INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2009; **77**:1501–1534
Published online 11 December 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2451

Accelerating adaptive trade-off model using shrinking space technique for constrained evolutionary optimization

Yong Wang[1,*,†], Zixing Cai[1] and Yuren Zhou[2]

[1]*School of Information Science and Engineering, Central South University, Changsha 410083, People's Republic of China*
[2]*School of Computer Science and Engineering, South China University of Technology, Guangzhou 516040, People's Republic of China*

SUMMARY

Adaptive trade-off model (ATM) is a constraint-handling mechanism proposed recently. The main advantages of this model are its simplicity and adaptation. Moreover, it can be easily embedded into evolutionary algorithms for solving constrained optimization problems. This paper proposes a novel method for constrained optimization, which aims at accelerating the ATM using shrinking space technique. Eighteen benchmark test functions and five engineering design problems are used to test the performance of the method proposed. Experimental results suggest that combining the ATM with the shrinking space technique is very beneficial. The method proposed can promptly converge to competitive results without loss of the quality and the precision of the final results. Performance comparisons with some other state-of-the-art approaches from the literature are also presented. Copyright © 2008 John Wiley & Sons, Ltd.

Received 27 March 2008; Revised 11 July 2008; Accepted 16 July 2008

KEY WORDS: constrained optimization problem; constraint-handling technique; adaptive trade-off model; shrinking space technique

1. INTRODUCTION

Considering a general constrained optimization problem (COP) formulated as minimizing $f(\mathbf{x})$, subject to

$$g_j(\mathbf{x}) \leqslant 0, \quad j=1,\ldots,l \tag{1}$$

$$h_j(\mathbf{x}) = 0, \quad j=l+1,\ldots,m \tag{2}$$

*Correspondence to: Yong Wang, School of Information Science and Engineering, Central South University, Changsha 410083, People's Republic of China.
†E-mail: ywang@csu.edu.cn, wangyong1226@gmail.com

Contract/grant sponsor: National Basic Scientific Research Founds; contract/grant number: A1420060159
Contract/grant sponsor: National Natural Science Foundation of China; contract/grant numbers: 60673062, 60805027

where $\mathbf{x}\in\Omega\subseteq S$, $\mathbf{x}=(x_1,x_2,\ldots,x_n)$ is the vector of decision variables to be optimized, each variable x_i is bounded by $[l_i,u_i]$, Ω is the feasible region, S is an n-dimensional rectangular space in $\Re^n$ defined by the parametric constraints: $l_i\leqslant x_i\leqslant u_i(1\leqslant i\leqslant n)$, and the functions $g_j(\mathbf{x})$ and $h_j(\mathbf{x})$ are the inequality and equality constraints, respectively.

For an inequality constraint that satisfies $g_j(\mathbf{x})=0(j\in\{1,\ldots,l\})$ at any point $\mathbf{x}\in\Omega$, it is considered *active* at $\mathbf{x}$. All equality constraints $h_j(\mathbf{x})(j=l+1,\ldots,m)$ are considered *active* at all points of Ω.

In general, COPs are difficult to solve due to the presence of constraints. Evolutionary algorithms (EAs) have been widely applied to solve COPs during the last decade [1, 2]. However, it is necessary to note that EAs are unconstrained search mechanisms; this has motivated the development of a considerable number of constraint-handling techniques. Nowadays, the most commonly used constraint-handling techniques include: methods based on penalty functions, methods based on biasing feasible solutions over infeasible solutions, and methods based on multi-objective optimization techniques [3].

In the penalty function methods, an infeasible solution is penalized based on the degree of constraint violation, which is the sum of the violation of all constraints. Let

$$G_j(\mathbf{x})=\begin{cases}\max\{0,g_j(\mathbf{x})\}, & 1\leqslant j\leqslant l\\ \max\{0,|h_j(\mathbf{x})|-\varepsilon\}, & l+1\leqslant j\leqslant m\end{cases} \tag{3}$$

where ε is a positive tolerance value for equality constraints. Then,

$$G(\mathbf{x})=\sum_{j=1}^{m}G_j(\mathbf{x}) \tag{4}$$

denotes the degree of constraint violation of individual $\mathbf{x}$. Afterward, a penalty term that depends on the degree of constraint violation is added into the objective function to penalize the infeasible solutions. It is noteworthy that the way in which the penalty term is added determines the type of the penalty function. If the penalty term added depends only on the degree of constraint violation, it is called static penalty function, and if the penalty term added depends both on the degree of constraint violation and generation number, it is called dynamic penalty function. Usually, adaptive penalty function methods are very effective for constrained optimization since they can make use of the information obtained during the evolution to adapt their parameters. Farmani and Wright [4] proposed the self-adaptive fitness formulation for solving COPs, which consists of two stages. Tessema and Yen [5] also proposed an adaptive penalty function strategy. In their strategy, a distance value is designed to encourage infeasible individuals with both low fitness values and low constraint violation values. In addition, two penalties are presented by which the algorithm switches between finding more feasible individuals and searching the whole search space for the optimum based on the feasible ratio of the population.

Some methods based on biasing feasible solutions over infeasible solutions have been described. Deb [6] proposed the feasibility-based criteria to compare pair-wise individuals: (1) any feasible solution is preferred over any infeasible solution; (2) between two feasible solutions, the one having a better objective function value is preferred; and (3) between two infeasible solutions, the one having a smaller degree of constraint violation is preferred. Furthermore, a niching method along with a controlled mutation operator is introduced to maintain the diversity of the population. Runarsson and Yao [7] proposed the stochastic ranking method called SR. This method adopts the following criteria to compare pair-wise individuals: (1) if both individuals are feasible, the one with a better objective function value wins, else (2) if a uniformly generated random number u between 0 and 1 is less than p_f, the one with a better objective function value wins, otherwise, the one with a smaller degree of constraint violation wins. Empirical evidence demonstrates that $p_f = 0.45$ is a suitable choice for this method. Based on the feasibility-based comparison criteria, Mezura-Montes and Coello Coello [8] used a diversity mechanism to allow infeasible solutions close to the boundaries of the feasible region to remain in the next population. This approach is called SMES. Takahama and Sakai [9] proposed the α constrained method, in which satisfaction level of constraints and α level comparisons are introduced. The α constrained method can convert an algorithm for unconstrained optimization problems into an algorithm for COPs by replacing ordinary comparison with the α level comparison.

The main ideas of the methods based on multi-objective optimization techniques are twofold: (1) transform the COPs into unconstrained multi-objective optimization problems (MOPs); and (2) utilize multi-objective optimization techniques to cope with the converted problems. In these kind of methods, constraints can be treated as one or more objectives. If the constraints are regarded as one objective, then the original COPs are converted into bi-objective optimization problems. In general, one objective is the objective function $f(\mathbf{x})$ and the other is the degree of constraint violation $G(\mathbf{x})$. In addition, if the constraints are considered as more objectives, then the original COPs are usually transformed into MOPs problems $(f(\mathbf{x}), G_1(\mathbf{x}), \ldots, G_m(\mathbf{x}))$ that have $m+1$ objectives.

Since some methods based on multi-objective optimization techniques will be introduced, the description of MOP and the basic definitions related to multi-objective optimization are first given. Assuming that a MOP has the following form:

$$\text{minimize } \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x}))$$

where k is the number of objective functions, $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in X \subset \Re^n$ is the vector of decision variables and X is the decision space.

Definition 1 (*Pareto Dominance*)
A vector $\mathbf{u} = (u_1, \ldots, u_k)$ is said to *Pareto dominate* another vector $\mathbf{v} = (v_1, \ldots, v_k)$, denoted as $\mathbf{u} \prec \mathbf{v}$, if

$$\forall i \in \{1, \ldots, k\}, u_i \leqslant v_i \quad \text{and} \quad \exists j \in \{1, \ldots, k\}, \quad u_j < v_j \tag{5}$$

Definition 2 (*Pareto Optimality*)
$\mathbf{x}_u \in X$ is said to be *Pareto optimal* in X, if and only if $\nexists \mathbf{x}_v \in X$, $\mathbf{f}(\mathbf{x}_v) \prec \mathbf{f}(\mathbf{x}_u)$.

Definition 3 (*Pareto Optimal Set*)
For a given MOP $\mathbf{f}(\mathbf{x})$, the *Pareto optimal set*, denoted as ρ^*, is defined as

$$\rho^* = \{\mathbf{x}_u \in X \mid \nexists \mathbf{x}_v \in X, \mathbf{f}(\mathbf{x}_v) \prec \mathbf{f}(\mathbf{x}_u)\} \tag{6}$$

The vectors included in the Pareto optimal set are called non-dominated individuals.

Venkatraman and Yen [3] proposed a generic, two-phase framework for solving COPs. In the first phase, the COP is regarded as a constrained satisfaction problem by completely disregarding the objective function. In the second phase, the objective function and the constraints are treated as bi-objective optimization problems and optimized simultaneously. Ray *et al.* [10] introduced a method where Pareto ranking is performed separately on individual constraints so that there can be cooperative mating between solutions that are good in objective function and solutions that are good in constraint violations. Kurpati *et al.* [11] proposed four constraint-handling improvements. In the last constraint-handling improvement, both the degree and the number of constraint violations are taken into account when calculating the fitness of all the infeasible solutions. Zhou *et al.* [12] converted a COP into a bi-objective optimization problem. In this method, Pareto strength is defined for each individual in the population based on the Pareto dominance. If the Pareto strength of an individual is large, it means that the number of individuals in the population Pareto dominated by this individual is also large and vice versa. Cai and Wang [13] proposed an approach, in which only one of the non-dominated individuals identified from the offspring population is selected and used to replace one individual randomly chosen from the parent population and Pareto dominated by it. An infeasible solution archiving and replacement mechanism is also introduced in this method. Many methods have been implemented as extensions of multi-objective EAs, for instance the method [14] based on Fonseca and Fleming's Pareto ranking process [15], the method [16] based on population-based multi-objective technique such as vector evaluated genetic algorithm [17], the method [18] based on the niched-Pareto genetic algorithm [19], and the method (abbreviated as IS-PAES) [20] based on the Pareto archived evolutionary strategy (ES) [21]. Mezura-Montes and Coello Coello [22] presented an extensive survey of the constraint-handling technique based on evolutionary multi-objective optimization concepts.

More recently, the authors proposed an adaptive trade-off model (ATM) [23] for constrained optimization. In essence, ATM belongs to the category of penalty function methods. The main characteristics of this model are its simplicity and adaptation. Compared with SR [7] and SMES [8] that are two representatives of the state-of-the-art methods in constrained evolutionary optimization, this model has very competitive performance. In order to further enhance the performance of this model, the authors have proposed a hybrid version [24].

The shrinking space technique is proposed by Aguirre *et al.* [20]. This technique focuses the search effort onto a specific area of the feasible region by shrinking the constrained search space. The main emphasis of this paper is to demonstrate how this technique can accelerate the ATM. By incorporating this technique with ATM, a new approach, namely, accelerating the adaptive trade-off model (denoted as AATM) is obtained. Experimental results on 18 benchmark test functions and five engineering design problems validate that integrating ATM with the shrinking space technique is very beneficial. The proposed approach can efficiently achieve competitive results without sacrificing the quality and the precision of the final results.

The remainder of the paper is organized as follows. In Section 2, the general ideas of ATM, the shrinking space technique, and how to speed up ATM by the shrinking space technique are introduced. Section 3 presents the experimental results. Moreover, the proposed method (AATM) is compared with ATM and IS-PAES [20] for standard benchmarks and with some recent methods for engineering design problems. The effectiveness of some mechanisms and the effect of some parameters adopted by AATM have been verified in Section 4. Section 5 concludes this paper.

2. GENERAL IDEAS

2.1. Main idea of the ATM

According to [23], the main challenge of constrained optimization is to determine trade-off between objective function and constraint violation, since the handling of constraint and the optimization of objective function should be conducted simultaneously.

DOI: 10.1002/nme

ATM includes three main stages, i.e. the infeasible stage, the semi-feasible stage, and the feasible stage. Let fp denote the proportion of feasible solutions in the current population. The infeasible stage means that the population contains only infeasible individuals (i.e. $fp=0$), the semi-feasible stage denotes that the population consists of a combination of feasible and infeasible individuals (i.e. $0<fp<1$), and the feasible stage signifies that the population is composed of feasible individuals only (i.e. $fp=1$). Over the course of the evolution, different trade-off schemes at different stages are designed to obtain an appropriate trade-off between objective function and constraint violation. It is necessary to note that in ATM, COP is converted into a bi-objective optimization problem $\mathbf{f}(\mathbf{x})=(f(\mathbf{x}), G(\mathbf{x}))$ in which $G(\mathbf{x})$ has been specified in Equation (4).

In the infeasible stage, a hierarchical non-dominated individual selection scheme is proposed. This scheme attempts to guide the population toward feasibility from various directions and is executed as follows: first, only the first half of non-dominated individuals are selected and stored into the offspring population, after ranking non-dominated individuals based on their constraint violations in ascending order. Note that the selected individuals are deleted from the parent population subsequently. Next, half of the non-dominated individuals with less constraint violations in the remaining population are also stored into the offspring population, and then eliminated from the parent population. This process continues until the number of the individuals archived reaches the size of the offspring population. In this stage, since non-dominated individuals should be identified from the population, the comparisons among the individuals in the population are based on Pareto dominance.

