



Identification of robust adaptation gene regulatory network parameters using an improved particle swarm optimization algorithm

X.N. Huang^{1,2} and H.P. Ren¹

¹Shaanxi Key Laboratory of CSCIP, Xi'an University of Technology, Xi'an, Shaanxi Province, China

²School of Physics and Electromechanical Engineering, HeXi University, Zhangye, Gansu Province, China

Corresponding author: H.P. Ren
E-mail: renhaipeng@xaut.edu.cn

Genet. Mol. Res. 15 (2): gmr.15028652

Received March 23, 2016

Accepted April 11, 2016

Published May 13, 2016

DOI <http://dx.doi.org/10.4238/gmr.15028652>

ABSTRACT. Robust adaptation is a critical ability of gene regulatory network (GRN) to survive in a fluctuating environment, which represents the system responding to an input stimulus rapidly and then returning to its pre-stimulus steady state timely. In this paper, the GRN is modeled using the Michaelis-Menten rate equations, which are highly nonlinear differential equations containing 12 undetermined parameters. The robust adaption is quantitatively described by two conflicting indices. To identify the parameter sets in order to confer the GRNs with robust adaptation is a multi-variable, multi-objective, and multi-peak optimization problem, which is difficult to acquire satisfactory solutions especially high-quality solutions. A new best-neighbor particle swarm optimization algorithm is proposed to implement this task. The proposed algorithm employs a Latin hypercube sampling method to generate the initial population. The particle crossover operation and elitist preservation strategy are also used in the proposed algorithm. The

simulation results revealed that the proposed algorithm could identify multiple solutions in one time running. Moreover, it demonstrated a superior performance as compared to the previous methods in the sense of detecting more high-quality solutions within an acceptable time. The proposed methodology, owing to its universality and simplicity, is useful for providing the guidance to design GRN with superior robust adaptation.

Key words: Parameter identification; Gene regulatory network; Robust adaptation; Particle swarm optimization algorithm

INTRODUCTION

Gene regulatory networks (GRNs) are used to investigate the interactions between genes; however, acquiring a systematic understanding of the interactions remains a challenge (Harbi and Harrison, 2014). The mathematical modeling of a GRN is a workable and effective way to address this issue (Politano et al., 2014; Tindall and Clerk, 2014). Adaptation is an important property of a GRN, which defines its reaction performance under variant input (Drengstig et al., 2008; David et al., 2013). Two characteristic quantities frequently used to describe the robust adaptation of a GRN are sensitivity and adaptation precision. They represent the system's ability to respond to the input stimulus and then return to its original steady state after being disturbed. Determining the parameters for a three-node GRN with robust adaptation is of biological universality and significance because most GRNs can be functionally reduced to a simplified framework with three nodes (Milo et al., 2002). Three-node enzyme GRNs with robust adaptation can be divided into two topological classes (Ma et al., 2009), which are the negative feedback loop with a buffering node (NFBLB) and incoherent feedforward loop with a proportioner node (IFFLP). It is worth noting that a well-studied example of a robust adaptation, i.e., *Escherichia coli* chemotaxis, is similar to the NFBLB topology GRN (Ma et al., 2009). Therefore, in the remaining sections of this paper, we have used the NFBLB GRN to exemplify the effectiveness of the proposed method.

For the three-node NFBLB type GRN model, there are 12 parameters to be determined to demonstrate its robust adaptation. Not all the parameter sets in this topology can confer the GRN with a robust adaptation. Ma et al. (2009) obtained only eight satisfactory solutions from 10^4 grid-division parameter sets using a Latin hypercube sampling (LHS) method, while Chiang et al. (2014) obtained 74 satisfactory solutions from 10^5 grid-division parameter sets using the LHS method. This observation indicates that it is very difficult for the LHS method to provide a large number of satisfactory solutions. Ren et al. (2015) proposed a multi-objective genetic algorithm (GA) to identify the undetermined and predetermined topology three-node GRNs.

In this study, we proposed a best-neighbor particle swarm optimization (PSO) algorithm, combined with an LHS-based initial population, crossover operation, and elitist preservation strategy to identify the robust adaptation GRN model parameters.

This paper is organized as follows: Section 2 provides a brief description of the problem, Section 3 enumerates the proposed algorithm, Section 4 presents the simulation results and the comparison, and Section 5 provides the conclusion.

