



A Two-Step Approach to the Search of Minimum Energy Designs via Swarm Intelligence

Frederick Kin Hing Phoa^(✉) and Tzu-Chieh Tsai

Institute of Statistical Science, Academia Sinica, Taipei, Taiwan
fredphoa@stat.sinica.edu.tw

Abstract. Recently, Swarm Intelligence Based (SIB) method, a nature-inspired metaheuristic optimization method, has been widely used in many problems that their solutions fall in discrete and continuous domains. SIB 1.0 is efficient to converge to optimal solution but its particle size is fixed and pre-defined, while SIB 2.0 allows particle size changes during the procedure but it takes longer time to converge. This paper introduces a two-step SIB method that combines the advantages of two SIB methods. The first step via SIB 2.0 serves as a preliminary study to determine the optimal particle size and the second step via SIB 1.0 serves as a follow-up study to obtain the optimal solution. This method is applied to the search of optimal minimum energy design and the result outperforms the results from both SIB 1.0 and SIB 2.0.

Keywords: Swarm intelligence · Nature-inspired metaheuristic method · Optimization · Two-step method · Minimum Energy Design

1 Introduction

Optimization problems in the real world are sometimes too complex and very challenging to solve within a reasonable amount of time. In order to obtain adequately good results, metaheuristics becomes a powerful method for this manner. Nature-inspired metaheuristic algorithms have been derived from the behavior of biological systems, physical or chemistry systems in nature [1]. A major class of metaheuristic is swarm intelligence (SI) [2], which is the collective intelligence behavior of self-organized and decentralized systems, including Genetic Algorithms (GA) [3], Artificial Bee Colony [4], Particle Swarm Optimization (PSO) [5] and many others listed in [6]. Among them, PSO is commonly used in solving many different types of high-dimensional optimization problems due to its simple form for implementation, its relatively small number of tuning parameters and a fast convergence towards optimum. However, the standard PSO method was not suitable for problems with the solutions in discrete domains.

Supported by Ministry of Science and Technology (MOST) of Taiwan (107-2118-M-001-011-MY3 and 108-2321-B-001-016) and Thematic Project of Academia Sinica, Taiwan (AS-TP-109-M07).

[7] proposed a new nature-inspired metaheuristic optimization method called the Swarm Intelligence Based (SIB) method. This algorithm works well in a wide range of discrete optimization problems, such as searching for circulant partial Hadamard matrices with maximum number of columns [8], $E(s^2)$ -optimal super-saturated designs [9], and optimal designs of computer experiments under multiple objectives [10]. In addition, SIB also performed well in optimization problems for continuous domains, like efficient construction of confidence sets and the confidence bands for target localization [11]. Besides, the computational efficiency of the SIB method can be improved via a smart initialization procedure [12]. Recently, [14] modified the standard framework of the SIB method by allowing the particle size to be changed during the search. The details of this augmented framework will be briefly reviewed in the next section.

Notice that the standard framework of the SIB method is limited to be applied due to the pre-defined particle sizes, but when the particle size is allowed to be changed, the SIB 2.0 suffers from high algorithmic complexity. The purpose of this paper is to provide a two-step approach that combines the standard framework and the augmented version of the SIB method, so that the particle size is allowed to be changed while the algorithm can be executed efficiently. We demonstrate our proposed method in the search of optimal Minimum Energy Design. The rest of the paper is organized as follows. We reviewed the essential components of the SIB method and the basic knowledge of the Minimum Energy Design in the second section. We introduce the proposed two-step method in the third section. In Sect. 4, we show a performance check via simulation studies, and we summarize the contribution and concluded remarks in the last section.

2 The Swarm Intelligence Based Method: A Review

2.1 The Swarm Intelligence Based (SIB) Method

A pseudo-code of the SIB method is given below.

- S0: Randomly generate a set of initial particles, each with different number of entries.
- S1: Evaluate objective function values of all particles.
- S2: Initialize the local best (LB) for each particle and the global best (GB).
- S3: For each particle, perform the MIX and MOVE operation.
- S4: If NO UPDATE in the previous MOVE operation: Perform the VARY and MOVE operation.
- S5: If NO UPDATE in the previous MOVE operation: Perform the random jump.
- S6: Update the LB for each particle and the GB among all particles.
- S7: If NOT CONVERGE or the stopping criterion is NOT REACHED, repeat (S3) to (S6).