With respect to the semi-feasible stage, an adaptive fitness transformation scheme is proposed, which tends to make some important feasible and infeasible solutions remain in the population. First, the population is divided into the feasible group Z_1 and the infeasible group Z_2. The best feasible solution $\mathbf{x}_{\text{best}}$ and the worst feasible solution $\mathbf{x}_{\text{worst}}$ are found from the feasible group. Then, the objective function is transformed into the following form:

$$f'(\mathbf{x}_i)=\begin{cases} f(\mathbf{x}_i), & i\in Z_1 \\ \max\{\varphi * f(\mathbf{x}_{\text{best}})+(1-\varphi)*f(\mathbf{x}_{\text{worst}}), f(\mathbf{x}_i)\}, & i\in Z_2 \end{cases} \tag{7}$$

where φ denotes the feasibility proportion of the last population. Afterward, each objective function value is normalized as follows:

$$f_{\text{nor}}(\mathbf{x}_i)=\frac{f'(\mathbf{x}_i)-\min_{j\in Z_1\cup Z_2} f'(\mathbf{x}_j)}{\max_{j\in Z_1\cup Z_2} f'(\mathbf{x}_j)-\min_{j\in Z_1\cup Z_2} f'(\mathbf{x}_j)}, \quad i\in Z_1\cup Z_2 \tag{8}$$

Similarly, the constraint violations are normalized according to

$$G_{\text{nor}}(\mathbf{x}_i)=\begin{cases} 0, & i\in Z_1 \\ \dfrac{G(\mathbf{x}_i)-\min_{j\in Z_2} G(\mathbf{x}_j)}{\max_{j\in Z_2} G(\mathbf{x}_j)-\min_{j\in Z_2} G(\mathbf{x}_j)}, & i\in Z_2 \end{cases} \tag{9}$$

A final fitness function is obtained by adding the normalized objective function and constraint violations together as

$$f_{\text{final}}(\mathbf{x}_i)=f_{\text{nor}}(\mathbf{x}_i)+G_{\text{nor}}(\mathbf{x}_i), \quad i\in Z_1\cup Z_2 \tag{10}$$

Afterward, Equation (10) will be used to rank the individuals in the population and identify the best ones.

DOI: 10.1002/nme

Since in the feasible stage the constraint violation is equal to zero for each individual, only the objective function value needs to be considered. Thus, the comparison and selection of individuals are based only on the objective function value in this stage.

It is noteworthy that (μ, λ)-ES is used as the search algorithm in [23]. For more details about ATM, see Wang *et al.* [23].

2.2. *Main idea of the shrinking space technique*

The shrinking space technique is the most important part of IS-PAES [20], the task of which is the reduction of the search space. In the shrinking space technique, global information carried by the individuals surrounding the feasible region is used to concentrate the search effort on smaller space as the evolutionary process takes place. Therefore, the search space being explored becomes increasingly smaller over time.

The procedure of the shrinking space technique is implemented as follows: first, the best individuals are found from the *file* (note that in IS-PAES, the individuals are stored into an external memory called *file*) and stored into a list. The size of the list is set to 15% of *maxsize*. Then, the lower bound $\underline{x}_{pob}$ and the upper bound $\overline{x}_{pob}$ of the decision variables represented by those individuals in the list are calculated. Afterward, the function $\text{trim}(\underline{x}_{pob}, \overline{x}_{pob})$ is performed to shrink the search space around the potential solutions enclosed in the hyper-volume defined by the vectors $\underline{x}_{pob}$ and $\overline{x}_{pob}$, and to determine the new boundaries for the decision variables. In the function $\text{trim}(\underline{x}_{pob}, \overline{x}_{pob})$, the search interval of each decision variable $x_i (i \in \{1, \ldots, n\})$ must be reduced by $(1-\beta)100\%$ compared with its previous value. In addition, a slack operator is proposed to prevent against the fast decreasing rate of the search space. The slack operator exploits parameter *slack* to accomplish this purpose. The parameter *slack* for each dimension of the decision variable is set as $\text{slack}_i = \delta(\overline{x}_{pob,i} - \underline{x}_{pob,i})$, where $i \in \{1, \ldots, n\}$. Based on the above operations, the new search space is determined. Since IS-PAES adopts ES as the search algorithm, the step-size σ should be re-started through

$$\sigma_i = (u_i^t - l_i^t)/\sqrt{n}, \quad i \in \{1, \ldots, n\} \tag{11}$$

The shrinking space technique is executed at every T number of generations. It is necessary to note that u_i^t and l_i^t denote the actual upper and lower bounds of the ith decision variable at the current generation t, respectively, and $\overline{x}_{pob}$ and $\underline{x}_{pob}$ denote the upper and lower bounds of the ith decision variable in the list, respectively. For more details about the shrinking space technique, see Aguirre *et al.* [20].

2.3. *Accelerating the ATM by the shrinking space technique*

After introducing ATM and the shrinking space technique, it is intended to accelerate ATM by the shrinking space technique for constrained optimization, since taking advantage of the shrinking space technique the search space will be shrunk constantly during the evolution and the search will focus only on the current reduced space.

As already stated, in ATM (μ, λ)-ES is used as the search algorithm. Therefore, in AATM (μ, λ)-ES is also applied for the same purpose. The main procedure of AATM is presented in Figure 1. Compared with ATM, AATM makes use of the shrinking space technique to enhance the velocity of convergence, which is the main difference between these two methods.

DOI: 10.1002/nme

Initialize:
  t=0; /* t denotes the current generation number */
  $\boldsymbol{\sigma}_k = \breve{\sigma}(\mathbf{u}^t - \mathbf{l}^t)/\sqrt{n}$, $k = 1, \cdots, \mu$; /* $\mathbf{u}^t = (u_1^t, \cdots, u_n^t)$ and $\mathbf{l}^t = (l_1^t, \cdots, l_n^t)$ are the upper and lower bounds of the decision variables, respectively, and n is the number of the decision variables. Note that $\mathbf{u}^0 = \mathbf{u} = (u_1, \cdots, u_n)$ and $\mathbf{l}^0 = \mathbf{l} = (l_1, \cdots, l_n)$ */
  $\mathbf{x}_k = \mathbf{l}^t + (\mathbf{u}^t - \mathbf{l}^t)\mathbf{U}(0,1)$, $k = 1, \cdots, \mu$;
  evaluate: $f(\mathbf{x}_k)$ and $G(\mathbf{x}_k)$, $k = 1, \cdots, \mu$;
while termination criteria not satisfied **do**
  $\varphi = num\_1/\mu$, where num_1 denotes the number of feasible solutions in the current population of size μ;
  for $k = 1$ to λ **do** /* replication */
    $i \leftarrow \text{mod}(k-1, \mu) + 1$; /* cycle through the best μ points */
    $\sigma'_{i,j} = (\sigma_{i,j} + \sigma_{k_j,j})/2$, $k_j \in \{1, \cdots, \mu\}$ and $j = 1, \cdots, n$; /* k_j is a random number in $\{1, \cdots, \mu\}$ */
    $\sigma'_{k,j} \leftarrow \sigma'_{i,j} \exp(\tau' \mathrm{N}(0,1) + \tau \mathrm{N}_j(0,1))$, where $\tau = (\sqrt{2\sqrt{n}})^{-1}$, $\tau' = (\sqrt{2n})^{-1}$ and $j = 1, \cdots, n$;
    $\mathbf{x}'_k \leftarrow \mathbf{x}_i + \boldsymbol{\sigma}'_k \mathbf{N}(0,1)$; /* retry if out of bounds */
  end for
  evaluate: $f(\mathbf{x}'_k)$ and $G(\mathbf{x}'_k)$, $k = 1, \cdots, \lambda$;
  $fp = num\_2/\lambda$, where num_2 denotes the number of feasible solutions in the current population of size λ;
  select the best μ individuals from the λ points based on the adaptive trade-off model, i.e. $(\mathbf{x}_i, \boldsymbol{\sigma}_i) \leftarrow (\mathbf{x}'_{i;\lambda}, \boldsymbol{\sigma}'_{i;\lambda})$, $i = 1, \cdots, \mu$;
  /* shrink the search space */
  if mod(t, T)=0 **then** /* the parameter T signifies that the shrinking space technique is performed every T generations */
    for $i = 1$ to n **do**
⇒   **if** $(u_i^t - l_i^t) > \alpha_i$ **then** /* the first difference between AATM and IS-PAES */
        $width\_pob_i = \overline{x}_{pob,i} - \underline{x}_{pob,i}$;
        $width_i^t = u_i^t - l_i^t$;
        $deltaMin_i = (\beta \times width_i^t - width\_pob_i)/2$;
⇒ $delta_i = \max(0, deltaMin_i)$; /* the second difference between AATM and IS-PAES */
        $u_i^{t+1} = \overline{x}_{pob,i} + delta_i$; $l_i^{t+1} = \underline{x}_{pob,i} - delta_i$;
        if $u_i^{t+1} > u_i^0$ **then**
          $l_i^{t+1} -= u_i^{t+1} - u_i^0$; $u_i^{t+1} = u_i^0$;
        end if
        if $l_i^{t+1} < l_i^0$ **then**
          $u_i^{t+1} += l_i^0 - l_i^{t+1}$; $l_i^{t+1} = l_i^0$;
        end if
⇒   **for** $j = 1$ to μ **do** /* the third difference between AATM and IS-PAES */
⇒ $\sigma_{j,i} = \breve{\sigma}(u_i^{t+1} - l_i^{t+1})/\sqrt{n}$;
⇒   **end for**
      end if
    end for
  else
    $\mathbf{u}^{t+1} = \mathbf{u}^t$; $\mathbf{l}^{t+1} = \mathbf{l}^t$;
  end if
  $t = t + 1$;
end while

Figure 1. Pseudo-code of the proposed AATM.

DOI: 10.1002/nme

In addition, there are five main differences between AATM and IS-PAES, which are summarized as follows:

- AATM proposes a shrinking termination scheme to avoid the search space being reduced too much, i.e. once the search space is shrunk to a certain degree, the reduction should be halted. This scheme is shown in step 16 of Figure 1 and executed as follows: the search interval of the ith dimension of the decision variables is shrunk continuously until the search interval is smaller than or equal to a threshold α_i. The reason why the shrinking termination scheme is applied is that if the current search space is very small, in one respect the population may easily deviate from the global optimum if the global optimum lies within the feasible region, which has an influence on the precision of the resulting solution. On the other hand, if the global optimum is located on the boundary of the feasible region, the population might frequently converge to infeasible solution because of the too small search space.
- The slack operator proposed by IS-PAES has been revised in this paper so as to ensure that the size of the reduced search space at each generation is not smaller than that of the region defined by the potential individuals. This new operator has been executed in step 20 of Figure 1, which makes AATM easier to implement and with less parameters than IS-PAES.
- In AATM the step-size of the ith decision variable for ES is modified according to the new search interval if the search interval of the ith decision variable does not reach the threshold defined in step 16 of Figure 1. Meanwhile, if the search interval of the ith decision variable is smaller than or equal to the threshold α_i, the corresponding step-size is unchanged. Whereas, in IS-PAES the step-size of all dimensions of the decision variables should be adjusted after the new search space is determined. The third difference is shown in steps 28–30 of Figure 1.
- In IS-PAES, only the best 15% individuals are extracted from the *file* and used to establish the referenced region when the shrinking space technique is implemented, so that the search space is reduced around this potential region. However, in AATM the population of size μ is used for this aim directly, which again makes AATM simpler and with less parameters in the evolutionary process compared with IS-PAES.
- Finally, in AATM and IS-PAES the parameter settings are distinct for the shrinking space technique, since AATM and IS-PAES combine this technique with different search algorithms.

3. EXPERIMENTAL STUDY

3.1. Benchmark test functions

Eighteen well-known benchmark test functions are optimized to evaluate the performance of AATM. The first 12 test functions and the remaining test functions are collected by Koziel and Michalewicz [25] and by Liang *et al.* [26], respectively. These test functions are depicted in the Appendix. The main characteristics of these test functions are reported in Table I. From Table I, it is obvious that the test functions include different types of objective functions (linear, non-linear, polynomial, quadratic, and cubic) and constraints (linear inequalities, non-linear inequalities, linear equalities, and non-linear equalities). The feasibility ratio ρ is determined experimentally by calculating the percentage of feasible solutions among 1 000 000 randomly generated individuals from the search space. a denotes the number of constraints *active* at the global optimum. Note that, test functions g02, g03, g08, and g12 are maximization problems that are transformed into minimization by applying $-f(\mathbf{x})$, and the others are minimization problems. In addition, only test

DOI: 10.1002/nme

Table I. The main characteristics of eighteen benchmark test functions.

Function	Number of decision variables	Type of objective function	ρ (%)	Number of linear inequality constraints	Number of non-linear inequality constraints	Number of linear equality constraints	Number of non-linear equality constraints	a
g01	13	quadratic	0.0111	9	0	0	0	6
g02	20	non-linear	99.9971	0	2	0	0	1
g03	10	polynomial	0.0000	0	0	0	1	1
g04	5	quadratic	52.1230	0	6	0	0	2
g05	4	cubic	0.0000	2	0	0	3	3
g06	2	cubic	0.0066	0	2	0	0	2
g07	10	quadratic	0.0003	3	5	0	0	6
g08	2	non-linear	0.8560	0	2	0	0	0
g09	7	polynomial	0.5121	0	4	0	0	2
g10	8	linear	0.0010	3	3	0	0	6
g11	2	quadratic	0.0000	0	0	0	1	1
g12	3	quadratic	4.7713	0	9^3	0	0	0
g14	10	non-linear	0.0000	0	0	3	0	3
g15	3	quadratic	0.0000	0	0	1	1	2
g16	5	non-linear	0.0204	4	34	0	0	4
g18	9	quadratic	0.0000	0	13	0	0	6
g19	15	non-linear	33.4761	0	5	0	0	0
g24	2	linear	79.6556	0	2	0	0	2

functions g03, g05, g11, g14, and g15 contain equality constraints. As for test function g12, the feasible region of the search space consists of 729 disjointed spheres.

