

A genetic algorithm and a particle swarm optimizer hybridized with Nelder–Mead simplex search

Shu-Kai S. Fan ^{a,*}, Yun-Chia Liang ^a, Erwie Zahara ^b

^a *Department of Industrial Engineering and Management, Yuan Ze University, Chung-Li, Taoyuan County 320, Taiwan, ROC*

^b *Department of Industrial Engineering and Management, St. John's and St. Mary's Institute of Technology, Tamsui, Taipei County 251, Taiwan, ROC*

Received 11 May 2004; received in revised form 24 November 2004; accepted 4 January 2005

Available online 1 August 2006

Abstract

This paper integrates Nelder–Mead simplex search method (NM) with genetic algorithm (GA) and particle swarm optimization (PSO), respectively, in an attempt to locate the global optimal solutions for the nonlinear continuous variable functions mainly focusing on response surface methodology (RSM). Both the hybrid NM–GA and NM–PSO algorithms incorporate concepts from the NM, GA or PSO, which are readily to implement in practice and the computation of functional derivatives is not necessary. The hybrid methods were first illustrated through four test functions from the RSM literature and were compared with original NM, GA and PSO algorithms. In each test scheme, the effectiveness, efficiency and robustness of these methods were evaluated via associated performance statistics, and the proposed hybrid approaches prove to be very suitable for solving the optimization problems of RSM-type. The hybrid methods were then tested by ten difficult nonlinear continuous functions and were compared with the best known heuristics in the literature. The results show that both hybrid algorithms were able to reach the global optimum in all runs within a comparably computational expense.

© 2006 Elsevier Ltd. All rights reserved.

Keywords: Nelder–Mead simplex method; Genetic algorithm; Particle swarm optimization; Response surface methodology

1. Introduction

Global optimization of a continuous variable function has been the focus of nonlinear programming (NLP) over decades. This type of optimization problem searches for the global optimum of a continuous function given a search domain and avoids being trapped into one of the local optima. Function optimization has been applied to many areas in physical sciences, such as root finding of polynomials or system of equations, and estimation of the parameters of nonlinear functions (Nash & Sofer (1996)). Among all applications, response surface methodology (RSM) owns its importance to the practitioners. Therefore, the primary goal of this

* Corresponding author. Fax: +88 63 4638907.

E-mail address: simonfan@saturn.yzu.edu.tw (Shu-Kai S. Fan).

paper is at how to anchor an optimum operating setting for nonlinear RSM problems under the assumption that the designed-experiment and empirical model-building phase were carried out without undue difficulty. Secondly, the proposed algorithms are tested by other difficult nonlinear continuous functions to observe its speed of convergence and robustness to instances.

RSM was first espoused by [Box and Wilson \(1951\)](#) and consolidated until the advent of classic textbooks by [Myers \(1971\)](#) and [Box and Draper \(1987\)](#). Most recently, [Myers and Montgomery \(2002\)](#) provide an updated volume particularly useful for the practitioners. This method is a collection of mathematical and statistical techniques that are used in an earlier stage to construct empirical models, and then to optimize the predicted response surface inside a compact region of experimentation. Conducting designed experiments and applying regression analysis is to examine the relationship between the response variable of the process and a set of influential input variables. Based on the fitted model previously obtained, the optimum operating conditions can be “sequentially” estimated by using appropriate search techniques. In conventional RSM work, the method of steepest ascent is recommended for hill-climbing as the first-order response surface model is fit. For situations where the second-order model is adequate, the behavior of the “unconstrained” stationary point can be studied by reducing the estimated response surface into its canonical form, and the corresponding “constrained” stationary point can be computed and analyzed by using the ridge analysis technique ([Draper \(1963\)](#), & [Fan \(2003a, 2003b\)](#)).

