

Interactive Multiobjective Metaheuristic algorithm integrating BP Neural Network for hybrid indices optimization problems

Junling Zhang

College of Economics and Management, Zhejiang Normal University, Jinhua China, 321004

zhangjunling@zjnu.cn

Keywords: Interactive decision making; metaheuristic algorithm; Hybrid indices optimization problems; Multiobjective optimization algorithm.

Abstract. In this paper, we focus on investigating effective approach for tackling the complex hybrid indices optimization problems (HIOPs). Firstly, we analyze the strategy that utilizes machine learning models for breaking restrictions on search ability of MOEAs employed in conventional optimization framework based on interactive evolutionary computation (IEC). Then we take a plant layout design problem as a typical instance of HIOPs to devise an effective mechanism for employing BP network to forecast tacit fitnesses of design solutions. Furthermore, we develop an interactive multiobjective metaheuristic algorithm (IMMA) integrating a BP neural network for HIOPs. We also develop a prototype system based on IMMA for plant layout design optimization and experimental results have verified effectiveness of the proposed IMMA.

Introduction

In practice, due to the increasing complexity in socioeconomic environments, many operational optimization problems not only include explicit objectives that can be quantified also include other tacit objectives that cannot be directly quantified, such as the decision maker's satisfaction in product designing problems [1-5], which have been recognized as instances of hybrid indices optimization problems (HIOPs) [6, 7]. According to [6, 8, 9], the complex HIOPs normally have the following main particular characteristics: problem domain is large; some objectives cannot be formulized as explicit functions; fitnesses of solutions that given by decision makers are noisy due to human fatigue; problem solving process needs to adopt man-machine interaction.

To tackle HIOPs, appropriate approaches should have functionalities in: providing man-machine interfaces; alleviating effect of noisy tacit fitnesses; controlling population diversity to maintain optimization algorithms' searching abilities. Because the methodology of Interactive Evolutionary computation (IEC) intrinsically incorporates human as one of its inputs and can attain search heuristics inherited from effective evolutionary algorithms [10], IEC-based approaches have been the mostly employed approach for accommodating HIOPs. Such as those methods developed in [8, 11] for single-tacit-objective optimization, and the interactive multiobjective optimization methods proposed in [4, 5, 12] by integrating conventional multi-objective evolutionary algorithms. However, when applying these methods to tackle complex HIOPs, the population size is normally restricted to no more than 20 [6, 10] due to decision makers' fatigue, which consequently imposes restrictions on search ability of traditional multiobjective heuristic optimization algorithms employed in IEC-based optimization framework.

Therefore, in order to improve optimization efficiency of interactive decision making approaches for processing HIOPs, in this paper, inspired by the effective bat algorithm (BA) [13] and NNIA [14] we propose an interactive multiobjective metaheuristic algorithm (IMMA). To break through the restrictions by limited population size on search ability of IMMA, we employ a typical BP neural network to learn from tacit fitnesses and then to forecast tacit fitnesses of individuals in population when decision makers are in fatigue. Finally, a prototype experimental system is developed to verify effectiveness of the proposed IMMA.

Formulation of HIOPs and the Plant Layout Design Problem

Formulation of HIOPs. According to reference [6, 7], the hybrid indices optimization problem (HIOP) can be formulized as the following Definition 1.

Definition 1: The HIOP can be symbolically denoted as following six tuple $\langle N, X, f, L, Y, P \rangle$, and

(1) N denotes the numbers of decision makers, generally $|N| \geq 2$, and $|N|=1$ for single decision maker.

(2) X denotes the set of solutions in the specific problem domain, $x = (x_1, x_2, \dots, x_n)$ denotes a feasible solution, and x_i denotes the value of i -th dimension, n denotes the dimension number in x .

(3) $f = (f_1, f_2, \dots, f_p, f_{p+1})$ denotes all the decision objectives. f_1, f_2, \dots, f_p denotes all the objectives which can be quantitatively modeling, such as the currency cost, time costs, etc. f_{p+1} denotes the objectives which cannot be quantitatively formulized and only can be judged by decision makers themselves qualitatively, such as the designers' expert satisfaction, customers' satisfaction, etc.

(4) $L(x) = (L(x_1), L(x_2), \mathbf{K}, L(x_m)) \leq 0$ denotes constraints.

(5) Y denotes the image set of X under the mapping of function f , that is $Y = \{ y \mid \exists x \in X, f(x) = y \}$.