3.1.1. Experimental results. In this paper, (50, 200)-ES is used as the search algorithm. The parameters of the shrinking space technique in Figure 1 are set as follows: $T=20$, $\alpha_i=(u_i^0-l_i^0)/(20\times 3^{\log_{10}^{(u_i^0-l_i^0)}})$, and $\beta=0.02^{(1/n)}$. The parameter $\breve{\sigma}$ for (50,200)-ES in Figure 1 is equal to 0.5. In addition, the equality constraints have been converted into inequality constraints $|h_j(\mathbf{x})|-\varepsilon<0$, with the tolerance value $\varepsilon=0.00005$. The number of generations is fixed to 600, as a result the number of fitness function evaluations (FFEs) is equal to 120 000. Having selected the parameter values for AATM, 30 independent runs are carried out for each test function in MATLAB.‡

The experimental results of AATM are summarized in Table II. This table shows the 'known' optimal solution for each test function and the 'best', 'median', 'mean', 'worst', and standard deviations of the objective function values achieved by AATM.

As shown in Table II, AATM has the ability to consistently converge to the global optima for nine test functions, i.e. g01, g03, g04, g06, g08, g11, g12, g16, and g24. Although for the rest of the test functions (i.e. g02, g05, g07, g09, g10, g14, g15, g18, and g19), the global optima cannot be found consistently, the 'best' results provided by AATM for test functions g05, g09, g15, and g18 are equal to the optimal values known, in addition, the 'best' results provided by AATM for test functions g02, g07, g10, g14, and g19 are fairly close to the optimal values known.

‡The source code may be obtained from the authors upon request.

DOI: 10.1002/nme

Table II. Statistical results obtained by AATM for eighteen benchmark test functions over 30 independent runs.

		Results of AATM				
Function	Optimal	Best	Median	Mean	Worst	St. dev.
g01	−15.000	−15.000	−15.000	−15.000	−15.000	3.1E−07
g02	−0.803619	−0.803389	−0.792408	−0.791213	−0.767040	8.6E−03
g03	−1.00	−1.00	−1.00	−1.00	−1.00	3.5E−04
g04	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539	1.0E−11
g05	5126.498	5126.498	5126.568	5126.714	5128.824	4.3E−01
g06	−6961.814	−6961.814	−6961.814	−6961.814	−6961.814	7.1E−12
g07	24.306	24.307	24.312	24.317	24.356	1.3E−02
g08	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825	5.8E−18
g09	680.630	680.630	680.632	680.634	680.646	4.5E−03
g10	7049.248	7049.603	7065.614	7077.477	7183.295	3.1E+01
g11	0.75	0.75	0.75	0.75	0.75	3.8E−06
g12	−1.000	−1.000	−1.000	−1.000	−1.000	0.0E+00
g14	−47.763	−47.762	−47.753	−47.750	−47.712	1.0E−02
g15	961.715	961.715	961.715	961.715	961.716	3.0E−04
g16	−1.905155	−1.905155	−1.905155	−1.905155	−1.905155	2.4E−14
g18	−0.866025	−0.866025	−0.866013	−0.865952	−0.864843	2.1E−04
g19	32.656	32.725	32.934	32.952	33.243	1.4E−01
g24	−5.508013	−5.508013	−5.508013	−5.508013	−5.508013	1.8E−15

The distribution of the resulting solutions for these nine test functions is shown in Figure 2. The standard deviations over 30 independent runs for all test functions other than g10 are extremely small and the 'mean' results provided by AATM are very near the 'best' results obtained, which indicates that the performance of AATM is very stable in producing consistent results for these test functions. Note that AATM can provide 100% feasible solutions for all test functions. Furthermore, the computational cost of AATM is relatively low since it only needs 120 000 FFEs for all test functions.

Based on the above discussion, it can be concluded that AATM is an effective and efficient approach for constrained optimization.

3.1.2. Comparison with the ATM and the IS-PAES. In order to verify that the performance of ATM can be further improved by the shrinking space technique, AATM is compared with ATM. In addition, since the similar shrinking space technique is adopted by both AATM and IS-PAES, AATM is also compared with IS-PAES. In ATM, for each test function, 30 independent runs are performed using (50, 300)-ES, and the number of FFEs is equal to 240 000. The parameter $\breve{\sigma}$ in (50, 300)-ES is set to 0.8. This method uses dynamic tolerance value for equality constraints. Besides the shrinking space technique, IS-PAES uses Pareto dominance as the comparison criterion and an adaptive grid to store the solutions found. In this approach, $maxsize = 200$, $T = 1$, $\delta = 0.05$, and $\beta = 0.9^{(1/n)}$. In order to maintain diversity, for every 10 individuals created, a new parent is randomly chosen from the less populated region of the grid. For each test function, 30 independent runs are performed with 350 000 FFEs in IS-PAES.

As shown in Tables III and IV, the performance of AATM is compared with ATM and IS-PAES in detail for the selected performance criteria. The results of ATM and IS-PAES are directly taken

DOI: 10.1002/nme

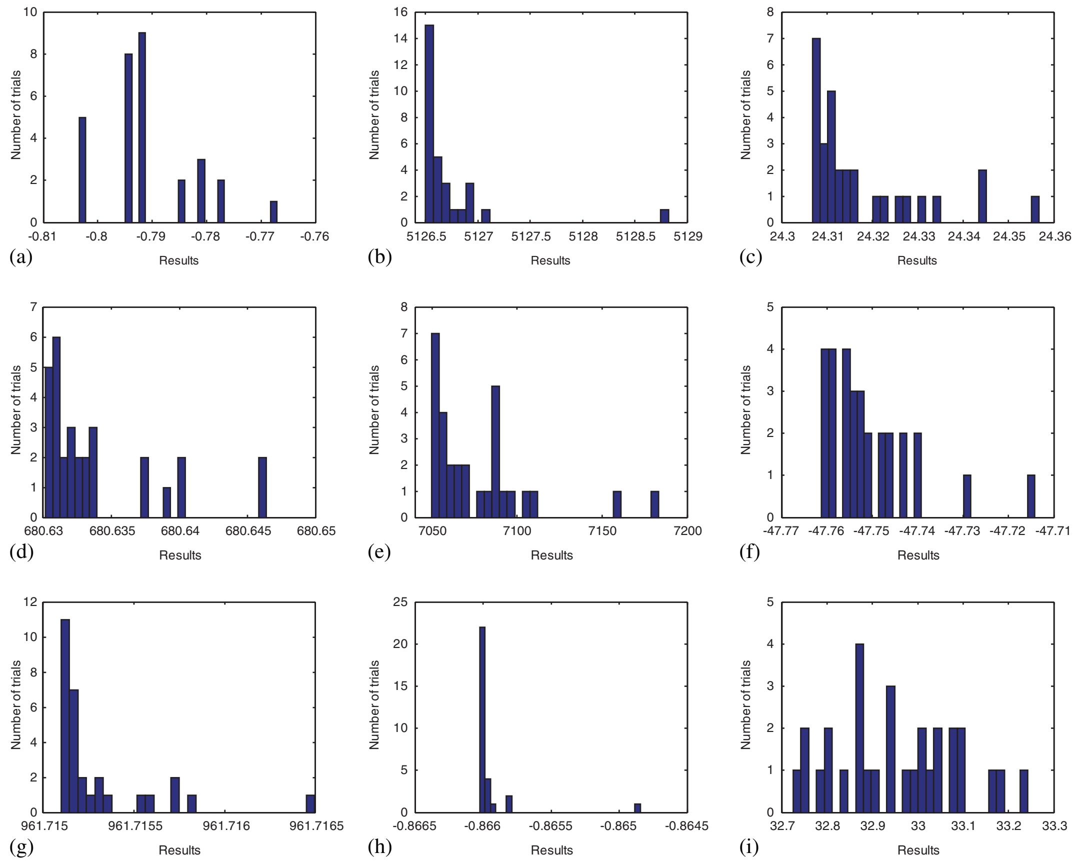

Figure 2. The distribution of the resulting solutions for test functions g02, g05, g07, g09, g10, g14, g15, g18, and g19 over 30 independent runs. (a) Test function g02; (b) Test function g05; (c) Test function g07; (d) Test function g09; (e) Test function g10; (f) Test function g14; (g) Test function g15; (h) Test function g18; and (i) Test function g19.

from [20, 23] to make a fair comparison. Furthermore, ATM is extended to yield more experimental results using the additional test functions adopted in this paper, i.e. test functions g14, g15, g16, g18, g19, and g24.

As can be seen from Table III, both ATM and AATM can consistently converge to the global optima for test functions g01, g03, g04, g06, g08, g11, and g24 in all trials. For test functions g05, g09, and g18, AATM reaches similar 'best' results and better 'mean' and 'worst' results compared with ATM. For test function g15, AATM presents similar 'best' and 'mean' results and better 'worst' result. ATM slightly outperforms AATM for test function g07 in terms of 'best' and 'mean' results; however, AATM finds better 'worst' result. AATM performs much better than ATM for test functions g02, g10, g14, g16, and g19 in terms of all the criteria. For test function g12, one or two out of the 30 runs of ATM are trapped in a local optimum, whereas AATM can consistently find the global optimum. It is noteworthy that in ATM the number of FFEs is 240 000, which is twice as many as that used by AATM, i.e. the computational cost of AATM is only half of ATM.

DOI: 10.1002/nme

Table III. Comparing AATM with respect to ATM on 18 benchmark test functions.

Function	Optimal	Best result		Mean result		Worst result	
		AATM	ATM	AATM	ATM	AATM	ATM
g01	−15.000	−15.000	−15.000	−15.000	−15.000	−15.000	−15.000
g02	−0.803619	−**0.803389**	−0.803388	−**0.791213**	−0.790148	−**0.767040**	−0.756986
g03	−1.00	−1.00	−1.00	−1.00	−1.00	−1.00	−1.00
g04	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539
g05	5126.498	5126.498	5126.498	**5126.714**	5127.648*	**5128.824**	5135.256
g06	−6961.814	−6961.814	−6961.814	−6961.814	−6961.814	−6961.814	−6961.814
g07	24.306	24.307	**24.306**	24.317	**24.316**	24.356	24.359
g08	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825
g09	680.630	680.630	680.630	**680.634**	680.639*	**680.646**	680.673
g10	7049.248	**7049.603**	7052.253	**7077.477**	7250.437*	**7183.295**	7560.224
g11	0.75	0.75	0.75	0.75	0.75	0.75	0.75
g12	−1.000	−1.000	−1.000	−1.000	−1.000	−**1.000**	−0.994
g14	−47.763	−**47.762**	−47.760	−**47.750**	−47.722*	−**47.712**	−47.618
g15	961.715	961.715	961.715	961.715	961.715	**961.716**	961.718
g16	−1.905155	−**1.905155**	−1.904391	−**1.905155**	−1.902950*	−**1.905155**	−1.900724
g18	−0.866025	−0.866025	−0.866025	−**0.865952**	−0.846860*	−**0.864843**	−0.674779
g19	32.656	**32.725**	32.915	**32.952**	33.416*	**33.243**	34.562
g24	−5.508013	−5.508013	−5.508013	−5.508013	−5.508013	−5.508013	−5.508013

A result in boldface indicates a better result between the two compared methods is reached. *indicates the t value is significant at a 0.05 level of significance by two-tailed t-test.

Table IV. Comparing AATM with respect to IS-PAES on 10 benchmark test functions.

Function	Optimal	Best result		Mean result		Worst result	
		AATM	IS-PAES	AATM	IS-PAES	AATM	IS-PAES
g01	−15.000	−15.000	−15.000	**−15.000**	−14.494*	**−15.000**	−12.446
g02	−0.803619	**−0.803389**	−0.803376	−0.791213	**−0.793281**	−0.767040	**−0.768291**
g03	−1.00	−1.00	−1.00	−1.00	−1.00	−1.00	−1.00
g04	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539	−30665.539
g06	−6961.814	−6961.814	−6961.814	**−6961.814**	−6961.813*	**−6961.814**	−6961.810
g07	24.306	**24.307**	24.338	**24.317**	24.527*	**24.356**	24.995
g08	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825	−0.095825
g09	680.630	−680.630	680.630	−680.634*	**680.631**	680.646	**680.634**
g10	7049.248	**7049.603**	7062.019	**7077.477**	7342.944*	**7183.295**	7588.054
g11	0.75	0.75	0.75	0.75	0.75	0.75	0.75

A result in boldface indicates a better result among the two compared methods is reached. *indicates the t value is significant at a 0.05 level of significance by two-tailed t-test.

DOI: 10.1002/nme

From the above discussion, one can conclude that AATM is superior to, or at least competitive with ATM in terms of the selected performance criteria, and that AATM is more efficient than ATM, which signifies that the shrinking space technique has the capability to significantly speed up ATM without loss of the quality and the precision of the resulting solutions.

