians of weighted graphs", *Int. J. Computer-Aided Eng. Softw.*, **25**, pp. 102-120 (2008).
9. Christofides, N. and Beasley, J.E. "A tree search algorithm for the p-median problem", *Europ J Oper Res.*, **10**, pp. 196-204 (1982).
10. Kariv, O. and Hakimi, S.L. "An algorithmic approach to network location problems", *SIAM J. Appl. Math.*, **37**, pp. 539-560 (1979).
11. Osman, I. and Christofides, N. "Capacitated clustering problems by hybrid simulated annealing and tabu search", *Int. Trans. Oper. Res.*, **1**(3), pp. 317-336 (1994).
12. Charikar, M., Guha, S., Tardos, E., and Shmoys, D.B. "A constant-factor approximation algorithm for the k-median problem", *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pp. 1-10 (1999).
13. Osman, A., Erhan, E., and Zvi, D. "An efficient genetic algorithm for the p-median problem", *Ann. Oper. Res.*, **122**, pp. 21-42 (2003).
14. Castro, V.E. and Vela Zquez, R.T. "Hybrid genetic algorithm for solving the p-median problem", Lecture Notes in Computer Science, paper presented at Selected papers from the *Second Asia-Pacific Conference on Simulated Evolution and Learning on Simulated Evolution and Learning*, **1585**, pp. 19-25 (1998).
15. Maniezzo, V., Mingozzi, A., and Baldacci, R. "A bionomic approach to the capacitated p-median problem", *J. Heurist.*, **4**, pp. 263-280 (1998).
16. McKendal, A.R. and Shang, J. "Hybrid ant systems for the dynamic facility layout problem", *Comput. Oper. Res.*, **33**, pp. 790-803 (2006).

17. Levanova, T.V. and Loresh, M. "Algorithms of ant system and simulated annealing for the p-median problem", *Auto Remote Control.*, **65**, pp. 431-438 (2004).

18. Kaveh, A. and Mahdavi, V.R. "Optimal domain decomposition using colliding bodies optimization and k-median method", *Finite Elem. Anal. Des.*, **98**, pp. 41-49 (2015).

19. Kaveh, A. and Mahdavi, V.R. "Optimal design of truss structures using a new optimization algorithm based on global sensitivity analysis", *Struct. Eng. Mech.*, Techno Press, **60**(6), pp. 1093-1117 (2016).

20. Jackson, L.E., Rouskas, G.N. and Stallmann, F.M. "The directional p-median problem; definitions, complexity, and algorithms", *Europ. J. Oper. Res.*, **179**, pp. 1097-1108 (2007).

21. Zhai, Q., Yang, J. and Zhao, Y. "Space-partition method for the variance-based sensitivity analysis", *Optimal Partition Scheme and Comparative Study, Reliab Eng Syst Safety*, **131**, pp. 66-82 (2014).

Biographies

Ali Kaveh was born in 1948 in Tabriz, Iran. After graduating from the Department of Civil Engineering at the University of Tabriz in 1969, he continued his studies on Structures at Imperial College of Science and Technology at London University and received his MSc, DIC, and PhD degrees in 1970 and 1974, respectively. He then joined the Iran University of Science and Technology. Professor Kaveh is the author of 595 papers published in international journals and 145 papers presented at national and international conferences. He has authored 23 books in Persian and 10 books in English published by Wiley, Research Studies Press, American Mechanical Society, and Springer.

Vahid Reza Mahdavi was born in 1988 in Zarand, Kerman, Iran. He graduated from Shahid Bahonar University of Kerman with BS degree in Civil Engineering. He obtained his MS and PhD degrees in Earthquake Engineering at Civil Engineering Department of Iran University of Science and Technology. He is now a postdoctoral researcher at Civil Engineering Department of Iran University of Science and Technology. His current research interests include meta-heuristic optimization algorithms and their applications in structural engineering.