MATERIAL AND METHODS

Problem description

Determining the parameter set that can confer the GRN with a robust adaptation is a generalized system parameter identification problem. The inherent nonlinear property of the GRN model prevents the use of conventional system identification methods for addressing this kind of problem. System biologists have proposed an ergodic-like searching method using the LHS method (Ma et al., 2009; Chiang et al., 2014). However, owing to its ergodic property, the LHS method is low efficiency; moreover, the results of the LHS depend on the grid division. Chiang et al. (2014) attempted to find more satisfactory solutions by increasing the grid division by one order. However, the number of satisfactory solutions was not increased significantly as compared to the increasing time cost. Moreover, this parameter identification is a multi-variable, multi-objective, and multi-peak optimization problem and the satisfactory parameter sets can be determined using optimization algorithms. We decided to commence addressing this problem using mathematical modeling of the NFBLB GRN.

Modeling NFBLB network using Michaelis-Menten rate equations

In this study, the three-node GRN with NFBLB topology was used to demonstrate robust adaptation, which in this case approximated the biological process of *E. coli* chemotaxis. The schematic diagram is shown in Figure 1, and can be described by the Michaelis-Menten rate equations provided in Equation 1.

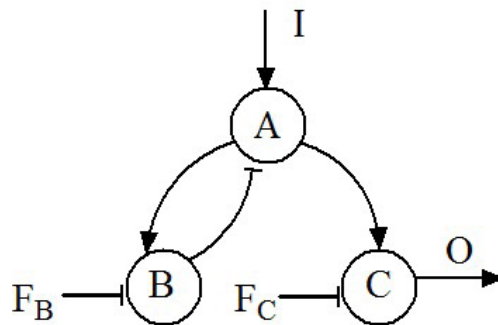


Figure 1. Three-node enzyme gene regulatory network (GRN) with a negative feedback loop with a buffering node (NFBLB) topology. The biological process of *Escherichia coli* chemotaxis is similar to NFBLB topology GRN. Node A receives input I , node C gives output O , and a third node B plays diverse regulatory roles.

$$\begin{aligned}
 \frac{dA}{dt} &= Ik_{IA} \frac{1-A}{1-A+K_{IA}} - Bk_{BA} \frac{A}{A+K_{BA}} \\
 \frac{dB}{dt} &= Ak_{AB} \frac{1-B}{1-B+K_{AB}} - F_B k_{F_B B} \frac{B}{B+K_{F_B B}} \\
 \frac{dC}{dt} &= Ak_{AC} \frac{1-C}{1-C+K_{AC}} - F_C k_{F_C C} \frac{C}{C+K_{F_C C}}
 \end{aligned}
 \tag{Equation 1}$$

where I denotes the input signal; X and $1-X$ denote the concentrations of enzyme X in the active and inactive states, respectively, where $X = A, B$, or C ; and F_B and F_C are the concentrations of the deactivating enzymes ($F_B = F_C = 0.5$), which transform the active states of B and C to the inactive states. The enzymatic regulation transforms its regulatory target between active and inactive states.

There are 12 parameters to be estimated in Equation 1, which are k_{IA} , K_{IA} , k_{BA} , K_{BA} , k_{AB} , K_{AB} , $k_{F_{BB}}$, $K_{F_{BB}}$, k_{AC} , K_{AC} , $k_{F_{CC}}$, and $K_{F_{CC}}$. The ranges of the catalytic rate constants k and Michaelis-Menten catalytic reaction constants K are $k \in [10^{-1}, 10^1]$ and $K \in [10^{-3}, 10^2]$, respectively. Furthermore, Equation 1 shows that the GRN is a highly nonlinear differential equation with as many as 12 undetermined parameters. Although I and O can be treated as the input and output of the conventional nonlinear dynamical system, the parameter identification of the GRN model is beyond the scope of the conventional system identification problem.

Robust adaptation

The adaptation of the GRN is measured quantitatively using two indices (Ma et al., 2009): the sensitivity and adaptation precision, given by Equations 2 and 3. The former index describes the system transient response peak with respect to the input change while the latter describes the system's ability to return to its original steady state in a timely fashion. The notations in Equations 2 and 3 are shown in Figure 2.