Since there are 10 same test functions used by AATM and IS-PAES, a comparison between AATM and IS-PAES is carried out on these test functions. As shown in Table IV, both IS-PAES and AATM can consistently converge to the global optima for test functions g03, g04, g08, and g11 in all trials. For test function g02, IS-PAES outperforms AATM in terms of 'mean' and 'worst' results, whereas AATM performs better than IS-PAES with regard to 'best' result. In addition, AATM performs much better than IS-PAES for test functions g07 and g10 in terms of all the criteria. IS-PAES finds better 'mean' and 'worst' results for test function g09; however, AATM finds better 'mean' and 'worst' results for test functions g01 and g06. More importantly, for test functions g01 and g06, AATM can converge to the global optima in all trials, but IS-PAES cannot. In addition, it is necessary to note that the number of FFEs used by IS-PAES is equal to 350 000, which is nearly three times more than that of AATM. Based on the above comparison, it is clear that AATM has very competitive performance with respect to IS-PAES in terms of the quality of the results and is more efficient than IS-PAES.

Since the above performance indicators (i.e. the 'best', 'mean', 'worst' and standard deviation of the resulting solutions) cannot provide precise statistical information, t-test is employed. The t-test is a statistical significance test, which assesses whether the mean of two groups is statistically different from each other or not. t-test is performed between AATM and each of ATM and IS-PAES. As shown in Tables III and IV, AATM significantly outperforms ATM on seven test functions (i.e. test functions g05, g09, g10, g14, g16, g18, and g19) out of the entire test functions, whereas ATM cannot perform significantly better than AATM on even one test function. In addition, AATM is remarkably superior to IS-PAES in terms of test functions g01, g06, g07, and g10. Meanwhile, IS-PAES can surpass AATM in one test function, i.e. test function g09.

3.2. Engineering design problems

In order to further study the performance of AATM, five engineering design problems are solved. Note that, this time 50 independent runs are performed. The parameter settings are the same as those used by the previous experiments for 18 benchmark test functions other than the number of FFEs, which is set based on the problems at hand.

3.2.1. The first four engineering design problems. The first four engineering design problems are chosen from [27], the mathematical description of which has been presented in the Appendix. For more details about them, see [27]. The results obtained by AATM are compared with the society and civilization method (denoted as SCM) [27]. SCM is proposed by Ray and Liew, which makes use of the intra- and inter-society interactions within a society and civilization model to solve COPs. In this method, a society corresponds to a set of points in the solution space whereas a civilization is a set of all such societies. During the evolution, the intra-society information exchange between the leader and the other individuals in the society leads to the migration of a point toward a better performing point. On the other hand, the inter-society information exchange of leaders helps the better performing societies to expand and flourish. For each problem, 50 independent runs are performed in SCM. The size of the civilization is set to $10n$, where n is the

DOI: 10.1002/nme

Table V. Comparison between AATM and SCM on four engineering design problems over 50 independent runs.

Problem	Method	Best	Mean	Worst	St. dev.	Number of FFEs
Welded-beam	AATM	2.3823262	2.3869762	2.3915924	2.2E−03	30 000
design	SCM	2.3854347	3.0025883	6.3996785	9.6E−01	33 095
Spring design	AATM	0.01266826198	0.012708075	0.012861375	4.5E−05	25 000
	SCM	0.01266924934	0.012922669	0.016717272	5.9E−04	25 167
Speed reducer	AATM	2994.516778	2994.585417	2994.659797	3.3E−02	40 000
design	SCM	2994.744241	3001.758264	3009.964736	4.0E+00	54 456
Three-bar truss	AATM	263.8958435	263.8966	263.90041	1.1E−03	17 000
design	SCM	263.8958466	263.9033	263.96975	1.3E−02	17 610

Table VI. Comparison of results for welded beam design.

	AATM	SCM
x_1	0.2441065868	0.2444382760
x_2	6.2209036335	6.2379672340
x_3	8.2981612295	8.2885761430
x_4	0.2443822319	0.2445661820
Best	2.38232623	2.3854347

number of decision variables. In addition, SCM uses different number of FFEs based upon the difficulty of the problems at hand.

Table V summarizes the experimental results obtained by AATM over 50 independent runs. The number of FFEs for these four engineering design problems are 30 000, 25 000, 40 000, and 17 000, respectively. From Table V, the results provided by AATM for these four problems are well under a low computational effort in terms of the selected performance metrics. Feasible solutions are consistently found for all test problems. Moreover, the standard deviations are relatively small, which is an important characteristic for the application of the method to the solution of problems in the real world.

The performance comparison between AATM and SCM is also shown in Table V, where the results of SCM are obtained from [27]. From Table V, AATM apparently outperforms SCM on the four problems in terms of all the criteria. In particular, for speed reducer design problem, even the 'worst' result found by AATM is better than the 'best' result found by SCM. Moreover, for all problems AATM uses smaller number of FFEs than SCM.

The best solutions for the four problems obtained by AATM and SCM are listed in Tables VI–IX. The constraints based on the best results derived from AATM for the four problems are: $[-0.681300\ \ -49.976182\ \ -0.000276\ \ -0.234280\ \ -4.337121]$, $[-0.000162\ \ -0.000042\ \ -4.058572\ \ -0.725664]$, $[-0.073924\ \ -0.198008\ \ -0.499128\ \ -0.904618\ \ -0.000022\ \ -0.000003\ \ -0.702499\ \ -0.000003\ \ -0.583332\ \ -0.051359\ \ -0.000094]$, and $[-0.000000\ \ -1.464123\ \ -0.535877]$, respectively.

3.2.2. The fifth engineering design problem. The last engineering design problem, i.e. pressure vessel design problem, is chosen from [28], the mathematical description of which has been

DOI: 10.1002/nme

Table VII. Comparison of results for spring design.

	AATM	SCM
x_1	0.3596904119	0.368158695
x_2	0.0518130955	0.0521602170
x_3	11.1192526803	10.6484422590
Best	0.0126682620	0.012669249

Table VIII. Comparison of results for speed reducer design.

	AATM	SCM
x_1	3.5000162216	3.50000681
x_2	0.7000011771	0.70000001
x_3	17.0000298836	17
x_4	7.3002972901	7.32760205
x_5	7.7160494656	7.71532175
x_6	3.3502397985	3.35026702
x_7	5.2866604766	5.28665450
Best	2994.516778	2994.744241

Table IX. Comparison of results for three-bar truss design.

	AATM	SCM
x_1	0.7886817551	0.7886210370
x_2	0.4082295659	0.4084013340
Best	263.895843	263.8958466

presented in the Appendix. For more details about it, see [28]. This problem has been solved by several researchers, including Huang *et al*. [29], who used a co-evolutionary differential evolution, He and Wang [30], who employed hybrid particle swarm optimization with a feasibility-based rule, He and Wang [31], who used a co-evolutionary particle swarm optimization, He *et al*. [32], who proposed an improved particle swarm optimizer, and Wang and Yin [33], who adopted a ranking selection-based particle swarm optimizer. Note that this problem has two discrete variables (x_1 and x_2), which are handled in this paper by just rounding the real value to its closest integer value.

Table X summarizes the experimental results obtained by AATM over 50 trials. The number of FFEs for this problem is 30 000. The statistical results of AATM and the approaches mentioned above are also listed in Table X. From Table X, while the 'best' results provided by He and Wang [30], He *et al*. [32], and Wang and Yin [33] are slightly better than that provided by AATM, the 'mean' and 'worst' results obtained by AATM are significantly better than those of these three methods. Note that the approach proposed by He and Wang [30] requires 81 000 FFEs to produce the results shown in Table X. In contrast, AATM requires only 30 000 FFEs. In addition, AATM is superior to Huang *et al*. [29] and He and Wang [31] when measured by the selected performance metrics.

DOI: 10.1002/nme

Table X. Comparison between AATM and other methods on pressure vessel design problem.

Problem	Method	Best	Mean	Worst	St. dev.	Number of FFEs
Pressure vessel design	AATM	6059.7255	6061.9878	6090.8022	4.7E+00	30000
	Huang *et al.* [29]	6059.7340	6085.2303	6371.0455	4.3E+01	204800
	He and Wang [30]	6059.7143	6099.9323	6288.6770	8.6E+01	81000
	He and Wang [31]	6061.0777	6147.1332	6363.8041	8.6E+01	200000
	He *et al.* [32]	6059.7143	6289.92881	NA	3.1E+02	30000
	Wang and Yin [33]	6059.7143	6066.2032	6100.31956	1.3E+01	30000

NA = not available.

Table XI. Comparison of results for pressure vessel design.

	AATM	Huang *et al.* [29]	He and Wang [30]	He and Wang [31]	He *et al.* [32]	Wang and Yin [33]
x_1	0.81250000	0.812500	0.8125	0.812500	0.81250000	0.81250000
x_2	0.43750000	0.437500	0.4375	0.437500	0.43750000	0.43750000
x_3	42.0983827620	42.098411	42.0984	42.091266	42.09844560	42.09844560
x_4	176.6375285430	176.637690	176.6366	176.746500	176.63659584	176.63659584
Best	6059.72558	6059.7340	6059.7143	6061.0777	6059.7143	6059.7143

The best solutions obtained by the above six methods are shown in Table XI. The constraints based on the best result derived from AATM for this problem are: [−1.212694E−06 −0.035881 −0.857920 −63.362471].

The above comparison suggests that AATM is capable of dealing with engineering design problems in the real world.

4. DISCUSSION

In this section, the effectiveness of the shrinking space technique and the shrinking termination scheme has been investigated experimentally. In addition, the effect of two vital parameters (i.e. T and β) in the shrinking space technique on the performance of AATM has been demonstrated by various experiments.

4.1. Effectiveness of the shrinking space technique

In order to verify the effectiveness of the shrinking space technique on the performance of AATM, an additional experiment (denoted as AATM_1) has been executed on 18 benchmark test functions in which the shrinking space technique is removed from AATM. All control parameters in AATM_1 are kept unchanged. For each test function, 30 independent runs are conducted to have a fair competition. The experimental results of AATM and AATM_1 have been compared in Table XII.

Table XII shows that for six test functions (i.e. g01, g04, g06, g08, g11, and g24), AATM and AATM_1 can consistently converge to the global optima in all trials. For test functions g02, g07, g10, g14, and g19, AATM outperforms AATM_1 in terms of all the criteria. For test functions g05,

DOI: 10.1002/nme

Table XII. Comparing AATM with respect to AATM_1 on 18 benchmark test functions over 30 independent runs.

Function	Optimal	Method	Best	Mean	Worst
g01	−15.000	AATM	−15.000	−15.000	−15.000
		AATM_1	−15.000	−15.000	−15.000
g02	−0.803619	AATM	−0.803389	−0.791213	−0.767040
		AATM_1	−0.801214	−0.786745	−0.740953
g03	−1.00	AATM	−1.00	−1.00	−1.00
		AATM_1	(28)		
g04	−30665.539	AATM	−30665.539	−30665.539	−30665.539
		AATM_1	−30665.539	−30665.539	−30665.539
g05	5126.498	AATM	5126.498	5126.714	5128.824
		AATM_1	5126.498	5129.542*	5145.420
g06	−6961.814	AATM	−6961.814	−6961.814	−6961.814
		AATM_1	−6961.814	−6961.814	−6961.814
g07	24.306	AATM	24.307	24.317	24.356
		AATM_1	24.308	24.322	24.380
g08	−0.095825	AATM	−0.095825	−0.095825	−0.095825
		AATM_1	−0.095825	−0.095825	−0.095825
g09	680.630	AATM	680.630	680.634	680.646
		AATM_1	680.630	680.642*	680.700
g10	7049.248	AATM	7049.603	7077.477	7183.295
		AATM_1	7052.570	7341.590*	7931.979
g11	0.75	AATM	0.75	0.75	0.75
		AATM_1	0.75	0.75	0.75
g12	−1.000	AATM	−1.000	−1.000	−1.000
		AATM_1	−1.000	−0.999*	−0.994
g14	−47.763	AATM	−47.762	−47.750	−47.712
		AATM_1	−47.760	−47.696*	−47.515
g15	961.715	AATM	961.715	961.715	961.716
		AATM_1	961.715	961.718*	961.728
g16	−1.905155	AATM	−1.905155	−1.905155	−1.905155
		AATM_1	−1.905155	−1.904223*	−1.900094
g18	−0.866025	AATM	−0.866025	−0.865952	−0.864843
		AATM_1	−0.866025	−0.840490*	−0.674874
g19	32.656	AATM	32.725	32.952	33.243
		AATM_1	32.947	33.412*	34.634
g24	−5.508013	AATM	−5.508013	−5.508013	−5.508013
		AATM_1	−5.508013	−5.508013	−5.508013

(#) denotes the number that feasible solutions are found in the final population over 30 trials.
*indicates the t value is significant at a 0.05 level of significance by two-tailed t-test.

g09, g12, g15, g16, and g18, compared with AATM_1, AATM provides similar 'best' results yet better 'mean' and 'worst' results. For test function g03, AATM_1 cannot find feasible solutions consistently.

t-test is performed between AATM and AATM_1. Since feasible solutions cannot be found consistently by AATM_1 on test function g03, t-test is not performed between AATM and AATM_1 on this test function. From Table XII, it can be seen that AATM performs better than AATM_1 on test functions g05, g09, g10, g12, g14, g15, g16, g18, and g19. AATM_1 cannot show better results than AATM even on one test function.

DOI: 10.1002/nme

Based on the comparison above, it can be concluded that AATM outperforms AATM_1 on the whole, which verifies that the shrinking space technique can improve the performance of ATM significantly.