The SIB method starts with a set of initialization steps. It includes the initialization of a user-defined (N) number of particles (S0), the evaluation of these

particles under an objective function of interest (S1), the definition of the LB for each particle, and the selection of the GB among all particles (S2). Additional parameters are being set before the end of the initialization, including the number of components being exchanged with the LB (q_{LB}), the number of components being exchanged with the GB (q_{GB}), and the maximum number of iteration allowed. The details on the setting of these parameters are referred to [7]. After the initialization steps, it enters the iteration steps for optimization.

The standard framework of the SIB method, abbreviated as SIB 1.0 thereafter, consists of all above step except S4. For each particle in S3, every iteration step starts with an exchange procedure called the MIX operation. It is an exchange procedure between the current particle and the best particles, both the particle's LB and the GB. In brief, q (q_{LB} when exchanged with the particle's LB and q_{GB} when exchanged with the GB) components are first selected in the best particle and add to the current particle, then q components among all components in the current particles are deleted. After the MIX operation, two new particles, namely *mixwLB* and *mixwGB*, are formed and they are compared to the current particle under the objective function in the MOVE operation. If one of the new particles has the best objective function value, the current particle will be updated to that new particle, otherwise, the random jump procedure that some components in the current current particle are exchanged to the randomly selected components is performed in S5. When the exchange procedures of all particles are completed, the updates of the LBs of all particles are performed followed by the update of the GB among all LBs (S6). If the GB does not converge to the optimal value of the objective function (if known), or the number of iteration has not reached its maximum number yet, the iteration will repeat. Readers who were interested in the details of the SIB 1.0 are referred to [7].

The wide use of the SIB 1.0 is hindered by its limitation of the fixed particle size. Note that the MIX operation adds and deletes exactly q number of components in the current particle, so the size is always fixed. It is sometimes a requirement in some applications, like the application in experimental designs [9], but in most cases, it becomes inefficient when one applies the SIB 1.0 multiple times for different particle sizes. Thus, it is desired to develop an enhanced framework of the SIB method, abbreviated as SIB 2.0, for problems with flexible run sizes during the optimization.

The key addition of the SIB 2.0 (S4) appears after the MIX and MOVE operations in S3. If the current particle is still the best among all three particles in MOVE operation in S3, or in other words, both best particles cannot further improve the current particle in this iteration, we make the size change on the current particle in S4 via the VARY operation first before considering the random jump. The VARY operation consists of unit shortening and unit expansion, and it results in a particle with reduced number of component and a particle with additional number of component respectively. Then we perform the MOVE operation to compare the objective function values among the current particle, the shortened particle and the expanded particle. Similar judgement is

used to determine whether particle is updated or the random jump is performed. Readers who were interested in the details of the SIB 2.0 are referred to [14].

3 The Two-Step Swarm Intelligence Based Method

As discussed in [14], the goal of developing SIB 2.0 is to avoid a multiple implementation of the SIB 1.0 for various particle sizes when the optimal particle size is unknown in prior. This greatly reduces the computational time as only one SIB algorithm is required to search for optimal solution together with the particle's optimal size. However, it is also true that if the optimal particle size is known in prior, SIB 1.0 performs better than SIB 2.0 under the same amount of searching time because of the much larger particle choices in SIB 2.0. Therefore, an ideal strategy of efficient use of the SIB method is to find the optimal particle size followed by the application of SIB 1.0 at that specific particle size. Thus, we propose our two-step SIB method as follows.

Input user-defined parameters: number of seeds N , number of iterations L , numbers of unit exchanges, and other parameters for specific problems.

Step 1: Perform SIB 2.0 to detect optimal particle size (L iterations).

Result: A suggested value of particle size k .

Part 2: Perform SIB 1.0 at particle size k to obtain the optimal solution ($0.2L$ iterations).

Result: GB particle as the optimal particle suggested in this method.