RSM has been widely used in many industrial applications (see, e.g., [Nybergh, Marjamaki, & Skarp \(1997\)](#), [D'Angelo, Gastaldi, & Levialdi \(1998\)](#), [Scotti, Malik, Cheung, & Nelder \(2000\)](#), among others). There is considerable, practical experience shown in the literature indicating that the second-order model works very well in solving real RSM problems. However, many of state-of-the-art processes (such as in semiconductor manufacturing) often involve the system nonlinearities of much more complex than quadratic, perhaps with some specific location where the functional form of a physical model might be non-differentiable or even discontinuous due to technological limitations. By definition, a model is said to be “nonlinear” if at least one of its unknown parameters appears in a nonlinear functional form. In [Khuri and Cornell \(1996\)](#), there is a full chapter to discuss the nonlinear response surface models, including nonlinear least squares estimation, tests of hypotheses and confidence intervals, numerical examples, and parametric designs and parameter-free designs (such as the Lagrangian interpolation of the mean response) for nonlinear regression models. PROC NLIN in [SAS \(1996\)](#) provides a powerful procedure to compute least squares or weighted least squares estimates of the parameters of a nonlinear regression model. The PROC NLIN uses one of the following five iterative methods for nonlinear least squares estimation: steepest-descent or gradient method, Newton method, modified Gauss–Newton method, Marquardt method, and multivariate secant or false position method. Regarding the designs for nonlinear regression models, the earliest and most commonly used design criterion is attributed to [Box and Lucas \(1959\)](#). Box and Lucas’ approach is to locate those points as a design that minimizes the generalized variance of the least squares estimators of the unknown parameters in the approximating linear model in terms of a Taylor’s expansion polynomial in the region of experimentation.

For those nonlinear optimization situations mentioned above, one must resort to general numerical optimization algorithms for process optimization. Thus far, optimization methods can generally be classified into three broad categories: gradient-based techniques (e.g., the model trust region algorithm, the quasi-Newton method, etc.), direct search techniques (e.g., the method of Hooke and Jeeves, the method of Rosenbrock, etc.) and meta-heuristic techniques (e.g., genetic algorithm, evolutionary strategy, etc.). To circumvent the deficiency in the gradient-based methods caused by non-differentiability, our attention is turned on the other two kinds of optimization methods. In this research, a study has been carried out to exploit the hybrid schemes combining a local search technique (Nelder–Mead simplex search method (NM, [Nelder & Mead \(1965\)](#))) with two most recent developed meta-heuristic methods ([Sondergeld & Voss \(1999\)](#)) genetic algorithms (GA, see [Goldberg \(1998\)](#)) and particle swarm optimization (PSO, see [Kennedy & Eberhart \(1995\)](#)), respectively, for the determination of a set of operating conditions that optimize the RSM problems. In fact, we do not intend to diminish the existing RSM optimization techniques in any way but instead would like to provide the RSM practitioners with an additional search tool while dealing with very complicated nonlinear response surface functions.

One of the main obstacles in applying meta-heuristic methods to complex problems has often been the high computational cost due to slow convergence as they do not utilize much local information to determine the

most promising search direction. Since NM has already been successfully embedded into the meta-heuristics acting as a “local hill-climber” so as to improve the rate of convergence (see, e.g., Yen, Liao, Lee, & Randolph (1998) Fan, Liang, & Zahara (2004)), an idea to combine GA and PSO with NM for the purpose of response surface exploration is therefore motivated. The rationale behind such a hybrid approach would be to include the merits of GA and PSO with those of NM such that these hybrid heuristics could exploit a better tradeoff between computational efforts and global optimality of the solution attained.

The remainder of the paper is organized as follows. Section 2 briefly reviews the fundamentals of GA, PSO and NM. Section 3 presents the hybrid NM–GA and NM–PSO structures and algorithms. Sections 4 and 5 illustrate efficiency, accuracy and robustness of the hybrid approaches while solving response surface optimization problems and other more difficult, nonlinear continuous optimization problems by means of test functions from the literature, respectively. At last, major results of the paper are summarized in Section 6.

2. Backgrounds of NM, GA and PSO

2.1. Nelder–Mead simplex search method (NM)

The simplex search method, first proposed by Spendley, Hext, and Himsworth (1962) and later refined by Nelder and Mead (1965), is a derivative-free line search method that was particularly designed for traditional unconstrained minimization scenarios, such as the problems of nonlinear least squares, nonlinear simultaneous equations, and other types of function minimization (see, e.g., Olsson & Nelson (1975)). First, function values at the $(N + 1)$ vertices of an initial simplex are evaluated, which is a polyhedron in the factor space of N input variables. In the minimization case, the vertex with the highest function value is replaced by a newly reflected, better point, which would be approximately located in the negative gradient direction. Clearly, NM can be deemed as a direct line-search method of steepest descent kind. The ingredients of replacement process consist of four basic operations: reflection, expansion, contraction and shrinkage. Through these operations, the simplex can improve itself and come closer and closer to a local optimum point sequentially. Furthermore, the simplex can vary its shape, size and orientation to adapt itself to the local contour of the objective function, so NM is extremely flexible and suitable for exploring difficult terrains. Obviously, the arguments noted above parallel to the sequential nature of experimental optimization in RSM (Box & Wilson (1951)). Besides speedy convergence, the preceding statement confirms in part NM’s suitability for RSM optimization.