4.2. Effectiveness of the shrinking termination scheme

As previously analyzed in Section 2.3, the shrinking termination scheme is proposed to avoid the search space being reduced too much. The effectiveness of this scheme is demonstrated by using two different experiments on 18 benchmark test functions, i.e. execute AATM with and without using this scheme. AATM without using this scheme is denoted as AATM_2.

The experimental results provided by AATM and AATM_2 are shown in Table XIII. From Table XIII, AATM and AATM_2 are able to locate the global optima consistently for test functions g04, g06, g08, g11, g12, g16, and g24. Compared with AATM, the overall performance deterioration take places in test functions g01, g02, g03, g05, g07, g09, g10, g14, g18, and g19 for AATM_2. More importantly, for test functions g03 and g05 AATM_2 can only find feasible solutions for 21 and 28 out of the 30 runs, respectively; for test function g14, AATM_2 cannot succeed in solving this test function even in one run. Additionally, AATM and AATM_2 perform similarly in test function g15.

t-test is performed between AATM and AATM_2. Note that feasible solutions cannot be found consistently by AATM_2 on test function g03, g05, and g14; therefore, t-test is not made between AATM and AATM_2 on these test functions. As shown in Table XIII, AATM has superior performance on test functions g01, g07, g09, g18, and g19 compared with AATM_2. However, AATM_2 cannot show better results than AATM even on one test function.

The above comparison suggests that the shrinking termination scheme has a great impact on the performance of AATM, and that by making use of this scheme the performance of AATM can be enhanced remarkably.

4.3. Effect of the parameter T in the shrinking space technique

The parameter T controls the shrinking rate of the search space. If the value of this parameter is too small, the search space will be reduced frequently, as a consequence the global search capability of the algorithm might not be good and the population might be easily trapped in a local optimum. In contrast, if the value of this parameter is too large, the search space will be shrunk less often, thus the shrinking space technique cannot exert its strength very well. Therefore, a suitable value of this parameter must be selected. Table XIV summarizes the mean of the objective function values, in the case of the parameter T alone being changed to 10, 15, 20, 25, and 30.

As shown in Table XIV, in the case of $T=10$, the degradation of capability arises for test functions g02, g05, g07, g10, g12, and g18. In addition, in the case of $T=30$, though the algorithm can provide the best result for test function g10, the results of test functions g01, g02, and g19 are clearly dominated by other results. In the case of $T=15$, 20, and 25, the algorithms can obtain similar performance. These results reveal that a value of the parameter T between 15 and 25 is an appropriate choice for AATM.

4.4. Influence of the parameter β in the shrinking space technique

The parameter β controls the shrinking range of the search space. In order to illustrate the effect of the parameter β on the performance of AATM, six different values of this parameter are

DOI: 10.1002/nme

Table XIII. Comparing AATM with respect to AATM_2 on 18 benchmark test functions over 30 independent runs.

Function	Optimal	Method	Best	Mean	Worst
g01	−15.000	AATM	−15.000	−15.000	−15.000
		AATM_2	−14.996	−14.995*	−14.993
g02	−0.803619	AATM	−0.803389	−0.791213	−0.767040
		AATM_2	−0.802864	−0.785616	−0.736193
g03	−1.00	AATM	−1.00	−1.00	−1.00
		AATM_2	(21)		
g04	−30665.539	AATM	−30665.539	−30665.539	−30665.539
		AATM_2	−30665.539	−30665.539	−30665.539
g05	5126.498	AATM	5126.498	5126.714	5128.824
		AATM_2	(28)		
g06	−6961.814	AATM	−6961.814	−6961.814	−6961.814
		AATM_2	−6961.814	−6961.814	−6961.814
g07	24.306	AATM	24.307	24.317	24.356
		AATM_2	24.318	24.368*	24.456
g08	−0.095825	AATM	−0.095825	−0.095825	−0.095825
		AATM_2	−0.095825	−0.095825	−0.095825
g09	680.630	AATM	680.630	680.634	680.646
		AATM_2	680.637	680.659*	680.696
g10	7049.248	AATM	7049.603	7077.477	7183.295
		AATM_2	7052.589	7078.186	7185.274
g11	0.75	AATM	0.75	0.75	0.75
		AATM_2	0.75	0.75	0.75
g12	−1.000	AATM	−1.000	−1.000	−1.000
		AATM_2	−1.000	−1.000	−1.000
g14	−47.763	AATM	−47.762	−47.750	−47.712
		AATM_2	(0)		
g15	961.715	AATM	961.715	961.715	961.716
		AATM_2	961.715	961.715	961.716
g16	−1.905155	AATM	−1.905155	−1.905155	−1.905155
		AATM_2	−1.905155	−1.905155	−1.905155
g18	−0.866025	AATM	−0.866025	−0.865952	−0.864843
		AATM_2	−0.865282	−0.862525*	−0.855155
g19	32.656	AATM	32.725	32.952	33.243
		AATM_2	32.938	33.159*	33.607
g24	−5.508013	AATM	−5.508013	−5.508013	−5.508013
		AATM_2	−5.508013	−5.508013	−5.508013

(#) denotes the number that feasible solutions are found in the final population over 30 trials.
*indicates the t value is significant at a 0.05 level of significance by two-tailed t-test.

tested using the same remaining parameters: $0.00^{(1/n)}$, $0.01^{(1/n)}$, $0.02^{(1/n)}$, $0.03^{(1/n)}$, $0.05^{(1/n)}$, and $0.10^{(1/n)}$. The mean of the objective function values are given in Table XV. It is noteworthy that $\beta = 0.00^{(1/n)}$ implies that the reduced search space is just equal to the region enclosed by the potential individuals. Moreover, a smaller value of this parameter signifies faster shrinking speed of the search space, and vice versa.

As shown in Table XV, in the case of $\beta = 0.00^{(1/n)}$, the results for test functions g02, g05, g10, g14, and g18 are much worse than other results. In the case of $\beta = 0.10^{(1/n)}$, the results for

DOI: 10.1002/nme

Table XIV. Experimental results on 18 benchmark functions with varying parameter T; 30 independent runs are performed.

Function	10	15	20	25	30
g01	**−15.000**	**−15.000**	**−15.000**	**−15.000**	−14.985
g02	−0.786946	−0.790825	**−0.791213**	−0.789612	−0.784542
g03	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**−1.00**
g04	**−30665.539**	**−30665.539**	**−30665.539**	**−30665.539**	**−30665.539**
g05	5126.915	5126.717	**5126.714**	5126.757	5126.737
g06	**−6961.814**	**−6961.814**	**−6961.814**	**−6961.814**	**−6961.814**
g07	24.325	24.318	**24.317**	24.318	24.324
g08	**−0.095825**	**−0.095825**	**−0.095825**	**−0.095825**	**−0.095825**
g09	**680.633**	680.634	680.634	680.635	**680.633**
g10	7228.175	7078.312	7077.477	7074.154	**7067.445**
g11	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**
g12	−0.973	−0.996	**−1.000**	**−1.000**	**−1.000**
g14	**−47.750**	−47.749	**−47.750**	−47.749	**−47.750**
g15	961.716	**961.715**	**961.715**	**961.715**	961.716
g16	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**
g18	−0.859418	**−0.865980**	−0.865952	−0.865958	−0.859542
g19	**32.931**	32.955	32.952	33.075	33.406
g24	**−5.508013**	**−5.508013**	**−5.508013**	**−5.508013**	**−5.508013**

A result in boldface indicates a better result or that the global optimum (or best known solution) is reached.

Table XV. Experimental results on 18 benchmark functions with varying parameter β; 30 independent runs are performed.

Function	0.00	0.01	0.02	0.03	0.05	0.10
g01	**−15.000**	**−15.000**	**−15.000**	**−15.000**	**−15.000**	**−14.900**
g02	−0.775297	−0.789543	**−0.791213**	−0.7901942	−0.789294	−0.786222
g03	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**−1.00**	**−1.00**
g04	**−30665.539**	**−30665.539**	**−30665.539**	**−30665.539**	**−30665.539**	**−30665.539**
g05	5127.770	5127.026	**5126.714**	5126.830	5126.715	5126.722
g06	**−6961.814**	**−6961.814**	**−6961.814**	**−6961.814**	**−6961.814**	**−6961.814**
g07	24.321	**24.317**	**24.317**	24.319	**24.317**	24.351
g08	**−0.095825**	**−0.095825**	**−0.095825**	**−0.095825**	**−0.095825**	**−0.095825**
g09	680.634	680.634	680.634	**680.633**	680.634	680.634
g10	7123.471	7088.479	7077.477	**7076.417**	7076.424	7077.676
g11	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**
g12	**−1.000**	**−1.000**	**−1.000**	**−1.000**	**−1.000**	**−1.000**
g14	−47.730	−47.747	**−47.750**	−47.748	−47.749	**−47.750**
g15	961.716	**961.715**	**961.715**	961.716	961.716	961.716
g16	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**	**−1.905155**
g18	−0.859589	−0.865950	**−0.865952**	−0.865951	**−0.865952**	−0.863880
g19	33.255	32.992	**32.952**	33.031	33.193	35.055
g24	**−5.508013**	**−5.508013**	**−5.508013**	**−5.508013**	**−5.508013**	**−5.508013**

A result in boldface indicates a better result or that the global optimum (or best known solution) is reached.

test functions g01, g07, and g19 are much worse than other results. On the average, there is no significant difference when $\beta=0.01^{(1/n)}$, $0.02^{(1/n)}$, $0.03^{(1/n)}$, and $0.05^{(1/n)}$, which means a value between $0.01^{(1/n)}$ and $0.05^{(1/n)}$ is suitable for this parameter of AATM.

DOI: 10.1002/nme

From Tables XIV and XV, one can conclude that AATM is not sensitive in relation to the two parameters (i.e. T and β) in the shrinking space technique and that these two parameters can be set among large ranges to obtain high-quality solutions.

5. CONCLUSION AND FUTURE WORK

The adaptive trade-off model (ATM) is a recent method to cope with constrained optimization problems (COPs), the main advantages of which are its simplicity, adaptation, and ease of implementation. In the real-world applications, the optimization algorithm should be able to reach the global optimum using as few number of FFEs as possible, since the evaluation of objective function and constraints is often the most expensive part of the search for constrained optimization. Therefore, in an attempt to accelerate ATM, the shrinking space technique [20] is incorporated. As a result, a new method named AATM is proposed. It is noteworthy that a shrinking termination scheme is proposed so as to make the shrinking space technique more effective.

The performance of AATM is investigated using 18 benchmark test functions and five engineering design problems. The experimental results show that AATM significantly outperforms ATM and IS-PAES in terms of convergence velocity in all benchmark test functions, and that AATM performs better than or similar to ATM and IS-PAES in terms of the quality of the resulting solutions. Besides, the overall performance of AATM is superior to SCM on four engineering design problems; moreover, AATM is competitive with five recent methods on pressure vessel design problem. AATM is also compared with two other versions of AATM, i.e. AATM without using the shrinking space technique denoted as AATM_1 and AATM without using the shrinking termination scheme denoted as AATM_2. Experimental results verify the effectiveness of the shrinking space technique and the shrinking termination scheme. Meanwhile, the effect of some crucial parameters in the shrinking space technique on the performance of AATM has also been demonstrated.

Although the capability of ATM can be significantly improved by taking advantage of the shrinking space technique, compared with other methods such as [24, 34], AATM still leaves plenty of room for improvement. When using EAs to deal with COPs, the search algorithm plays a very important role on the performance in addition to the constraint-handling technique. It is necessary to note that the classical ES is used as the search engine in this paper, whereas [24, 34] use hybrid EA and improved particle swarm optimization as the search engine, respectively. Therefore, the main work of future research consists in improving the classical ES or employing some advanced search algorithms (such as particle swarm optimization and differential evolution) to further enhance the performance of AATM.

APPENDIX A

(1) g01

$$\text{Minimize} \quad f(\mathbf{x}) = 5\sum_{i=1}^{4} x_i - 5\sum_{i=1}^{4} x_i^2 - \sum_{i=5}^{13} x_i$$

$$\text{subject to} \quad g_1(\mathbf{x}) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leqslant 0, \quad g_2(\mathbf{x}) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leqslant 0$$

$$g_3(\mathbf{x}) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leqslant 0, \quad g_4(\mathbf{x}) = -8x_1 + x_{10} \leqslant 0$$

DOI: 10.1002/nme

$$g_5(\mathbf{x}) = -8x_2 + x_{11} \leqslant 0, \quad g_6(\mathbf{x}) = -8x_3 + x_{12} \leqslant 0$$

$$g_7(\mathbf{x}) = -2x_4 - x_5 + x_{10} \leqslant 0, \quad g_8(\mathbf{x}) = -2x_6 - x_7 + x_{11} \leqslant 0$$

$$g_9(\mathbf{x}) = -2x_8 - x_9 + x_{12} \leqslant 0$$

where $0 \leqslant x_i \leqslant 1$ $(i = 1, \ldots, 9)$, $0 \leqslant x_i \leqslant 100$ $(i = 10, 11, 12)$ and $0 \leqslant x_{13} \leqslant 1$.

The global minimum is at $\mathbf{x}^* = (1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$, where $f(\mathbf{x}^*) = -15$.