(6) P denotes the satisfaction standards of decision makers.

HIOPs are intrinsically a kind of multiobjective optimization problems having qualitative fitness with noise given by decision makers. The major difference of problem solving process between HIOP and traditional multiobjective optimization problems lies in the incorporation of human interaction throughout the whole optimization process.

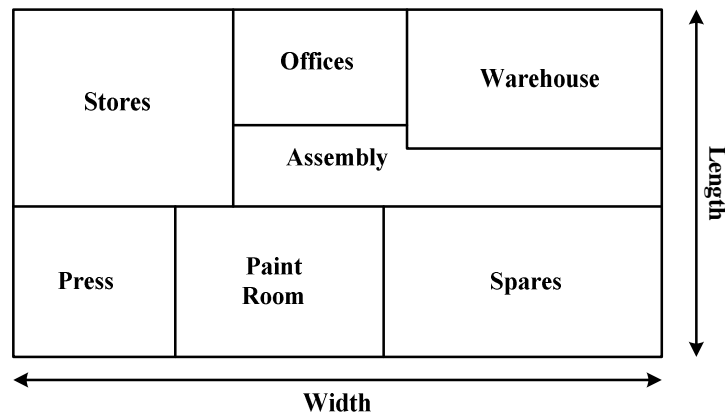


Fig. 1. Plant Layout Design Problem

Table 1 Plant layout design parameters

Room	Parameter	Label	Area
Stores area	Width	X_0	$A_1 = X_0(2.2 - X_1)$
Press area	Length	X_1	$A_2 = 2X_1X_2$
Press area	Width	X_2	
Paint room	Width	X_3	$A_3 = X_3X_4$
Paint room	Length	X_4	
Offices	Length	X_5	$A_4 = 2 \times [3.6 - (X_0 + X_6)] \times X_5$
Warehouse	Width	X_6	$A_5 = X_6X_7$
Warehouse	Length	X_7	
Spares area	-	-	$A_6 = X_1 \times [3.6 - (X_2 + X_3)]$
Assembly area	-	-	$A_7 = [3.6 - (X_0 + X_6)] \times [2.2 - (X_1 + X_5)] + X_6 \times [2.2 - (X_1 + X_7)]$

Plant Layout Design Problem. This special type of optimum design problem, as shown in Fig.1, is a typical instance of HIOPs [5]. According to reference [5], obviously, plant layout design problem normally have to consider both tacit and explicit objectives: (1) the cost of building is minimized, and (2) the subjective expert satisfaction that concentrates on the efficiency of the plant is maximized.

Therefore, in this paper, we take the plant layout design problem as the illustrative example under discussion. According to reference [5], in designing the most appropriate solution, there are eight parameters are required to represent different solutions. All eight parameters and their relations are listed in Table 1.

Note that the cost of each facility is proportional to its area, except the press area and the warehouse area, whose cost is doubled. All facilities except the assembly area should be rectangular. The lengths of the press area, paint room and spares area are kept the same. Overall size of the plant is constant. As a result, according to Table 1, the function of the cost of building to be minimized can be formulated as

$$F = \sum_{i=1}^7 A_i . \tag{1}$$

On the other hand, the assessments on individual design solution will be given as preferences by decision makers through interaction with multiobjective optimization algorithm.