2.2. Genetic algorithm (GA)

The enlightenment of genetic algorithms (GA) was dated to the 1960s by Holland (1975) and further described by Goldberg (1998). GA is a stochastic global search technique that solves problems by imitating processes observed during natural evolution. Based on the survival and reproduction of the fitness, GA continually exploits new and better solutions without any pre-assumptions, such as continuity and unimodality. GA has been successfully applied to many complex optimization problems and shows its merits to traditional optimization methods, especially when the system under study has multiple optimum solutions.

GA evolves a population of candidate solutions. Each solution is represented by a chromosome that is usually coded as a binary string. The fitness of each chromosome is then evaluated using a performance function after the chromosome has been decoded. Upon completion of the evaluation, a biased roulette wheel is used to randomly select pairs of chromosomes to undergo genetic operations that mimic natural phenomena observed in nature (such as crossover and mutation). This evolution process continues until the stopping criteria are reached.

A real-coded GA is a genetic algorithm representation that uses a vector of floating-point numbers to encode the chromosome. The crossover operator of a real-coded GA is performed by borrowing the concept of convex combination of vectors. The random mutation operator proposed for real-coded GA is to change the gene with a random number in the problem’s domain. With some modifications of genetic operators, real-coded GA has resulted in better performance than binary coded GA for continuous problems (see Janikow & Michalewicz (1991)).

2.3. Particle swarm optimization (PSO)

Particle swarm optimization (PSO) is one of the latest meta-heuristic techniques developed by Kennedy and Eberhart (1995). PSO concept is based on a metaphor of social interaction such as bird flocking and fish schooling. The potential solutions, called “particles”, in PSO algorithm fly around in the multidimensional search space and the positions of individual particles are adjusted according to its own flying experience, i.e., previous best, and its companions’ flying experiences, i.e., neighborhood best or global best. The major difference between PSO and other so-called “evolutionary-type” algorithms such as GA is that PSO does not implement survival of the fitness, since all particles in PSO are kept as members of the population through the course of the searching process. As simple and economic in concept and computational cost, PSO has been shown to successfully optimize a wide range of continuous optimization problems (see Yoshida, Kawata, Fukuyama, Takayama, & Nakanishi (2000) & Brandstatter & Baumgartner (2002)).

3. Proposed hybrid algorithms

The hybrid idea behind the methods introduced in Section 2 is to combine their advantages and avoid disadvantages. Similar ideas have been discussed in hybrid methods using GA and direct search technique, and they emphasized the trade-off between solution quality, reliability and computation time in global optimization (Renders & Flasse (1996) & Yen et al. (1998)). This section starts from recollecting the procedures of NM (Section 3.1), GA (Section 3.2) and PSO (Section 3.3), respectively, that will be used for computational comparison purposes. The origins and literatures of these algorithms can be found in Section 2. It then introduces the hybrid methods and also demonstrates that the convergence of NM and the accuracy of PSO can be improved simultaneously by adopting a few straightforward modifications. It should be noted again that our focus is still placed on the synergy of NM and evolutionary heuristics.

3.1. The procedure of NM

An example of the function minimization of two variables will illustrate the basic procedure of NM. Starting point B together with initial step sizes will construct an initial simplex design (shown as A, B and C), as illustrated in Fig. 1. Suppose $f(A)$ owns the highest function value among these three points and is to be replaced. In this case, a reflection is made through the centroid of BC (the midpoint D) to the point E. Suppose $f(C) < f(B) < f(A)$. At this stage, three situations can arise.

1. If $f(E) < f(C)$, an extension is made to point J. We then keep E or J as a replacement for A that depends which function value is lower.
2. If $f(E) > f(C)$, a contraction is made to point G or H depending on which of $f(A)$ or $f(E)$ is lower.
3. If $f(G)$ or $f(H)$ is higher than $f(C)$, the contraction fails and then we perform shrinkage operation. The shrinkage operation reduces the size of the simplex by moving all but the best point C halfway towards the best point C. The algorithm then evaluates function values at each vertex and returns to the reflection step to start a new iteration. The stopping criterion of NM will be introduced in Section 4.