(2) g02

$$\text{Maximize} \quad f(\mathbf{x}) = \left| \frac{\sum_{i=1}^{n} \cos^4(x_i) - 2\prod_{i=1}^{n} \cos^2(x_i)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \right|$$

$$\text{subject to} \quad g_1(\mathbf{x}) = 0.75 - \prod_{i=1}^{n} x_i \leqslant 0, \quad g_2(\mathbf{x}) = \sum_{i=1}^{n} x_i - 7.5n \leqslant 0$$

where $n = 20$ and $0 \leqslant x_i \leqslant 10$ $(i = 1, \ldots, n)$.

The global maximum is unknown; the best reported solution is $f(\mathbf{x}^*) = 0.803619$.

(3) g03

$$\text{Maximize} \quad f(\mathbf{x}) = (\sqrt{n})^n \prod_{i=1}^{n} x_i$$

$$\text{subject to} \quad h(\mathbf{x}) = \sum_{i=1}^{n} x_i^2 - 1 = 0$$

where $n = 10$ and $0 \leqslant x_i \leqslant 1$ $(i = 1, \ldots, n)$.

The global maximum is at $x_i^* = 1/\sqrt{n}$ $(i = 1, \ldots, n)$, where $f(\mathbf{x}^*) = 1$.

(4) g04

$$\text{Minimize} \quad f(\mathbf{x}) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141$$

$$\text{subject to} \quad g_1(\mathbf{x}) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leqslant 0$$

$$g_2(\mathbf{x}) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leqslant 0$$

$$g_3(\mathbf{x}) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \leqslant 0$$

$$g_4(\mathbf{x}) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leqslant 0$$

$$g_5(\mathbf{x}) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \leqslant 0$$

$$g_6(\mathbf{x}) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leqslant 0$$

where $78 \leqslant x_1 \leqslant 102$, $33 \leqslant x_2 \leqslant 45$, and $27 \leqslant x_i \leqslant 45$ $(i = 3, 4, 5)$.

The optimum solution is $\mathbf{x}^* = (78, 33, 29.995256025682, 45, 36.775812905788)$, where $f(\mathbf{x}^*) = -30665.539$.

DOI: 10.1002/nme

(5) g05

$$\begin{aligned}
\text{Minimize} \quad & f(\mathbf{x}) = 3x_1 + 0.000001x_1^3 + 2x_2 + (0.000002/3)x_2^3 \\
\text{subject to} \quad & g_1(\mathbf{x}) = -x_4 + x_3 - 0.55 \leqslant 0, \quad g_2(\mathbf{x}) = -x_3 + x_4 - 0.55 \leqslant 0 \\
& h_3(\mathbf{x}) = 1000\sin(-x_3 - 0.25) + 1000\sin(-x_4 - 0.25) + 894.8 - x_1 = 0 \\
& h_4(\mathbf{x}) = 1000\sin(x_3 - 0.25) + 1000\sin(x_3 - x_4 - 0.25) + 894.8 - x_2 = 0 \\
& h_5(\mathbf{x}) = 1000\sin(x_4 - 0.25) + 1000\sin(x_4 - x_3 - 0.25) + 1294.8 = 0
\end{aligned}$$

where $0 \leqslant x_1 \leqslant 1200$, $0 \leqslant x_2 \leqslant 1200$, $-0.55 \leqslant x_3 \leqslant 0.55$, and $-0.55 \leqslant x_4 \leqslant 0.55$.

The best-known solution is $\mathbf{x}^* = (679.9453, 1026.067, 0.1188764, -0.3962336)$, where $f(\mathbf{x}^*) = 5126.4981$.

(6) g06

$$\begin{aligned}
\text{Minimize} \quad & f(\mathbf{x}) = (x_1 - 10)^3 + (x_2 - 20)^3 \\
\text{subject to} \quad & g_1(\mathbf{x}) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leqslant 0 \\
& g_2(\mathbf{x}) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leqslant 0
\end{aligned}$$

where $13 \leqslant x_1 \leqslant 100$ and $0 \leqslant x_2 \leqslant 100$.

The optimum solution is $\mathbf{x}^* = (14.095, 0.84296)$, where $f(\mathbf{x}^*) = -6961.81388$.

(7) g07

$$\begin{aligned}
\text{Minimize} \quad & f(\mathbf{x}) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 \\
& \qquad + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45 \\
\text{subject to} \quad & g_1(\mathbf{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leqslant 0 \\
& g_2(\mathbf{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leqslant 0 \\
& g_3(\mathbf{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leqslant 0 \\
& g_4(\mathbf{x}) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leqslant 0 \\
& g_5(\mathbf{x}) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leqslant 0 \\
& g_6(\mathbf{x}) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leqslant 0 \\
& g_7(\mathbf{x}) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leqslant 0 \\
& g_8(\mathbf{x}) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leqslant 0
\end{aligned}$$

where $-10 \leqslant x_i \leqslant 10$ $(i = 1, \ldots, 10)$.

The optimum solution is $\mathbf{x}^* = (2.171996, 2.363683, 8.773926, 5.095984, 0.9906548, 1.430574, 1.321644, 9.828726, 8.280092, 8.375927)$, where $f(\mathbf{x}^*) = 24.3062091$.

(8) g08

$$\text{Maximize} \quad f(\mathbf{x}) = \frac{\sin^3(2\pi x_1)\sin(2\pi x_2)}{x_1^3(x_1+x_2)}$$

$$\text{subject to} \quad g_1(\mathbf{x}) = x_1^2 - x_2 + 1 \leqslant 0, \quad g_2(\mathbf{x}) = 1 - x_1 + (x_2-4)^2 \leqslant 0$$

where $0 \leqslant x_1 \leqslant 10$ and $0 \leqslant x_2 \leqslant 10$.

The optimum solution is $\mathbf{x}^* = (1.2279713, 4.2453733)$, where $f(\mathbf{x}^*) = 0.095825$.

(9) g09

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = (x_1-10)^2 + 5(x_2-12)^2 + x_3^4 + 3(x_4-11)^2 \\ & \quad + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \\ \text{subject to} \quad & g_1(\mathbf{x}) = -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leqslant 0 \\ & g_2(\mathbf{x}) = -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leqslant 0 \\ & g_3(\mathbf{x}) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leqslant 0 \\ & g_4(\mathbf{x}) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leqslant 0 \end{aligned}$$

where $-10 \leqslant x_i \leqslant 10$ $(i = 1, \ldots, 7)$.

The optimum solution is $\mathbf{x}^* = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.1038131, 1.594227)$, where $f(\mathbf{x}^*) = 680.6300573$.

(10) g10

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = x_1 + x_2 + x_3 \\ \text{subject to} \quad & g_1(\mathbf{x}) = -1 + 0.0025(x_4 + x_6) \leqslant 0 \\ & g_2(\mathbf{x}) = -1 + 0.0025(x_5 + x_7 - x_4) \leqslant 0 \\ & g_3(\mathbf{x}) = -1 + 0.01(x_8 - x_5) \leqslant 0 \\ & g_4(\mathbf{x}) = -x_1x_6 + 833.33252x_4 + 100x_1 - 83\,333.333 \leqslant 0 \\ & g_5(\mathbf{x}) = -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leqslant 0 \\ & g_6(\mathbf{x}) = -x_3x_8 + 1\,250\,000 + x_3x_5 - 2500x_5 \leqslant 0 \end{aligned}$$

where $100 \leqslant x_1 \leqslant 10\,000$, $1000 \leqslant x_i \leqslant 10\,000$ $(i = 2, 3)$, and $10 \leqslant x_i \leqslant 1000$ $(i = 4, \ldots, 8)$.

The optimum solution is $\mathbf{x}^* = (579.306685, 1359.970678, 5109.970657, 182.017700, 295.601174, 217.982300, 286.416526, 395.601174)$, where $f(\mathbf{x}^*) = 7049.248$.

(11) g11

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = x_1^2 + (x_2-1)^2 \\ \text{subject to} \quad & h(\mathbf{x}) = x_2 - x_1^2 = 0 \end{aligned}$$

where $-1 \leqslant x_1 \leqslant 1$ and $-1 \leqslant x_2 \leqslant 1$.

DOI: 10.1002/nme

The optimum solution is $\mathbf{x}^* = (\pm 1/\sqrt{2}, \frac{1}{2})$, where $f(\mathbf{x}^*) = 0.75$.
(12) g12

$$\text{Maximize} \quad f(\mathbf{x}) = (100 - (x_1-5)^2 - (x_2-5)^2 - (x_3-5)^2)/100$$
$$\text{subject to} \quad g(\mathbf{x}) = (x_1-p)^2 + (x_2-q)^2 + (x_3-r)^2 - 0.0625 \leqslant 0$$

where $0 \leqslant x_i \leqslant 10$ $(i = 1, 2, 3)$ and $p, q, r = 1, 2, \ldots, 9$.
The optimum solution is $\mathbf{x}^* = (5, 5, 5)$, where $f(\mathbf{x}^*) = 1$.
(13) g14

$$\text{Minimize} \quad f(\mathbf{x}) = \sum_{i=1}^{10} x_i \left(c_i + \ln \frac{x_i}{\sum_{j=1}^{10} x_j} \right)$$
$$\text{subject to} \quad h_1(\mathbf{x}) = x_1 + 2x_2 + 2x_3 + x_6 + x_{10} - 2 = 0, \quad h_2(\mathbf{x}) = x_4 + 2x_5 + x_6 + x_7 - 1 = 0$$
$$h_3(\mathbf{x}) = x_3 + x_7 + x_8 + 2x_9 + x_{10} - 1 = 0$$

where $0 \leqslant x_i \leqslant 10$ $(i = 1, \ldots, 10)$, $c_1 = -6.089$, $c_2 = -17.164$, $c_3 = -34.054$, $c_4 = -5.914$, $c_5 = -24.721$, $c_6 = -14.986$, $c_7 = -24.1$, $c_8 = -10.708$, $c_9 = -26.662$, and $c_{10} = -22.179$.
The optimum solution is $\mathbf{x}^* = (0.040668, 0.147726, 0.783180, 0.001414, 0.485270, 0.000693, 0.027402, 0.017949, 0.037321, 0.096878)$ where $f(\mathbf{x}^*) = -47.762990$.
(14) g15

$$\text{Minimize} \quad f(\mathbf{x}) = 1000 - x_1^2 - 2x_2^2 - x_3^2 - x_1x_2 - x_1x_3$$
$$\text{subject to} \quad h_1(\mathbf{x}) = x_1^2 + x_2^2 + x_3^2 - 25 = 0, \quad h_2(\mathbf{x}) = 8x_1 + 14x_2 + 7x_3 - 56 = 0$$

where $0 \leqslant x_i \leqslant 10$ $(i = 1, 2, 3)$.
The optimum solution is $\mathbf{x}^* = (3.512128, 0.216988, 3.552179)$ where $f(\mathbf{x}^*) = 961.715$.
(15) g16

$$\text{Minimize} \quad f(\mathbf{x}) = 0.000117y_{14} + 0.1365 + 0.00002358y_{13} + 0.000001502y_{16} + 0.0321y_{12}$$
$$+ 0.004324y_5 + 0.0001\frac{c_{15}}{c_{16}} + 37.48\frac{y_2}{c_{12}} - 0.0000005843y_{17}$$
$$\text{subject to} \quad g_1(\mathbf{x}) = \frac{0.28}{0.72}y_5 - y_4 \leqslant 0, \quad g_2(\mathbf{x}) = x_3 - 1.5x_2 \leqslant 0, \quad g_3(\mathbf{x}) = 3496\frac{y_2}{c_{12}} - 21 \leqslant 0$$
$$g_4(\mathbf{x}) = 110.6 + y_1 - \frac{62212}{c_{17}} \leqslant 0, \quad g_5(\mathbf{x}) = 213.1 - y_1 \leqslant 0, \quad g_6(\mathbf{x}) = y_1 - 405.23 \leqslant 0$$
$$g_7(\mathbf{x}) = 17.505 - y_2 \leqslant 0, \quad g_8(\mathbf{x}) = y_2 - 1053.6667 \leqslant 0$$
$$g_9(\mathbf{x}) = 11.275 - y_3 \leqslant 0, \quad g_{10}(\mathbf{x}) = y_3 - 35.03 \leqslant 0, \quad g_{11}(\mathbf{x}) = 214.228 - y_4 \leqslant 0$$
$$g_{12}(\mathbf{x}) = y_4 - 665.585 \leqslant 0, \quad g_{13}(\mathbf{x}) = 7.458 - y_5 \leqslant 0, \quad g_{14}(\mathbf{x}) = y_5 - 584.463 \leqslant 0$$
$$g_{15}(\mathbf{x}) = 0.961 - y_6 \leqslant 0, \quad g_{16}(\mathbf{x}) = y_6 - 265.916 \leqslant 0, \quad g_{17}(\mathbf{x}) = 1.612 - y_7 \leqslant 0$$
$$g_{18}(\mathbf{x}) = y_7 - 7.046 \leqslant 0, \quad g_{19}(\mathbf{x}) = 0.146 - y_8 \leqslant 0, \quad g_{20}(\mathbf{x}) = y_8 - 0.222 \leqslant 0$$