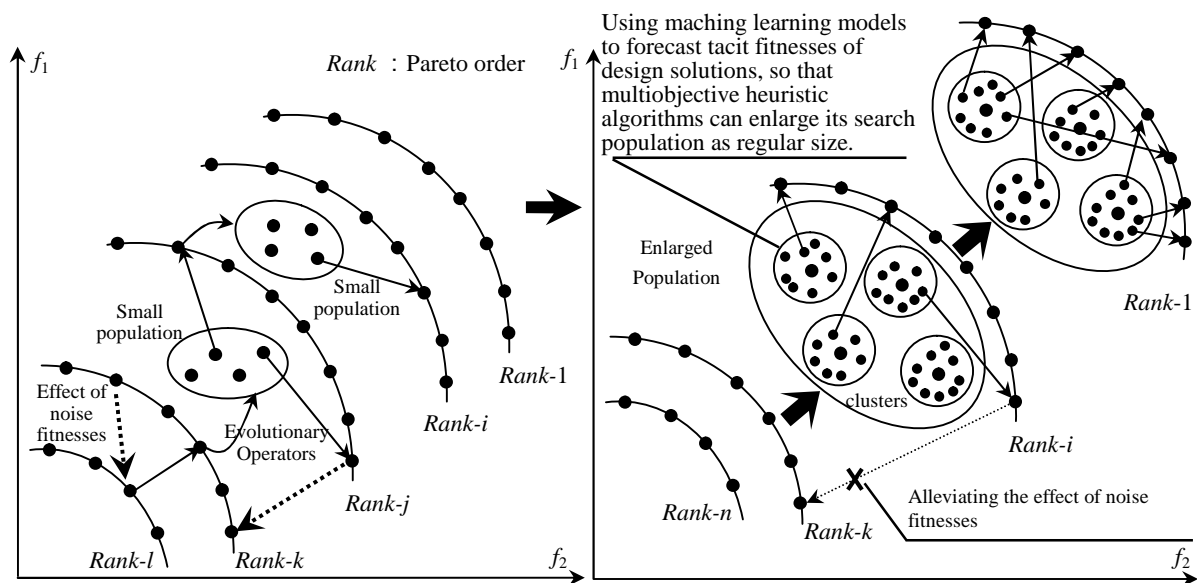


Fig.2. Strategy for breaking restrictions on search ability of MOEAs by using machine learning models

BP Network and its Application to Forecast Tacit Interval-valued Fitness

Strategy for breaking restrictions on search ability of multiobjective heuristic algorithms. In conventional multiobjective heuristic algorithms, such as MOEAs, normally the size of their populations are set larger than 30, sometimes more than 100. However, due to the user fatigue phenomenon in IEC based decision making framework, the population size of its employed multiobjective heuristic algorithms are restricted down to small than 20 individuals. Consequently, the search ability of conventional algorithms is reduced to a great extent.

In this subsection, inspired by reference [15], we here employ BP neural network to deduce effective strategy for breaking restrictions on search ability of multiobjective heuristic algorithms. During interaction with optimization algorithms, decision makers normally will get into the stage of

fatigue, and then their preferences will be biased by fatigue noise. Obviously, all the tacit fitnesses presented by decision maker into algorithms should be treated as information that reflects decision maker' preferences or judgment on design solutions. Further, these tacit fitnesses can surely be utilized as input data to train machine learning models, such as BP neural network [16], so as to work as forecasting models to evaluate alternative design solutions when decision makers are in fatigue. In sum, the effective strategy for breaking restrictions on search ability of multiobjective heuristic algorithms is demonstrated in Fig. 2.

In Fig. 2, after obtaining trained machine learning models, we integrate these models into the IEC-based interactive decision making framework to provide forecasted tacit fitnesses $f_{forecasted}$. As such, employed heuristic algorithms can maintain its original search ability through running with regular population size.

Basic Processing Steps of BP Neural Network [16]. The training steps of BP neural network can be summarized as follows:

Step 1: Initialization. Based on the input data serials (X,Y), determine the number (n) of nodes in input layer, the number (l) in hidden layer, and the number (m) in output layer; initialize the weights w_{ij} that connects input layer and the hidden layer, and the weights w_{jk} that connects the hidden layer and the output layer; initialize the threshold a on hidden layer and the threshold b on output layer. Generate learning rate h randomly, and choose an appropriate activation function.

Step 2: Calculate the outputs of hidden layer. Based on the input vector X, the weighting vector w_{ij} and the threshold a , obtain outputs from hidden layer according to

$$H_j = f\left(\sum_{i=1}^n w_{ij}x_i + a_j\right), j = 1, 2, \dots, l, \quad (2)$$

where l denotes the number of nodes in hidden layer, f denotes the activation function adopted as following

$$f(x) = \frac{1}{1 + e^{-x}}. \quad (3)$$

Step 3: Calculate the results of output layer. Based on the outputs H from hidden layer, the weighting vector w_{jk} , and the threshold b , compute the forecasting results O , where

$$O_k = \sum_{j=1}^l H_j w_{jk} + b_k, k = 1, 2, \dots, m. \quad (4)$$

Step 4: Calculate the output error. Based on the forecasting results O and the expectation output Y, derive the error e of forecasting according to

$$e_k = Y_k - O_k, k = 1, 2, \dots, m. \quad (5)$$

Step 5: Update the weighing vector w_{jk} and the weighing vector w_{ij} . Based on the error e of forecasting, we can update the w_{ij} and w_{jk} by use of following equations:

$$w_{ij} = w_{ij} + hH_j(1 - H_j) \sum_{k=1}^m w_{jk}e_k, \quad i = 1, 2, \dots, n; \quad j = 1, 2, \dots, l; \quad k = 1, 2, \dots, m; \quad (6)$$

$$w_{jk} = w_{jk} + hH_j e_k, \quad j = 1, 2, \dots, l; \quad k = 1, 2, \dots, m; \quad (7)$$

Here, h denotes the learning rate.