In the original NM simplex search method, the initial simplex constructed is a right-angled triangle for rectangular coordinates, but this characteristic is rapidly destroyed as the successive iterations introduce four fundamental operations; i.e., reflection, expansion, contraction and shrinkage. The distribution of the $(N + 1)$ vertex points of the initial simplex determines its “shape”, which may vary widely. Based on the investigation of variants on the NM simplex method reported by Parkinson and Hutchinson (1972), the experimental results showed that the shape and step size of the NM initial simplex are relatively unimportant. In our hybrid methods to be shown shortly, the initial population of random generation by GA and PSO will make the effect of step size used in the NM part far less critical. By the above two reasons, we fix the step size at 1.0 during hybridization in this study.

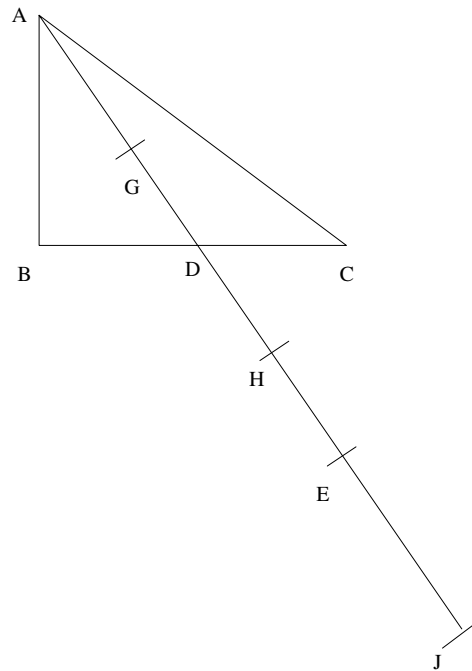


Fig. 1. A two-dimensional NM operations.

3.2. The procedure of GA

The procedure of GA is to make a population of individuals evolve according to a replica of Darwinian theory. The initial population consists of one predetermined starting point and $5N - 1$ randomly generated chromosomes which are coded using floating-point numbers. Each chromosome represents a complete solution and contains the information of a set of independent variables. For this GA, parents are selected based on the ordinal ranking of their performance function, i.e., the response variable in RSM. The offspring are created by the application of genetic operators such as crossover and mutation. The main operator is originally motivated by the “reflection” operation in the Nelder–Mead simplex search procedure with the crossover probability of 100%. The random mutation operator is initially motivated by the “contraction” operation in the Nelder–Mead simplex search procedure with 30% probability. This process terminates when it satisfies the stopping criteria. Fig. 2 summarizes the procedure of GA algorithm. Note that all the GA parameter settings in Fig. 2 have been justified via previous simulations, including the population size $5N$ in **Initialization**, the random number sampling from $U(1.2, 2.2)$ in **Crossover**, and the mutation probability 0.3 and the random number sampling from $U(0.3, 0.7)$ in **Mutation**. For further details of complete analysis, see Zahara (2003) and Zahara and Fan (2004).

3.3. The procedure of PSO

Like GA, PSO is a population-based method. However, the major difference compared to GA is that it does not implement the filtering, i.e., all members in population survive through the entire search process. In addition, a commonly observed social behavior, where members of a group tend to follow the lead of the best of the group, is simulated by PSO. The procedure of PSO is illustrated as follows.

1. **Initialization.** Randomly generate a population of the potential solutions, called “particles”, and each particle is assigned a randomized velocity. For the computational experiments as will be conducted in Section 4, the first particle is the initial starting point chosen and the other $5N - 1$ particles are randomly generated for solving N -dimensional problems.