DOI: 10.1002/nme

$$g_{21}(\mathbf{x}) = 107.99 - y_9 \leqslant 0, \quad g_{22}(\mathbf{x}) = y_9 - 273.366 \leqslant 0, \quad g_{23}(\mathbf{x}) = 922.693 - y_{10} \leqslant 0$$

$$g_{24}(\mathbf{x}) = y_{10} - 1286.105 \leqslant 0, \quad g_{25}(\mathbf{x}) = 926.832 - y_{11} \leqslant 0, \quad g_{26}(\mathbf{x}) = y_{11} - 1444.046 \leqslant 0$$

$$g_{27}(\mathbf{x}) = 18.766 - y_{12} \leqslant 0, \quad g_{28}(\mathbf{x}) = y_{12} - 537.141 \leqslant 0, \quad g_{29}(\mathbf{x}) = 1072.163 - y_{13} \leqslant 0$$

$$g_{30}(\mathbf{x}) = y_{13} - 3247.039 \leqslant 0, \quad g_{31}(\mathbf{x}) = 8961.448 - y_{14} \leqslant 0$$

$$g_{32}(\mathbf{x}) = y_{14} - 26844.086 \leqslant 0, \quad g_{33}(\mathbf{x}) = 0.063 - y_{15} \leqslant 0, \quad g_{34}(\mathbf{x}) = y_{15} - 0.386 \leqslant 0$$

$$g_{35}(\mathbf{x}) = 71084.33 - y_{16} \leqslant 0, \quad g_{36}(\mathbf{x}) = -140000 + y_{16} \leqslant 0$$

$$g_{37}(\mathbf{x}) = 2802713 - y_{17} \leqslant 0, \quad g_{38}(\mathbf{x}) = y_{17} - 12146108 \leqslant 0$$

where

$$y_1 = x_2 + x_3 + 41.6, \quad c_1 = 0.024x_4 - 4.62, \quad y_2 = \frac{12.5}{c_1} + 12$$

$$c_2 = 0.0003535x_1^2 + 0.5311x_1 + 0.08705y_2x_1, \quad c_3 = 0.052x_1 + 78 + 0.002377y_2x_1$$

$$y_3 = \frac{c_2}{c_3}, \quad y_4 = 19y_3, \quad c_4 = 0.04782(x_1 - y_3) + \frac{0.1956(x_1 - y_3)^2}{x_2} + 0.6376y_4 + 1.594y_3$$

$$c_5 = 100x_2, \quad c_6 = x_1 - y_3 - y_4, \quad c_7 = 0.950 - \frac{c_4}{c_5}, \quad y_5 = c_6c_7, \quad y_6 = x_1 - y_5 - y_4 - y_3$$

$$c_8 = (y_5 + y_4)0.995, \quad y_7 = \frac{c_8}{y_1}, \quad y_8 = \frac{c_8}{3798}, \quad c_9 = y_7 - \frac{0.0663y_7}{y_8} - 0.3153$$

$$y_9 = \frac{96.82}{c_9} + 0.321y_1, \quad y_{10} = 1.29y_5 + 1.258y_4 + 2.29y_3 + 1.71y_6$$

$$y_{11} = 1.71x_1 - 0.452y_4 + 0.580y_3, \quad c_{10} = \frac{12.3}{752.3}, \quad c_{11} = (1.75y_2)(0.995x_1)$$

$$c_{12} = 0.995y_{10} + 1998, \quad y_{12} = c_{10}x_1 + \frac{c_{11}}{c_{12}}, \quad y_{13} = c_{12} - 1.75y_2$$

$$y_{14} = 3623 + 64.4x_2 + 58.4x_3 + \frac{146312}{y_9 + x_5}, \quad c_{13} = 0.995y_{10} + 60.8x_2 + 48x_4 - 0.1121y_{14} - 5095$$

$$y_{15} = \frac{y_{13}}{c_{13}}, \quad y_{16} = 148\,000 - 331\,000y_{15} + 40y_{13} - 61y_{15}y_{13}, \quad c_{14} = 2324y_{10} - 28\,740\,000y_2$$

$$y_{17} = 14\,130\,000 - 1328y_{10} - 531y_{11} + \frac{c_{14}}{c_{12}}, \quad c_{15} = \frac{y_{13}}{y_{15}} - \frac{y_{13}}{0.52}$$

$$c_{16} = 1.104 - 0.72y_{15}, \quad c_{17} = y_9 + x_5$$

and where $704.4148 \leqslant x_1 \leqslant 906.3855$, $68.6 \leqslant x_2 \leqslant 288.88$, $0 \leqslant x_3 \leqslant 134.75$, $193 \leqslant x_4 \leqslant 287.0966$, and $25 \leqslant x_5 \leqslant 84.1988$.

The optimum solution is $\mathbf{x}^* = (705.174537, 68.600000, 102.900000, 282.324932, 37.584116)$ where $f(\mathbf{x}^*) = -1.905155$.

DOI: 10.1002/nme

(16) g18

$$\text{Minimize} \quad f(\mathbf{x}) = -0.5(x_1x_4 - x_2x_3 + x_3x_9 - x_5x_9 + x_5x_8 - x_6x_7)$$

$$\text{subject to} \quad g_1(\mathbf{x}) = x_3^2 + x_4^2 - 1 \leqslant 0, \quad g_2(\mathbf{x}) = x_9^2 - 1 \leqslant 0, \quad g_3(\mathbf{x}) = x_5^2 + x_6^2 - 1 \leqslant 0$$

$$g_4(\mathbf{x}) = x_1^2 + (x_2 - x_9)^2 - 1 \leqslant 0, \quad g_5(\mathbf{x}) = (x_1 - x_5)^2 + (x_2 - x_6)^2 - 1 \leqslant 0$$

$$g_6(\mathbf{x}) = (x_1 - x_7)^2 + (x_2 - x_8)^2 - 1 \leqslant 0, \quad g_7(\mathbf{x}) = (x_3 - x_5)^2 + (x_4 - x_6)^2 - 1 \leqslant 0$$

$$g_8(\mathbf{x}) = (x_3 - x_7)^2 + (x_4 - x_8)^2 - 1 \leqslant 0, \quad g_9(\mathbf{x}) = x_7^2 + (x_8 - x_9)^2 - 1 \leqslant 0$$

$$g_{10}(\mathbf{x}) = x_2x_3 - x_1x_4 \leqslant 0, \quad g_{11}(\mathbf{x}) = -x_3x_9 \leqslant 0$$

$$g_{12}(\mathbf{x}) = x_5x_9 \leqslant 0, \quad g_{13}(\mathbf{x}) = x_6x_7 - x_5x_8 \leqslant 0$$

where $-10 \leqslant x_i \leqslant 10$ $(i = 1, \ldots, 8)$ and $0 \leqslant x_9 \leqslant 20$.

The optimum solution is $\mathbf{x}^* = (-0.657776, -0.153419, 0.323414, -0.946258, -0.657776, -0.753213, 0.323414, -0.346463, 0.599795)$ where $f(\mathbf{x}^*) = -0.866025$.

(17) g19

$$\text{Minimize} \quad f(\mathbf{x}) = \sum_{j=1}^{5}\sum_{i=1}^{5} c_{ij}x_{(10+i)}x_{(10+j)} + 2\sum_{j=1}^{5} d_j x_{(10+j)}^3 - \sum_{i=1}^{10} b_i x_i$$

$$\text{subject to} \quad g_j(\mathbf{x}) = -2\sum_{i=1}^{5} c_{ij}x_{(10+i)} - 3d_j x_{(10+j)}^2 - \mathrm{e}_j + \sum_{i=1}^{10} a_{ij}x_i \leqslant 0, \quad j = 1, \ldots, 5$$

where $0 \leqslant x_i \leqslant 10$ $(i = 1, \ldots, 15)$, $\mathbf{b} = [-40\ \ -2\ \ -0.25\ \ -4\ \ -4\ \ -1\ \ -40\ \ -60\ \ 5\ \ 1]$ and the remaining data is on Table AI.

Table AI. Data set for test function g19.

j	1	2	3	4	5
e_j	−15	−27	−36	−18	−12
c_{1j}	30	−20	−10	32	−10
c_{2j}	−20	39	−6	−31	32
c_{3j}	−10	−6	10	−6	−10
c_{4j}	32	−31	−6	39	−20
c_{5j}	−10	32	−10	−20	30
d_j	4	8	10	6	2
a_{1j}	−16	2	0	1	0
a_{2j}	0	−2	0	0.4	2
a_{3j}	−3.5	0	2	0	0
a_{4j}	0	−2	0	−4	−1
a_{5j}	0	−9	−2	1	−2.8
a_{6j}	2	0	−4	0	0
a_{7j}	−1	−1	−1	−1	−1
a_{8j}	−1	−2	−3	−2	−1
a_{9j}	1	2	3	4	5
a_{10j}	1	1	1	1	1

DOI: 10.1002/nme

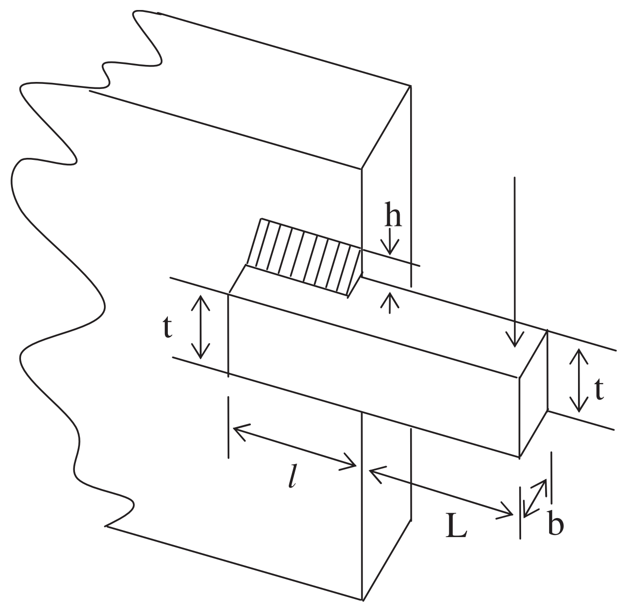

Figure A1. The welded beam design problem.

The optimum solution is $\mathbf{x} = (1.669913e{-}17; 3.953782e{-}16; 3.945990; 1.060366e{-}16; 3.283177; 10.000000; 1.128294e{-}17; 1.202619e{-}17; 2.507063e{-}15; 2.246241e{-}15; 0.370765; 0.278456; 0.523838; 0.388620; 0.298157)$ where $f(\mathbf{x}^*) = 32.655593$.

(18) g24

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = -x_1 - x_2 \\ \text{subject to} \quad & g_1(\mathbf{x}) = -2x_1^4 + 8x_1^3 - 8x_1^2 + x_2 - 2 \leqslant 0 \\ & g_2(\mathbf{x}) = -4x_1^4 + 32x_1^3 - 88x_1^2 + 96x_1 + x_2 - 36 \leqslant 0 \end{aligned}$$

where $0 \leqslant x_1 \leqslant 3$ and $0 \leqslant x_2 \leqslant 4$.

The optimum solution is $\mathbf{x}^* = (2.329520, 3.178493)$ where $f(\mathbf{x}^*) = -5.508013$.

(19) Welded beam design

A welded beam (see Figure A1) designed for the minimum cost is subject to constraints on shear stress (τ), bending stress in the beam (σ), buckling load on the bar (P_c), end deflection of the beam (δ) and side constraints. The four continuous design variables are thickness of the weld $h(x_1)$, the length of the welded joint $l(x_2)$, the width of the beam $t(x_3)$, and the thickness of the beam $b(x_4)$.

The problem can be mathematically formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2) \\ \text{subject to} \quad & g_1(\mathbf{x}) = \tau(\mathbf{x}) - \tau_{\max} \leqslant 0, \quad g_2(\mathbf{x}) = \sigma(\mathbf{x}) - \sigma_{\max} \leqslant 0, \quad g_3(\mathbf{x}) = x_1 - x_4 \leqslant 0 \\ & g_4(\mathbf{x}) = \delta(\mathbf{x}) - \delta_{\max} \leqslant 0, \quad g_5(\mathbf{x}) = P - P_c(\mathbf{x}) \leqslant 0 \end{aligned}$$

DOI: 10.1002/nme

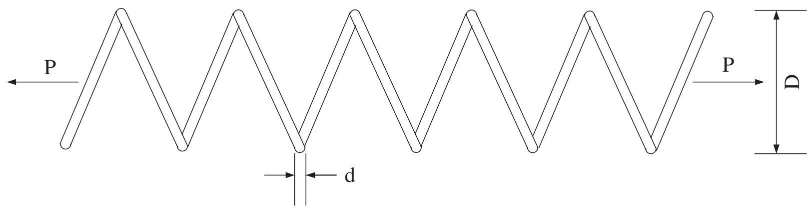

Figure A2. The tension/compression spring design problem.

The other parameters are defined as follows:

$$\tau(\mathbf{x}) = \sqrt{(\tau')^2 + \frac{2\tau'\tau'' x_2}{2R} + (\tau'')^2}, \quad \tau' = \frac{P}{\sqrt{2}x_1x_2}, \quad \tau'' = \frac{MR}{J}, \quad M = P\left(L + \frac{x_2}{2}\right)$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1+x_3}{2}\right)^2}, \quad J = 2\left\{\frac{x_1x_2}{\sqrt{2}}\left[\frac{x_2^2}{12} + \left(\frac{x_1+x_3}{2}\right)^2\right]\right\}, \quad \sigma(\mathbf{x}) = \frac{6PL}{x_4x_3^2}$$

$$\delta(\mathbf{x}) = \frac{4PL^3}{Ex_4x_3^3}, \quad P_c(\mathbf{x}) = \frac{4.013\sqrt{EGx_3^2x_4^6/36}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

where $P = 6000\,\text{lb}$, $L = 14$, $\delta_{\max} = 0.25$ in, $E = 30\times 10^6$ psi, $G = 12\times 10^6$ psi, $\tau_{\max} = 13600$ psi, $\sigma_{\max} = 30000$ psi, $0.125 \leqslant x_1 \leqslant 10.0$, $0.1 \leqslant x_2 \leqslant 10.0$, $0.1 \leqslant x_3 \leqslant 10.0$, and $0.1 \leqslant x_4 \leqslant 10.0$.