Step 6: Update the threshold. By use of the error e of forecasting, we also can update the thresholds a and b according to

$$a_j = a_j + hH_j(1 - H_j) \sum_{k=1}^m w_{jk} e_k, \quad j = 1, 2, \dots, l; \quad k = 1, 2, \dots, m; \quad (8)$$

$$b_k = b_k + h e_k, \quad j = 1, 2, \dots, l; \quad k = 1, 2, \dots, m; \quad (9)$$

Step 7: Check termination condition. If satisfied, then return; else go to Step 2.

The figure shows a user interface for evaluating a plant layout design solution. It features a grid of cells representing the layout. To the right of the grid, there is a text box labeled "Your assessment interval is:" with the value "[2 2]" entered. Below the grid, there are two sets of radio buttons for selecting linguistic terms. The first set is labeled "Choose LowBound:" and the second is "Choose UpperBound:". Both sets have five options: "worst", "Almost Medium", "Medium", "better", and "best". The "Medium" option is selected for both the LowBound and UpperBound.

Fig. 3. Plant layout design solution with interval-valued tacit fitness

Table 2 Mapping Relations between Interval-Valued Assessments and Outputs States in Employed BP Neural Network

Linguistic terms	Label	Interval-valued Assessment	Scores	Mapped Output Sates in Applying BP Neural Network												
Worst	0	[0, 0]	0	1	0	0	0	0	0	0	0	0	0	0	0	0
↑ ↓		[0, 1]	0.5	0	1	0	0	0	0	0	0	0	0	0	0	0
Almost Medium	1	[1, 1]	1	0	0	1	0	0	0	0	0	0	0	0	0	0
↑ ↓		[1, 2]	1.5	0	0	0	1	0	0	0	0	0	0	0	0	0
Medium	2	[2, 2]	2	0	0	0	0	1	0	0	0	0	0	0	0	0
↑ ↓		[2, 3]	2.5	0	0	0	0	0	1	0	0	0	0	0	0	0
Better	3	[3, 3]	3	0	0	0	0	0	0	1	0	0	0	0	0	0
↑ ↓		[3, 4]	3.5	0	0	0	0	0	0	0	0	1	0	0	0	0
Best	4	[4, 4]	4	0	0	0	0	0	0	0	0	0	0	0	0	1

Application of BP Neural Network to forecast interval-valued tacit fitness of Plant Layout Design Solution. When confronted with assessing on the alternative plant layout design solutions, decision makers are often only willing to denote their preferences with interval values rather than crisp numbers. Therefore, in this paper, we employ linguistic terms to allow decision makers to express their assessments. The user interface for evaluating individual plant layout solution is devised as shown in Fig.3.

As can be seen from Fig. 3, 5-scale linguistic term set, i.e. {worst (0), almost medium (1), medium (2), better (3), best (4)}, is hired as labels for decision makers to express their preferences on each solution. Then the labels for chosen upper bound and lower bound are collected as an interval values, for example, firstly we collect interval value [0, 1] to represent [worst, almost medium], we next assign the medium value of [0, 1] as the final score for the evaluated solution, which is shown in Table 2 for clarity. Obviously, there are nine possible score degrees for a certain solution. In order to apply BP neural network to learn from those evaluations associated with corresponding design solutions, in table 2, we also devise a scheme to map the total nine possible score degrees to nine output states for BP neural network.

Interactive Multiobjective Metaheuristic Algorithm Integrating BP Neural Network

Recently, swarm intelligence has been studied in depth and a serial of metaheuristic inspired by swarm intelligence have been proposed [17], among which bat algorithm [18] as a promising methodology have been successfully applied to different areas [19]. Till now, according to the literature survey carried out in reference [18], there is still less attention on metaheuristic approaches based on bat algorithms for multiobjective optimization, much less interactive multiobjective metaheuristic algorithms for HIOPs. Therefore, in this section, inspired by bat algorithm and the effective multiobjective evolutionary algorithm NNIA [14], we propose an interactive multiobjective metaheuristic algorithm (IMMA), in which the above-discussed BP neural network is integrated to substitute decision makers to forecast tacit fitnesses of design solutions when they are in fatigue. The proposed IMMA is detailed as following Algorithm I.