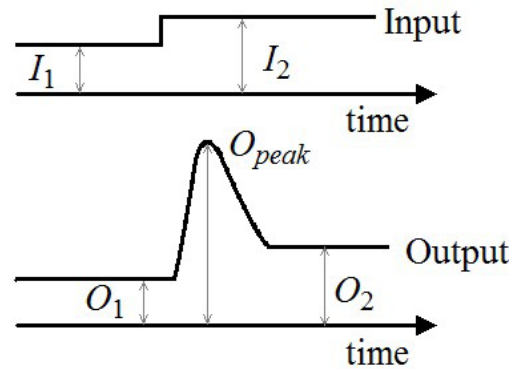


Figure 2. Input-output curve of gene regulatory network (GRN) used to define adaptation. I_1 and I_2 are initial and changed inputs; O_1 and O_2 represent steady-state output before and after stimuli, corresponding to inputs I_1 and I_2 , respectively; and O_{peak} is peak value of transient output, due to input change.

$$S = \left| \frac{(O_{peak} - O_1) / O_1}{(I_2 - I_1) / I_1} \right| \quad (\text{Equation 2})$$

$$P = \left| \frac{(I_2 - I_1) / I_1}{(O_2 - O_1) / O_1} \right| \quad (\text{Equation 3})$$

The robust adaptation is defined as follows: the transient peak output O_{peak} is more than 20% higher than the initial steady-state output O_1 is; the after-stimulus steady-state output O_2 is close to the initial steady-state output O_1 with a deviation less than 2% (Ma et al., 2009; Chiang et al., 2014; Ren et al., 2015). This indicates that $S > 1$ and $P > 10$.

Methods

Based on the GRN model given by Equation 1, this study focused on identifying the parameter sets required to confer the GRNs with a robust adaptation. The strategy for selecting the parameter set is a challenging task owing to the following two main reasons. First, the model has as many as 12 parameters, which are interdependent to have the desired adaptation. Second, it is a multi-objective optimization problem involving two conflicting objectives, which are the high sensitivity and adaptation precision (Ehrgott et al., 2014). In this study, we proposed a novel multi-objective best-neighbor PSO algorithm, combined with an LHS-based initial population, crossover operation, and elitist preservation strategy to obtain higher quality solutions than obtained with previous methods.

PSO algorithm

The PSO algorithm is a swarm intelligence evolutionary optimization method (Engelbrecht, 2006), where the swarm consists of a number of particles. Each particle represents a potential solution and flies within the search space with a velocity and position vector (v_i and x_i , respectively). The PSO algorithm is divided into two types, the best-global and best-neighbor PSO. The former is likely to have a fast convergence rate but may fall into the local optimum. The latter is likely to have a slow convergence rate but to avoid the local optimum. The best-global PSO updates its velocity and position according to its previous personal best position ($pbest_i$) and the global best position ($gbest$); the position and velocity update algorithms are given in Equations 4 and 5. The best-neighbor PSO updates its velocity and position according to its previous personal best position ($pbest_i$) and its neighbor best position ($nbest_i$); the position and velocity update algorithms are given in Equations 6 and 7.

$$v_i(t) = w \times v_i(t-1) + c_1 \times rand_1 \times (pbest_i - x_i[t]) + c_2 \times rand_2 \times (gbest - x_i[t]) \quad (\text{Equation 4})$$

$$x_i(t) = x_i(t-1) + v_i(t) \quad (\text{Equation 5})$$

$$v_i(t) = w \times v_i(t-1) + c_1 \times rand_1 \times (pbest_i - x_i[t]) + c_2 \times rand_2 \times (nbest_i - x_i[t]) \quad (\text{Equation 6})$$

$$x_i(t) = x_i(t-1) + v_i(t) \quad (\text{Equation 7})$$

where, N is the population size, w is the inertia weight, T is the generation number, $rand_1$ and $rand_2$ are two random values in $[0, 1]$, and c_1 and c_2 are the positive learning factors.