Here are some details on this two-step SIB method. The first step is basically a standard SIB 2.0 with a slightly different goal. Instead of finding the optimal solution via the objective function, it simply needs to detect the best choice of particle size that leads to good objective function values. Therefore, even though it is known that the solution of SIB 2.0 may not be as good as SIB 1.0 (known particle size in prior) at the same computational time, it is still good enough to detect a good particle size only. Next, the second step is a standard SIB 1.0 that the particle size is obtained from the first step. We suggest to use only $1/5$ number of iterations because we have inserted the best particle obtained from the first step into the initial particle set. This implies that the GB in the very first step has already been as good as all searches in the first step, so it is reasonable to save some iterations in the second step.

In fact, the idea of the second step in our proposed method is equivalent to the follow-up procedure in many industrial experiments. Thus, when compared to the SIB 2.0, our proposed method spends slightly more resources (20% more) to efficiently optimize the results based on the findings of the SIB 2.0 (the first step). On the other hands, our proposed method still saves a tremendous amount of computational time when compared to multiple implementation of SIB 1.0 at various particle sizes.

4 Demonstration: A Search of Minimum Energy Designs

4.1 A Brief Introduction to Minimum Energy Design (MED)

Computer experiment has become a new trend for many industrial experiments and scientific testings, especially for those with astronomical costs or infeasible procedures. Traditionally, Latin hypercube designs and uniform designs are two classical types of designs for computer experiments. The former design, constructed systematically via algebra, is particularly useful in grid-type domain while the latter design, obtained via computer search, enjoys a uniform distribution of particles over continuous domain. However, both designs are desired under an important assumption that there is its domain gradient is a uniform function, meaning that every single position in the domain is evenly important.

It is obvious that most domains do not have a uniform gradient in the real world, because some parts are more important than others when prior knowledge from experts are given. [13] proposed the minimum energy design (MED) for the computer experiments when the domain gradient follows a given function known in prior. When a defined number of particles are assigned to the domain, they suggest to put more particles in more important regions instead of evenly distributed all particles. This suggestion is reasonable because an important domain region deserves to spend extra particles to detect its value of interest.

[13] provides examples that the MEDs are obtained via computer searches. When the domain is in grid forms and the grid size is not large, it is certainly desired to search over all possible combinations to obtain the best particle orientation. However, when the grid size is too large that becomes infeasible to search over all possible combinations, or the grid size is continuous in nature, we need an efficient approach to obtain the adequately good MED, even if it is not optimal. Thus, we suggest to use our proposed two-step SIB method.

We recall some basic terminologies in MED below. Based on an analogy of the physical system, MED treated every design points as charged particles. The key idea (objective function) was to minimize the total potential energy of the system. In specific, an MED with its design points $\{x_1, \dots, x_n\}$ was optimal if it satisfied the criterion $\max_D \min_{i,j} \frac{d(x_i, x_j)}{q(x_i)q(x_j)}$ and the charge function was defined as $q(x) = \frac{1}{f(x)^{1/(2p)}}$, where p is the dimension and $f(x)$ is a desired density. In general, a good MED tried to place its design points as apart as possible under a modified distance weighted by the charge function $f(x)$.

4.2 Implementation of the Two-Step SIB Method

Instead of the computer search in [13], a recent work by [12] applied the SIB 1.0 method to obtain an improved result on the search of MEDs, and the search efficiency is also improved. Their results are under an assumption that the particle size is given, and they argued that it was normal because the experimental budget was usually fixed in prior. However, this argument fails to consider the possibility that less amount of resources sometimes can achieve similar performance, or

in other words, it is still able to save experimental resources by conducting an experiment via MED below the maximum possible budget.

In specific, our experimental domain is a three-dimensional cube that different locations (points) have different temperatures shown in Fig. 1. This temperature gradient is arbitrary for demonstration purpose. Each pair of points is apart from 0.2 units, so there were 4096 points in the cube. Our goal is to create a MED with unknown number of points in this experimental region. Notice that a evenly-distributed point allocation will obviously not satisfy the minimum energy criterion. For the initialization, we set the number of iteration in the first step as $L = 500$, the number of iteration in the second step as 100 ($0.2L$), and the number of MED sets $N = 204$ (about 5% of maximum number of points). We also initialize these particles with different sizes ranging from 3 to 20 (about 10% of N), and all exchange parameters are set as $q_{LB} = q_{GB} = q_s = q_e = 1$.