Algorithm I. Interactive Multiobjective Metaheuristic Algorithm Integrating a BP Neural Network

Step 1: Configuration. Let POP denote the active population with POP_size individuals, ArchivedPOP denote the population containing all nondominated solutions found so far, and the AssessedPOP denote the database storing all individuals and their scores evaluated by decision makers.

Step 2: Initialization. Randomly generate POP_size individuals and store them in POP. Based on the interface as shown in Fig. 3, all individuals in POP are displayed to decision makers for obtaining their tacit fitnesses, then their explicit fitnesses are also calculated according to Eq.(1). Further, based on the tacit and explicit fitnesses, we can derive nondominated solutions and feed them into ArchivedPOP. Collect all evaluated solutions and their tacit fitnesses into AssessedPOP.

Step 3: Carry out with equal possibilities the following disturbing operations in Eq. (10) and Eq. (11) on POP.

$$x_{new} = mutate(x_{old}). \quad (10)$$

Here $mutate(\cdot)$ denotes the bit mutation operator in reference [14].

$$x_{new} = x_{old} + eA^t. \quad (11)$$

Here e is drawn from [-1, 1]. A^t abides by the strategy $A^t = aA^{t-1}$ employed in bat algorithm where a is a constant and normally is set to 0.9 [18].

Step 4: Randomly chose half POP to execute following flying operation:

$$f_i = f_{\min} + (f_{\max} - f_{\min})b, \quad (12)$$

$$x_i^t = x_i^{t-1} + (x^* - x_i^{t-1})f_i. \quad (13)$$

Here, $f_{\min} = 0$, $f_{\max} = 1$, \mathbf{b} is a random vector in $[0, 1]$ which is drawn from a uniform distribution. x^* is randomly selected from nondominated solutions in ArchivedPOP and holds best tacit fitness.

Step 5: If the size of AssessedPOP meet the minimum number required, then use AssessedPOP to train employed BP neural network. Check the fatigue signal controlled by decision maker and the readiness of trained BP neural network. If fatigue signal is not on or BP neural network does not meet the minimum success rate, proceed to Step 6; otherwise go to Step 7.

Step 6: Display all solutions in current POP to decision makers and collect their tacit fitnesses. Store all evaluated solutions and their tacit fitnesses into AssessedPOP.

Step 7: If fatigue signal is on, recall Eq. (12) and Eq. (13) to generate another $(z - 1) * \text{POP_size}$ solutions, where z is the expansion rate. Then, utilize trained BP neural network to forecast the fitnesses of all solutions in current POP.

Step 8: Update ArchivedPOP through nondomination operations on combination of current POP and ArchivedPOP. Update POP by crowding-distance operations [14] on ArchivedPOP.

Step 9: Check stop criteria, if met then stop else then go to Step 3.

Experiment

In this section, we use Matlab[®] and Java[®] to implement prototype experimental system, as shown in Fig. 4, to verify effectiveness of Algorithm I. In our prototype system, POP_size is set to 9, minimum size of AssessedPOP is set to 20, expansion rate z is set to 4, and the minimum forecasting success rate of employed BP neural network is set to 90%. Experiments are conducted under two scenarios: IMMA with or without support of BP neural network. The maximum generation number at which the prototype system stops is set to 20. Four students from related majors are organized to carry out experiments independently. Results are collected in Table 3 for comparison.