LHS algorithm

Population initialization using the LHS method is a random-based sampling method, which ensures that the samples are distributed across the entire range (Shu and Jirutitijaroen,

2011). Assuming that Z is an n -dimensional hypercube, and then Z_j^L and Z_j^U are the lower and upper bands of the j^{th} variable Z_j , where $j=1, \dots, n$. The LHS algorithm required to generate H samples (solutions) within this hypercube is given as follows: 1) Determine the sampling size H . 2) Divide the Z_j range, i.e., $[Z_j^L, Z_j^U]$, into H sub-ranges, so that $Z_j^L = Z_j^0 < Z_j^1 < Z_j^2 < \dots < Z_j^H = Z_j^U$, thereby the hypercube Z is divided into H^n sub-hypercubes. 3) Generate a matrix $\mathbf{D}_{H \times n}$, such that each column of $\mathbf{D}_{H \times n}$ is the random full permutation of the sequence $\{1, 2, \dots, H\}$. 4) Each row of $\mathbf{D}_{H \times n}$ consisting of n elements is used as an index to select one sub-hypercube, and a sample point (solution) is generated randomly within this sub-hypercube.

For example, Figure 3 illustrates the LHS method using $H=10$ and $n=2$, and the corresponding matrix $\mathbf{D}_{10 \times 2}$ is

$$\begin{bmatrix} 1 & 3 \\ 2 & 10 \\ 3 & 4 \\ 4 & 8 \\ 5 & 1 \\ 6 & 6 \\ 7 & 7 \\ 8 & 2 \\ 9 & 5 \\ 10 & 9 \end{bmatrix} \quad (\text{Equation 8})$$

This configuration of $\mathbf{D}_{10 \times 2}$ indicates that there is a two-dimensional (2-D) space searching problem (within $[0, 1; 0, 1]$ in Figure 3), and each dimension is divided into 10 sub-spaces indexed by $[1, 2, \dots, 10]$ and, therefore, 100 sub-squares are latticed in the whole space, which can be referred to by two index number. The black dots in the grids in Figure 3 correspond to the $\mathbf{D}_{10 \times 2}$ selection in Equation (8). Because the row index in Equation 8 is in ascending order in Figure 3, i.e., the first row corresponds to the bottom row in Figure 3. Only one grid is selected, which is determined by the indexes given by the first row of $\mathbf{D}_{10 \times 2}$, i.e., $[1, 3]$. For any other row in the grids in Figure 3, only one grid is selected and marked with a black dot; the 10 dots shown in Figure 3 are 10 solutions produced by the LHS method. Therefore, the samples generated by the LHS algorithm are distributed in the entire hypercube spaces and, therefore, it is a space-filling sampling method. The LHS simply guarantees the space filling but not their distribution.

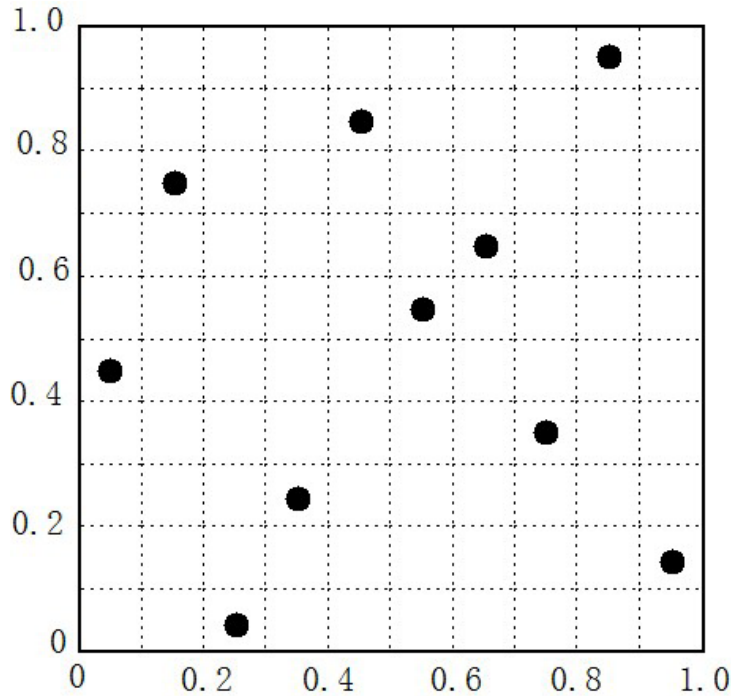


Figure 3. Example of Latin hypercube sampling (LHS) method with $H = 10$ and $n = 2$.