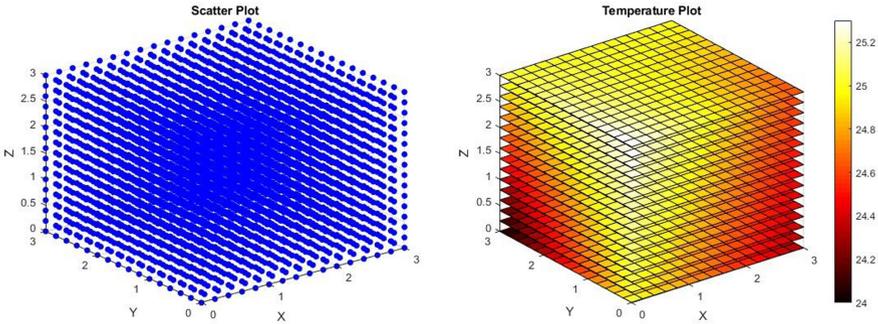


Fig. 1. Experimental region

4.3 Result

After 500 iterations of 204 particles, the GB particle resulted in the first step suggests that the optimal particle size is 7 and this particle has energy -11.831543 . Then we set $k = 7$ in the second step and an extra 100 iterations are run. Table 1 lists the coordinates of the seven points in the domain and the dots in Fig. 2 visually shows their locations in the domain. This GB particle has the minimum energy -12.856852 . The whole procedure can be done within 5 min using a standard personal computer.

In order to check if the MED suggested in this method is good, we conduct 18 SIB 1.0, each with different particle size ranging from 3 to 20 and the maximum number of iteration is 500. The energies of their GB particles are shown in Fig. 3. Although this figure suggests that $k = 8$ may have the lowest energy among all 18 k , the difference between the results of $k = 7$ (-12.33198) and $k = 8$ (-12.38761) is very small. Notice that an additional point may imply an increase in the experimental cost, the marginal improvement by adding one point

Table 1. Coordinates of positions with temperatures

	Node1	Node2	Node3	Node4	Node5	Node6	Node7
X	2.8	0.2	3.0	0.4	3.0	0.2	0.4
Y	2.6	0.4	2.8	3.0	0.0	2.8	0.2
Z	2.8	0.2	0.2	0.0	0.0	2.6	3.0
Temperature	24.9267	24.8709	24.6702	24.34	25.3	24.2576	24.1448

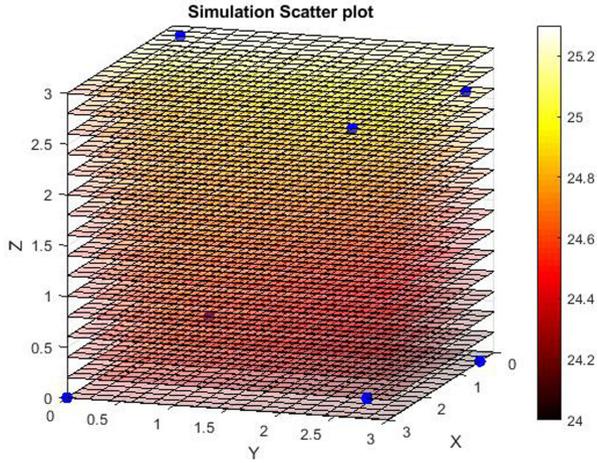


Fig. 2. SIB 2.0 simulation scatter plot

from $k = 7$ and $k = 8$ is small. In addition, our result is better than all results in Fig. 3 including those of $k = 7$ and $k = 8$. It is the advantage of inserting the best particle from the first step to the initial stage of the second step.