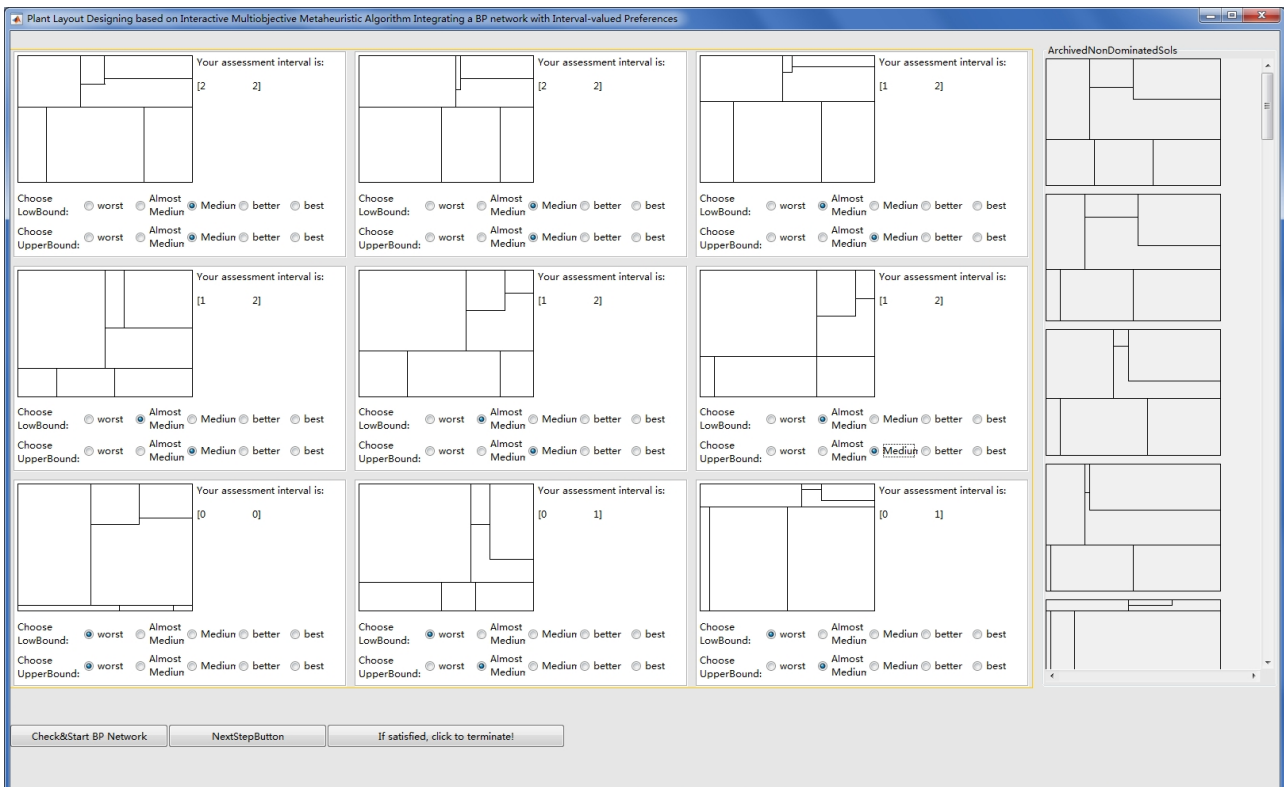


Fig. 4. Prototype experimental system for plant layout design optimization

Table 3 Comparative results of IMMA with or without support of BP neural network

Algorithms	items	1	2	3	4	Avg	S_{rate}
IMMA without BP neural network	T_{stop}	21	21	21	17	20	25%
	N_{eval}	189	189	189	153	180	
IMMA	T_{stop}	12/15	15/18	17/21	11/15	13.75/17.25	75%
	N_{eval}	108	135	153	99	123.75	

In Table 3, T_{stop} denotes the generation number at which system exists or decision maker terminate the system. N_{eval} denotes the total number of solutions that decision maker have evaluated before T_{stop} . As for the scenario of IMMA without support of BP network, all students are asked to evaluate all solutions throughout the whole optimization process. Only one student got satisfied and stopped the system at the 17th generation, the success rate S_{rate} is rather low in comparison with results from the scenario of IMMA with support of BP network. According to the data under indicator T_{stop} , students unanimously got into fatigue after 10th generation, but the variance of T_{stop} is relatively large. The main reason for this phenomenon is that readiness condition (the minimum forecasting success rate of employed BP neural network is set to 90%) prohibited activating employed BP neural network to forecast tacit fitnesses even though users were already in fatigue. Overall, as can be seen, IMMA is effective both in design optimization and in alleviating user's fatigue.

Conclusions

To effectively tackle HIOPs, we have proposed an interactive multiobjective metaheuristic algorithm (IMMA) and carried out investigation on the typical HIOP: plant layout design optimization. Aiming at alleviating user fatigue during evaluation, we have employed a BP neural network to forecast tacit fitnesses of design solutions. Furthermore, we also have developed a prototype experimental system based on IMMA for plant layout design optimization, and the results show that the proposed IMMA is effective. One of next research directions, one can incorporate combined forecasting machine models to improve the training efficiency in IMMA, such as the grey forecasting models working with small training datasets.