Best-neighbor PSO algorithm with LHS-based population initialization, crossover operation, and elitist preservation strategy

We proposed a novel best-neighbor PSO, characterized by an LHS-based initial population, particle crossover operation, and elitist preservation strategy. The flow chart of the algorithm is shown in Figure 4 and can be summarized as follows: 1) The population is initialized with N individuals using the LHS method. 2) The fitness of each particle is evaluated. 3) The elitists with the 10% top fitness are preserved, i.e., $P_{elite} = 0.1$. 4) The velocity and position of each particle are updated. 5) The crossover operation to the particles with probability $P_c = 0.5$ is performed. 6) The termination condition is tested. 7.1) Yes, indicates to stop iteration and show the result. 7.2) No, indicates to return to step (2).

Here, there are three key points: population initialization, fitness function selection, and crossover operation. Moreover, the elitist preservation strategy is performed to preserve the individuals with 10% top fitness, which directly proceed to the next generation instead of being involved in the subsequent operations including steps 4 and 5.

LHS-based initial population

Population initialization is important to the PSO algorithm because it contributes to the quality of the final solutions. Conventionally, the random initialization method is used to generate the initial population. However, for the multi-peak multi-parameter optimization

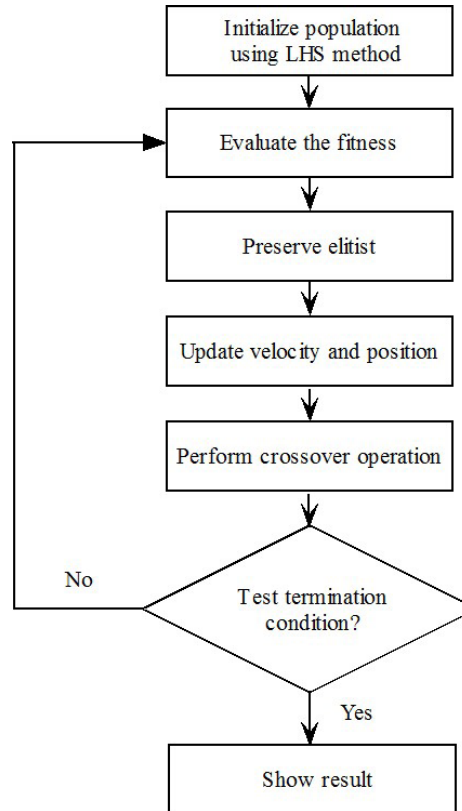


Figure 4. Flow chart of proposed algorithm.

problem, the conventional random initialization has the propensity to cause the initial particles to concentrate in a particular region and, thereby, lead to locally optimal solutions. Replacing the random population initialization with the LHS-based population initialization can enhance the diversity of the initial population.

In this study, the initialization strategy using LHS method was as follows: the catalytic rate constants k were divided into two ranges, $[10^{-1}, 10^0]$ and $[10^0, 10^1]$, from which each of the $N/2$ individuals were sampled. Then, the Michaelis-Menten catalytic reaction constants K were divided into five ranges, $[10^{-3}, 10^{-2}]$, $[10^{-2}, 10^{-1}]$, $[10^{-1}, 10^0]$, $[10^0, 10^1]$, and $[10^1, 10^2]$, from which each of the $N/5$ individuals were sampled, where N is the sampling size.

Pareto rank-based fitness function

In this study, two conflicting objectives required optimization, which were the sensitivity and adaptation precision that needed to be higher, and no single objective was considered more important than the other. There are multiple optimal solutions that can satisfy multiple objectives based on the different compromising strategies. These solutions are called non-dominated solutions or Pareto-optimal fronts. For the m objectives minimization problem,

PSO with crossover operation

The crossover operation is a genetic operator in the GA, which exchanges the partial chromosomes between two individuals (Schlottfeldt et al., 2015). Using a crossover strategy, the offsprings combine the characteristics of both parents, which improves their diversity. The crossover operation is performed using a specified number of individuals based on the crossover probability, P_c . In this study, the crossover operation is given as follows:

$$x_{child_1} = q \times x_{parent_1} + (1-q) \times x_{parent_2} \quad (\text{Equation 12})$$

$$x_{child_2} = q \times x_{parent_2} + (1-q) \times x_{parent_1} \quad (\text{Equation 13})$$

$$v_{child_1} = \frac{v_{parent_1} + v_{parent_2}}{|v_{parent_1} + v_{parent_2}|} * |v_{parent_1}| \quad (\text{Equation 14})$$

$$v_{child_2} = \frac{v_{parent_1} + v_{parent_2}}{|v_{parent_1} + v_{parent_2}|} * |v_{parent_2}| \quad (\text{Equation 15})$$

where q is the random value in $[0, 1]$; x_{parent_i} and x_{child_i} are the positions of the parent particle i and the position of offspring i , respectively; and v_{parent_i} and v_{child_i} are the speed of the parent particle i and the speed of offspring i , respectively, where $i = 1, 2$.