(20) A tension/compression spring design

This problem (see Figure A2) needs to minimize the weight ($f(\mathbf{x})$) of a tension/compression spring design subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter and on design variables. The design variables are the mean coil diameter $D(x_1)$, the wire diameter $d(x_2)$, and the number of active coils $p(x_3)$.

The problem can be mathematically formulated as follows:

$$\text{Minimize} \quad f(\mathbf{x}) = (x_3+2)x_1x_2^2$$

$$\text{subject to} \quad g_1(\mathbf{x}) = 1 - \frac{x_1^3x_3}{71\,785x_2^4} \leqslant 0, \quad g_2(\mathbf{x}) = \frac{4x_1^2 - x_1x_2}{12\,566(x_1x_2^3 - x_2^4)} + \frac{1}{5108x_2^2} - 1 \leqslant 0$$

$$g_3(\mathbf{x}) = 1 - \frac{140.45x_2}{x_1^2x_3} \leqslant 0, \quad g_4(\mathbf{x}) = \frac{x_1+x_2}{1.5} - 1 \leqslant 0$$

where $0.25 \leqslant x_1 \leqslant 1.3, 0.05 \leqslant x_2 \leqslant 2.0$, and $2 \leqslant x_3 \leqslant 15$.

(21) Speed reducer design

The weight of the speed reducer is to be minimized subject to constraints on bending stress of the gear teeth, surface stress, transverse deflections of the shafts, and stresses in the shafts. The variables $x_1, \ldots, x_7$ are the face width, module of teeth, number of teeth in the pinion, length of the first shaft between bearings, length of the second shaft between bearings, and the diameter of the first and second shafts.

DOI: 10.1002/nme

The problem can be mathematically formulated as follows:

$$\text{Minimize} \quad f(\mathbf{x}) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2)$$

$$\text{subject to} \quad g_1(\mathbf{x}) = \frac{27}{x_1x_2^2x_3} - 1 \leqslant 0, \quad g_2(\mathbf{x}) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \leqslant 0, \quad g_3(\mathbf{x}) = \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \leqslant 0$$

$$g_4(\mathbf{x}) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leqslant 0, \quad g_5(\mathbf{x}) = \frac{\left[\left(\dfrac{745x_4}{x_2x_3}\right)^2 + 16.9 \times 10^6\right]^{1/2}}{110.0x_6^3} - 1 \leqslant 0$$

$$g_6(\mathbf{x}) = \frac{\left[\left(\dfrac{745x_5}{x_2x_3}\right)^2 + 157.5 \times 10^6\right]^{1/2}}{85.0x_7^3} - 1 \leqslant 0$$

$$g_7(\mathbf{x}) = \frac{x_2x_3}{40} - 1 \leqslant 0, \quad g_8(\mathbf{x}) = \frac{5x_2}{x_1} - 1 \leqslant 0, \quad g_9(\mathbf{x}) = \frac{x_1}{12x_2} - 1 \leqslant 0$$

$$g_{10}(\mathbf{x}) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leqslant 0$$

$$g_{11}(\mathbf{x}) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leqslant 0$$

where $2.6 \leqslant x_1 \leqslant 3.6$, $0.7 \leqslant x_2 \leqslant 0.8$, $17 \leqslant x_3 \leqslant 28$, $7.3 \leqslant x_4 \leqslant 8.3$, $7.3 \leqslant x_5 \leqslant 8.3$, $2.9 \leqslant x_6 \leqslant 3.9$, and $5.0 \leqslant x_7 \leqslant 5.5$.

(22) Three-bar truss design

This problem is to deal with the design of a three-bar truss structure in which the volume is to be minimized subject to stress constraints.

The problem can be mathematically formulated as follows:

$$\text{Minimize} \quad f(\mathbf{x}) = (2\sqrt{2}x_1 + x_2) \times l$$

$$\text{subject to} \quad g_1(\mathbf{x}) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leqslant 0, \quad g_2(\mathbf{x}) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leqslant 0$$

$$g_3(\mathbf{x}) = \frac{1}{x_1 + \sqrt{2}x_2}P - \sigma \leqslant 0$$

where $0 \leqslant x_1 \leqslant 1, 0 \leqslant x_2 \leqslant 1, l = 100\,\text{cm}, P = 2\,\text{KN/cm}^2$, and $\sigma = 2\,\text{KN/cm}^2$.

(23) Pressure vessel design problem

The objective is to minimize the total cost ($f(\mathbf{x})$), including the cost of the material, forming, and welding. A cylindrical vessel is capped at both ends by hemispherical heads as shown in Figure A3. There are four design variables: T_s (x_1, thickness of the shell), T_h (x_2, thickness of the head), R (x_3, inner radius), and L (x_4, length of the cylindrical section of the vessel, not including

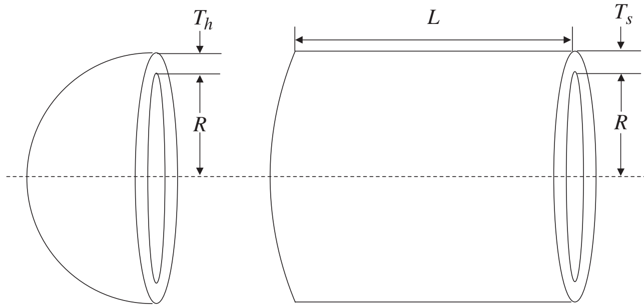

Figure A3. Center and end section of pressure vessel design problem.

the head). T_s and T_h are integer multiples of 0.0625 in, which are the available thicknesses of rolled steel plates, and R and L are continuous variables.

The problem can be mathematically formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \\ \text{subject to} \quad & g_1(\mathbf{x}) = -x_1 + 0.0193x_3 \leqslant 0, \quad g_2(\mathbf{x}) = -x_2 + 0.00954x_3 \leqslant 0 \\ & g_3(\mathbf{x}) = -\pi x_3^2 x_4 - 4/3\pi x_3^3 + 1296000 \leqslant 0, \quad g_4(\mathbf{x}) = x_4 - 240 \leqslant 0 \end{aligned}$$

where $0.0625 \leqslant x_1 \leqslant 6.1875$, $0.0625 \leqslant x_2 \leqslant 6.1875$, $10 \leqslant x_3 \leqslant 200$, and $10 \leqslant x_4 \leqslant 200$.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their constructive and helpful comments and suggestions.

This work was supported in part by the National Basic Scientific Research Founds under Grant A1420060159, and in part by the National Natural Science Foundation of China under Grants 60673062 and 60805027.

REFERENCES

1. Michalewicz Z, Schoenauer M. Evolutionary algorithm for constrained parameter optimization problems. *Evolutionary Computation* 1996; **4**(1):1–32.
2. Coello CA. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**(11–12):1245–1287.
3. Venkatraman S, Yen GG. A generic framework for constrained optimization using genetic algorithms. *IEEE Transactions on Evolutionary Computation* 2005; **9**(4):424–435.
4. Farmani R, Wright JA. Self-adaptive fitness formulation for constrained optimization. *IEEE Transactions on Evolutionary Computation* 2005; **7**(5):445–455.
5. Tessema B, Yen GG. A self adaptive penalty function based algorithm for constrained optimization. *Proceedings of the 2006 IEEE Congress on Evolutionary Computation* (*CEC'2006*), Piscataway, NJ. IEEE Service Center: Vancouver, BC, Canada, July 2006; 950–957.
6. Deb K. An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering* 2000; **186**(2–4):311–338.
7. Runarsson TP, Yao X. Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation* 2000; **4**(3):284–294.

8. Mezura-Montes E, Coello Coello CA. A simple multi-membered evolution strategy to solve constrained optimization problems. *IEEE Transactions on Evolutionary Computation* 2005; **9**(1):1–17.
9. Takahama T, Sakai S. Constrained optimization by applying the α constrained method to the nonlinear simplex method with mutations. *IEEE Transactions on Evolutionary Computation* 2005; **9**(5):437–451.
10. Ray T, Tai K, Seow KC. Multiobjective design optimization by an evolutionary algorithm. *Engineering Optimization* 2001; **33**(4):399–424.
11. Kurpati A, Azarm S, Wu J. Constraint handling improvements for multiobjective genetic algorithms. *Structural and Multidisciplinary Optimization* 2002; **23**(3):204–213.
12. Zhou Y, Li Y, He J, Kang L. Multi-objective and MGG evolutionary algorithm for constrained optimization. *Proceedings of the 2003 IEEE Congress on Evolutionary Computation* (*CEC'2003*), vol. 1, Piscataway, NJ. IEEE Service Center: Canberra, Australia, 2003; 1–5.
13. Cai Z, Wang Y. A multi-objective optimization-based evolutionary algorithm for constrained optimization. *IEEE Transactions on Evolutionary Computation* 2006; **10**(6):658–675.
14. Coello Coello CA. Constraint handling using an evolutionary multi-objective optimization technique. *Civil Engineering and Environmental Systems* 2000; **17**(4):319–346.
15. Fonseca CM, Fleming PJ. Multi-objective optimization and multiple constraint handling with evolutionary algorithms—part I: a unified formulation. *IEEE Transactions on Systems*, *Man and Cybernetics*, *A*, *Systems and Humans* 1999; **28**(1):6–37.
16. Coello Coello CA. Treating constraints as objectives for single-objective evolutionary optimization. *Engineering Optimization* 2000; **32**(3):275–308.
17. Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*, Grefenstette JJ (ed.). Hillsdale, NJ, 1985; 93–100.
18. Coello Coello CA, Mezura-Montes E. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Advanced Engineering Informatics* 2002; **16**(3):193–203.
19. Horn J, Nafpliotis N, Goldberg DE. A niched Pareto genetic algorithm for multiobjective optimization. *Proceedings of the First IEEE Conference on Evolutionary Computation*, *IEEE World Congress on Computational Intelligence*, vol. 1, Piscataway, NJ. IEEE Service Center: Canberra, Australia, June 1994; 82–87.
20. Aguirre AH, Rionda SB, Coello Coello CA, Lizáraga GL, Mezura-Montes E. Handling constraints using multi-objective optimization concepts. *International Journal for Numerical Methods in Engineering* 2004; **59**(15): 1989–2017.
21. Knowles JD, Corne DW. Approximating the non-dominated front using the Pareto archived evolutionary strategy. *Evolutionary Computation* 2000; **8**(2):149–172.
22. Mezura-Montes E, Coello Coello CA. Constrained optimization via multiobjective evolutionary algorithms. *Multi-Objective Problem Solving from Nature*: *From Concepts to Applications*, Knowles J, Corne D, Deb K (eds). Natural Computing Series. Springer: Berlin, 2008; 53–75.
23. Wang Y, Cai Z, Zhou Y, Zeng W. An adaptive trade-off model for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation* 2008; **12**(1):80–92.
24. Wang Y, Cai Z, Zhou Y, Fan Z. Constrained optimization based on hybrid evolutionary algorithm and adaptive constraint-handling technique. *Structural and Multidisciplinary Optimization* 2008; DOI: 10.1007/s00158-008-0238-3. Available from: http://www.springerlink.com/content/y57047081x0l4qm5/?p=ca39053a84174b99909b96eafa5c365a&pi=7.
25. Koziel S, Michalewicz Z. Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. *Evolutionary Computation* 1999; **7**(1):19–44.
26. Liang JJ, Runarsson TP, Mezura-Montes E, Clerc M, Suganthan PN, Coello Coello CA, Deb K. Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization. *Technical Report*, 2006.
27. Ray T, Liew KM. Society and civilization: an optimization algorithm based on the simulation of social behavior. *IEEE Transactions on Evolutionary Computation* 2003; **7**(4):386–396.
28. Kannan BK, Kramer SN. An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. *Journal of Mechanical Design* 1994; **116**:318–320.
29. Huang F, Wang L, He Q. An effective co-evolutionary differential evolution for constrained optimization. *Applied Mathematics and Computation* 2007; **186**(1):340–356.
30. He Q, Wang L. A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization. *Applied Mathematics and Computation* 2007; **186**(2):1407–1422.

DOI: 10.1002/nme

31. He Q, Wang L. An effective co-evolutionary particle swarm optimization for constrained engineering design problems. *Engineering Applications of Artificial Intelligence* 2007; **20**(1):89–99.
32. He S, Prempain E, Wu QH. An improved particle swarm optimizer for mechanical design optimization problems. *Engineering Optimization* 2004; **36**(5):585–605.
33. Wang J, Yin Z. A ranking selection-based particle swarm optimizer for engineering design optimization problems. *Structural and Multidisciplinary Optimization* 2008; DOI: 10.1007/s00158-007-0222-3. Available from: http://www.springerlink.com/content/qt180wq62h232874/?p=bfb79f1ff14245c48a0085413ba389ee&pi=16.
34. Muñoz-Zavala AE, Hernández-Aguirre A, Villa-Diharce ER, Botello-Rionda S. PESO+for constrained optimization. *Proceedings of the 2006 IEEE Congress on Evolutionary Computation* (*CEC'2006*), Piscataway, NJ. IEEE Service Center: Vancouver, BC, Canada, July 2006; 935–942.