Acknowledgements

This work was supported in part by China Scholarship Council, the National Science Foundation of China (Nos. 71201145, 71271072, and 71331002), the Social Science Foundation of Ministry of Education of China (No. 11YJC630283).

References

- [1] H.S. Kim, S.B. Cho, Development of an IGA-based fashion design aid system with domain specific knowledge, in: Proceedings of IEEE SMC, Tokyo, Japan, 1999, pp. III-663–III-668.
- [2] H.S. Kim, S.B. Cho, Application of interactive genetic algorithm to fashion design, Eng Appl Artif Intel, 13 (2000) 635-644.
- [3] H.-S. Kim, S.-B. Cho, Knowledge-based encoding in interactivity genetic algorithm for a fashion design aid system, in: Proc. of GECCO, 2000, pp. 757.
- [4] A.M. Brintrup, J. Ramsden, H. Takagi, T. Ashutosh, Ergonomic Chair Design by Fusing Qualitative and Quantitative Criteria Using Interactive Genetic Algorithms, IEEE Trans on Evolu Comput, 12 (2008) 343-354.
- [5] A.M. Brintrup, J. Ramsden, A. Tiwari, An interactive genetic algorithm-based framework for handling qualitative criteria in design optimization, Comput Ind, 58 (2007) 279-291.

- [6] D. Gong, G. Hao, Y. Zhou, Y. Guo, Theory and Applications of Interactive Genetic Algorithms, National Defense Industry Press, Beijing, 2007.
- [7] P. Funes, E. Bonabeau, J. Herve, a. Et, Interactive multiparticipant tour allocation, in: Proceedings of the 2004 Congress on Evolutionary Computation, 2004, pp. 1699-1705.
- [8] G.-S. Hao, D.-W. Gong, Y.-Q. Huang, Interactive Genetic Algorithms Based on Estimation of User's Most Satisfactory Individuals, in: Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications, 2006, pp. 132 - 137.
- [9] G.-S. Hao, Y.-Q. Shi, Y.-Q. Huang, J.-H. Lv, Interactive Evolutionary Computation with Fitness Noise and Its Convergence Robustness, J Soft, 18 (2007) 2183-2193.
- [10] H. Takagi, Interactive Evolutionary Computation: Fusion of the Capabilities of EC Optimization and Human Evaluation, in: Proceedings of the IEEE International Conference on Intelligent Engineering system, San Diego, 2001, pp. 1275-1296.
- [11] G. Dun-Wei, H. Guo-Sheng, Z. Yong, S. Xiao-Yan, Interactive genetic algorithms with multi-population adaptive hierarchy and their application in fashion design, Appl Math Comput, 185 (2007) 1098-1108
- [12] A.M. Brintrup, J. Ramsden, A. Tiwari, Integrated qualitiveness in design by multi-objective optimization and interactive evolutionary computation, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2005, pp. 2154-2160.
- [13] X.S. Yang, A.H. Gandomi, Bat algorithm: a novel approach for global engineering optimization, Eng Compu, 29 (2012) 464-483.
- [14] M. Gong, L. Jiao, H. Du, L. Bo, Multiobjective Immune Algorithm with Nondominated Neighbor-based Selection, Evol Compu, 18 (2008) 455-461.
- [15] D.-w. Gong, X.-y. Sun, J. Ren, Surrogate Models Based on Individual's Interval Fitness in Interactive Genetic Algorithms, Chin J Electro, 18 (2009) 689-694.
- [16] Chinese-Matlab-Forum, Studies on Thirty Cases of Neural Networks by Using Matlab, Beihang University Press, Beijing, China, 2010.
- [17] A.H. Gandomi, X.S. Yang, A.H. Alavi, S. Talatahari, Bat algorithm for constrained optimization tasks, Neural Comput Appl, 22 (2013) 1239-1255.
- [18] X.S. Yang, X.S. He, Bat algorithm: literature review and applications, I J Bio-Ins Compu, 5 (2013) 141-149.
- [19] M. Chawla, M. Duhan, Bat Algorithm: A Survey of the State-of-the-Art, Appl Artif Intell, 29 (2015) 617-634.