RESULTS

In this study, based on the three-node enzyme GRN with NFBLB topology, we identify the parameter sets that created GRNs with satisfactory adaptation of $S > 1$ and $P > 10$. The initial input was given as $I_1 = 0.5$, and then it is increased by 20%, that is, $I_2 = 1.2 \times I_1 = 0.6$ to produce the changed input.

To compare the performance of the proposed algorithm with that of existing methods, we used six methods to identify the parameter set of the GRN. These were the LHS method (Ma et al., 2009; Chiang et al., 2014); best-global PSO with random initial population (GPSO) (Engelbrecht, 2006); best-global PSO with LHS-based initial population (GPSO-LHS); best-global PSO with LHS-based initial population, elitist preservation strategy and crossover operation (ECGPSO-LHS); GA with LHS-based initial population and elitist preservation strategy (EGA-LHS) (Ren et al., 2015); and the newly proposed method, the best-neighbor PSO algorithm with LHS-based initial population, elitist preservation strategy, and crossover operation.

Algorithm parameters

The parameters of the four PSO methods were as follows: the learning factors $c_1 = c_2 = 1.5$; the inertia weight w , which linearly decreased from 0.9 to 0.4; the population size $N = 500$; and the maximum iteration times, $iter_{max} = 50$. Furthermore, for the ECGPSO-LHS and the proposed methods, the crossover and elitist preservation probability were $P_c = 0.5$ and $P_{elite} = 0.1$, respectively. For EGA-LHS method, the catalytic rate constants k and the Michaelis-Menten catalytic reaction constants K were encoded by 18 and 26 binary bits, respectively (Ren et al., 2015) (because of the similar parameter resolution (precision) as compared to LHS method; Chiang et al., 2014). In addition, the elitist preservation probability was $P_{elite} = 0.1$, the population size was also $N = 500$, the maximum iteration time was also $iter_{max} = 50$, the GA crossover probability was $p_c = 0.85$, and the GA mutation probability was $p_m = 0.02$. The sampling size was $H = 2.5 \times 10^4$ for the LHS method and all the parameters were determined using a trial and error procedure.

Performance comparison

All the results reported in this section were obtained based on 30 independent runs (Monte Carlo tests) to avoid the randomness of the algorithms and evaluate the algorithms statistically. Four criteria were used to compare the performance of all the algorithms.

- 1) *Time*, the total computation time (for 30 runs).
- 2) P^{find} , the possibility of finding satisfactory solutions, defined as:

$$P^{find} = run^{find} / 30 \quad (\text{Equation 16})$$

where run^{find} is the number of runs that detected, at least, one satisfactory solution in 30 independent runs.

- 3) $time^{nd}$, the average computation time of finding one non-dominated solution, defined as:

$$time^{nd} = Time / num^{nd} \quad (\text{Equation 17})$$

where num^{nd} is the number of non-dominated solutions (best solutions in the sense of the Pareto optimum) derived from 30 times runs.

- 4) P^{nd} , the percentage of non-dominated solutions, with respect to all non-dominated solutions found by all algorithms.

The comparisons of the results of the six methods are shown in Table 1. The six sub non-dominated sets derived using the six methods are shown in Figure 6, and only the solutions above the dotted line are the final non-dominated solutions with respect to all the solutions obtained by all algorithms. The square, circle, eight-line, triangle, and pentagram symbols represent the LHS, GPSO-LHS, EGA-LHS, ECGPSO-LHS, and the newly proposed algorithm, respectively.