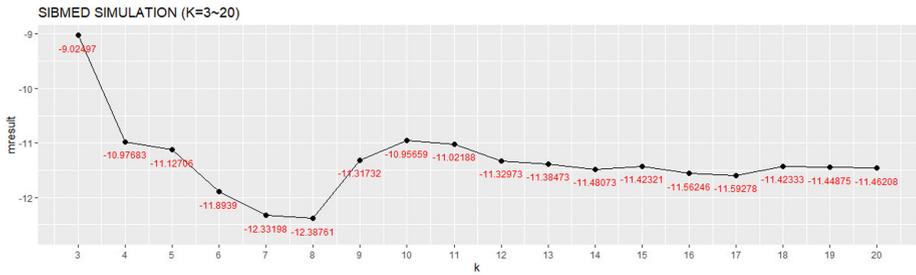


Fig. 3. SIB 1.0 simulation result plot

5 Discussion and Conclusion

SIB 1.0 is widely known as an efficient nature-inspired metaheuristic method for optimization problems in discrete and continuous domain, but its use is hindered by its pre-defined particle size. SIB 2.0 allows particle size to be changed during the optimization procedure, but it also greatly expands the number of possible outcomes. Thus it needs to spend longer time to obtain the results as good as the results obtained by SIB 1.0 when the particle size is known. Therefore, the main contribution of this paper is to propose a two-step method that combines SIB 1.0 and SIB 2.0 and takes their advantages into account. We treat SIB 2.0 as a preliminary study to obtain the optimal particle size, then we treat SIB 1.0 as a follow-up experiment and conduct it under that optimal particle size. We demonstrate the use of this two-step method in the search of optimal minimum energy design. The result outperforms both results from SIB 1.0 and SIB 2.0 and the method is comparably efficient to conducting multiple SIB 1.0 at various sizes.

There are many potential applications to this two-step methods. For example, it can be used to determine the sensor allocations in a given space to monitor environmental factors (temperature, humidity, gas concentration, etc.). It is better to achieve the same performance by using small number of sensors to reduce costs. Another example is to determine a set of biomarkers that have significant impact to the response. The exact number of biomarker in this significant set is not known in prior, but the fewer the biomarkers, the lower the experimental costs to detect the quantities of these biomarkers from the instruments.

References

1. Mirjalili, S.: The art lion optimizer. *Adv. Eng. Softw.* **83**, 80–98 (2015)
2. Ab Wahab, M.N., Nefti-Meziani, S., Atyabi, A.: A comprehensive review of swarm optimization algorithms. *PLoS ONE* **10**(5), e0122827 (2015)
3. Goldberg, D.: *Genetic Algorithms in Optimization. Search and Machine Learning*. Addison Wesley, New York (2003)
4. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **39**(3), 459–471 (2007)
5. Kennedy, J.: Particle swarm optimization. In: Sammut, C., Webb, G.I. (eds.) *Encyclopedia of Machine Learning*, /textbf10, pp. 760–766. Springer, Berlin (2010)
6. Fister Jr., I., Yang, X.S., Fister, I., Brest, J., Fister, D.: arXiv preprint [arXiv:1307.4186](https://arxiv.org/abs/1307.4186) (2013)
7. Phoa, F.K.H.: A Swarm Intelligence Based (SIB) method for optimization in designs of experiments. *Nat. Comput.* **16**(4), 597–605 (2016). <https://doi.org/10.1007/s11047-016-9555-4>
8. Phoa, F.K.H., Lin, Y.-L., Wang, T.-C.: Using swarm intelligence to search for circulant partial Hadamard matrices. In: Tan, Y., Shi, Y., Coello, C.A.C. (eds.) *ICSI 2014. LNCS*, vol. 8794, pp. 158–164. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11857-4_18

9. Phoa, F.K.H., Chen, R.B., Wang, W.C., Wong, W.K.: Optimizing two-level super-saturated designs using swarm intelligence techniques. *Technometrics* **58**(1), 43–49 (2016)
10. Phoa, F.K.H., Chang, L.L.N.: A multi-objective implementation in swarm intelligence and its applications in designs of computer experiments. In: *Proceedings of 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD) 2016*, pp. 253–258. IEEE (2016)
11. Lin, F.P.C., Phoa, F.K.H.: An efficient construction of confidence regions via swarm intelligence and its application in target localization. *IEEE Access* **6**, 8610–8618 (2017)
12. Hsu, T.-C., Phoa, F.K.H.: A smart initialization on the swarm intelligence based method for efficient search of optimal minimum energy design. In: Tan, Y., Shi, Y., Tang, Q. (eds.) *ICSI 2018*. LNCS, vol. 10941, pp. 78–87. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93815-8_9
13. Joseph, V.R., Dasgupta, T., Tuo, R., Wu, C.F.J.: Sequential exploration of complex surfaces using minimum energy designs. *Technometrics* **57**(1), 64–74 (2015)
14. Phoa, F.K.H., Wang, T.C., Chang, L.L.N.: An enhanced version of the swarm intelligence based method (SIB 2.0). In Review