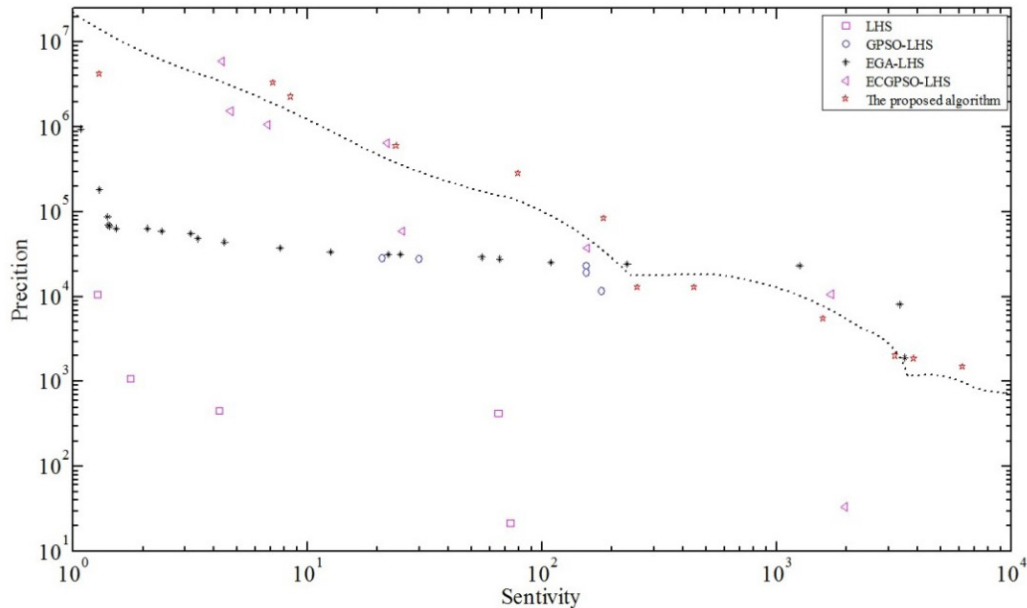


Figure 6. Distribution diagram of non-dominated solutions for six methods. Square, circle, eight-line, triangle, and pentagram represent the LHS, GPSO-LHS, EGA-LHS, ECGPSO-LHS, and newly proposed algorithm, respectively. No solution was obtained for GPSO in the 30 independent runs, therefore, nothing was plotted for this method. Final non-dominated solutions are located above the dotted line.

As shown in Table 1, the *Time* of the GPSO and the proposed algorithm were the shortest and longest, respectively in their reading values. However, it was challenging for the GPSO to determine satisfactory solutions from the $P^{find} = 0\%$. The proposed algorithm as well as the LHS and EGA-LHS methods all detected satisfactory solutions with the $P^{find} = 100\%$. The GPSO-LHS and ECGPSO-LHS methods discovered satisfactory solutions with the $P^{find} = 86.7\%$ and 70.0% , respectively. The index P^{nd} demonstrates the method's ability to discover the best solutions. The average computation time of finding one non-dominated solution, i.e., $time^{nd}$, represents the efficiency of discovering the best solutions. Figure 6 and Table 1 show that three methods, the ECGPSO-LHS, EGA-LHS, and the proposed methods (triangle, eight-line, and pentagram symbols, respectively) discovered non-dominated solutions while the other three methods did not. The proposed method discovered half of non-dominated solutions, i.e., 7 out of all the 14 non-dominated solutions. This suggests that the proposed method discovered the most high-quality solutions compared to the other five methods. For the average computation time of detecting one non-dominated solution, $time^{nd}$, we found that the EGA-LHS method performed the best while that the proposed method was slightly longer than that of the EGA-LHS method. Consequently, the proposed method discovered more of the best solutions (in the sense of Pareto rank) within acceptable average time than the other methods did.

Table 1. Comparison of six methods.

Algorithm	Time (min)	P^{find} (%)	P^{nd} (%)	$time^{nd}$ (min)
LHS (Ma et al., 2009; Chiang et al., 2014)	1311.6	100	0	inf
GPSO (Engelbrecht, 2006)	346.0	0	0	inf
GPSO-LHS	675.7	86.7	0	inf
ECGPSO-LHS	903.2	70.0	21.4	301.1
EGA-LHS (Ren et al., 2015)	737.5	100	28.6	184.4
Proposed algorithm	1385.9	100	50.0	198.0

LHS = latin hypercube sampling; GPSO = best-global particle swarm optimization with random initial population; GPSO-LHS = best-global particle swarm optimization with LHS-based initial population; ECGPSO-LHS = best-global particle swarm optimization with LHS-based initial population, elitist preservation strategy and crossover operation; EGA-LHS = genetic algorithm with LHS-based initial population, elitist preservation strategy; P^{find} = possibility of finding satisfactory solutions; P^{nd} = percentage of non-dominated solutions; $time^{nd}$ = average computation time of finding one non-dominated solution; inf = infinity.

CONCLUSION

In this study, a crossover operation from the genetic algorithm was introduced into the best-neighbor PSO algorithm together with an LHS-based population initialization and elitist preservation strategy. The proposed optimization approach successfully identified the parameter sets with a possibility equal to 1, which conferred the GRNs with robust adaptation. Compared to the conventional exhaustive searching LHS (Ma et al., 2009; Chiang et al., 2014), GPSO (Engelbrecht, 2006), GPSO-LHS, and ECGPSO-LHS methods, the newly proposed method discovered more high-quality solutions with a high efficiency. Compared to the EGA-LHS (Ren et al., 2015), the proposed method determined more best solutions with a slightly longer time.

ACKNOWLEDGMENTS

Research supported by the Innovative Research Team of Shaanxi Province (#2013KCT-04).

REFERENCES

- Chaudhry SB, Hung VC, Guha RK and Stanley KO (2011). Pareto-based evolutionary computational approach for wireless sensor placement. *Eng. Appl. Artif. Intell.* 24: 409-425. <http://dx.doi.org/10.1016/j.engappai.2010.07.007>
- Chiang AWT, Liu WC, Charusanti P and Hwang MJ (2014). Understanding system dynamics of an adaptive enzyme network from globally profiled kinetic parameters. *BMC Syst. Biol.* 8: 4. <http://dx.doi.org/10.1186/1752-0509-8-4>
- David L, Ben-Harosh Y, Stolovicki E, Moore LS, et al. (2013). Multiple genomic changes associated with reorganization of gene regulation and adaptation in yeast. *Mol. Biol. Evol.* 30: 1514-1526. <http://dx.doi.org/10.1093/molbev/mst071>
- Drengstig T, Ueda HR and Ruoff P (2008). Predicting perfect adaptation motifs in reaction kinetic networks. *J. Phys. Chem. B* 112: 16752-16758. <http://dx.doi.org/10.1021/jp806818c>
- Ehrgott M, Ide J and Schobel A (2014). Minmax robustness for multi-objective optimization problems. *Eur. J. Oper. Res.* 239: 17-31. <http://dx.doi.org/10.1016/j.ejor.2014.03.013>
- Engelbrecht AP (2006). Fundamentals of computational swarm intelligence. John Wiley & Sons Press.
- Harbi D and Harrison PM (2014). Interaction networks of prion, prionogenic and prion-like proteins in budding yeast, and their role in gene regulation. *PLoS One* 9: e100615. <http://dx.doi.org/10.1371/journal.pone.0100615>
- Ma W, Trusina A, El-Samad H, Lim WA, et al. (2009). Defining network topologies that can achieve biochemical adaptation. *Cell* 138: 760-773. <http://dx.doi.org/10.1016/j.cell.2009.06.013>
- Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, et al. (2002). Network motifs: simple building blocks of complex networks. *Science* 298: 824-827. <http://dx.doi.org/10.1126/science.298.5594.824>
- Politano G, Savino A, Benso A, Carlo SD, et al. (2014). Using Boolean networks to model post-transcriptional regulation in gene regulatory networks. *J. Comput. Sci.* 5: 332-344. <http://dx.doi.org/10.1016/j.jocs.2013.10.005>

- Ren HP, Huang XN and Hao JX (2015). Finding robust adaptation gene regulatory networks using multi-objective genetic algorithm. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. Early access on IEEE Xplore.
- Schlottfeldt S, Walter ME, Carvalho AC, Soares TN, et al. (2015). Multi-objective optimization in systematic conservation planning and the representation of genetic variability among populations. *Genet. Mol. Res.* 14: 6744-6761. <http://dx.doi.org/10.4238/2015.June.18.18>
- Shu Z and Jirutitijaroen P (2011). Latin hypercube sampling techniques for power systems reliability analysis with renewable energy sources. *IEEE Trans. Power Syst.* 26: 2066-2073. <http://dx.doi.org/10.1109/TPWRS.2011.2113380>
- Tindall MJ and Clerk A (2014). Modelling negative feedback networks for activating transcription factor 3 predicts a dominant role for miRNAs in immediate early gene regulation. *PLoS Comput. Biol.* 10: e1003597. <http://dx.doi.org/10.1371/journal.pcbi.1003597>