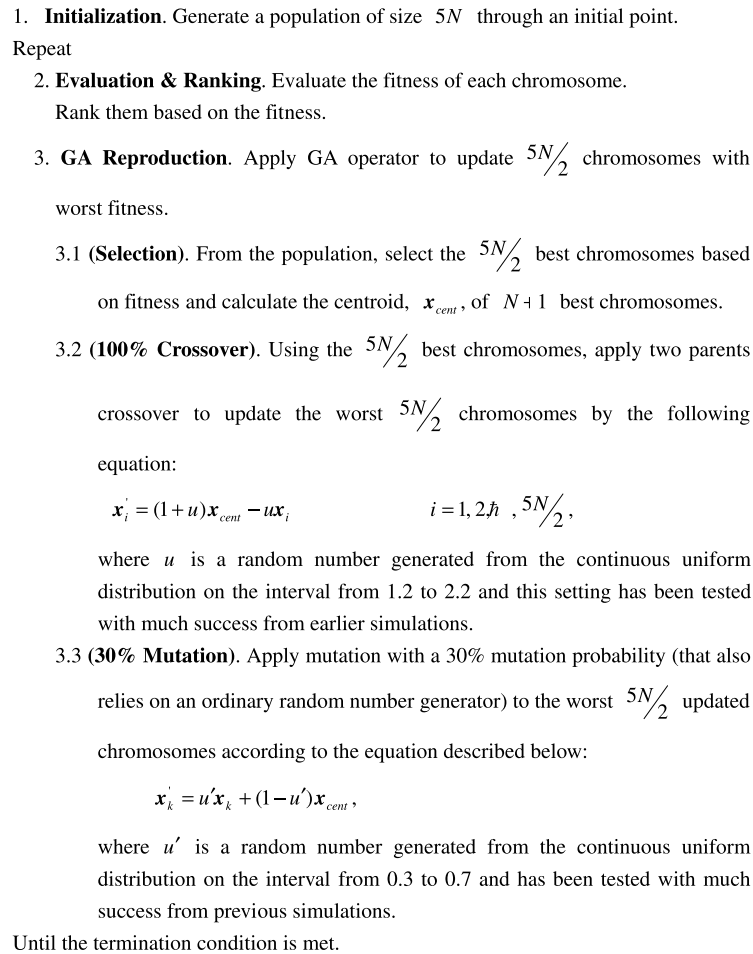


Fig. 2. The procedure of GA algorithm.

2. *Velocity update.* The particles then “fly” through hyperspace by updating their own velocity. The velocity update of a particle is dynamically adjusted, subject to its own past flight experience and those of its companions. The particle’s velocity and position are updated by the following equations:

$$V_{id}^{New} = w \times V_{id}^{old} + c_1 \times rand_1 \times (p_{id} - x_{id}^{old}) + c_2 \times rand_2 \times (p_{gd} - x_{id}^{old}), \quad (1)$$

$$x_{id}^{New} = x_{id}^{old} + V_{id}^{New}, \quad (2)$$

where c_1 and c_2 are two positive constants; w is an inertia weight and $rand_1$ and $rand_2$ are two independent random sequences generated from $U(0, 1)$. Eberhart and Shi (2001) and Hu and Eberhart (2001) suggested $c_1 = c_2 = 2$ and $w = [0.5 + (rand/2.0)]$. Eq. (1) illustrates the calculation of a new velocity for each individual. The velocity of each particle is updated according to its previous velocity (V_{id}), the personal (or previous) best location of the particle (p_{id}) and the global best location (p_{gd}). Particle’s velocities on each dimension are clamped to a maximum velocity V_{max} and the maximum velocity V_{max} is set to the range of the search space

in each dimension. Eq. (2) shows how each particle's position (x_{id}) is updated in the search of solution space. In general, two versions of PSO exist, called the global best and local best models (Kennedy & Shi, 2001). The major difference between these two models depends on the set of particles with which a given particle will interact directly. Individuals in PSO compare themselves to their neighbors on the critical measure and imitate only those neighbors who are superior to themselves. Eq. (1) corresponds to the global best model, and the local best model can be expressed by replacing p_{gd} in Eq. (1) with the local best position p_{id} . Note that the hybrid PSO algorithm proposed in this paper adopts a different structure of cognitive behavior, called the cluster best position, as can be seen in a later section.

3.4. Hybrid NM–GA

In an N -dimensional problem, the population size of this hybrid NM–GA approach is set to $2(N + 1)$. Each chromosome represents a solution and contains the information of a set of independent variables. In order to balance the deterministic exploitation of NM and stochastic exploration of GA, the initial population is constructed by two approaches. First, $N + 1$ solutions are generated using a predetermined starting point and a positive step size of 1.0 in each coordinate direction. For example, in a two-dimensional problem, if the point (1, 3) is assigned as the starting point, it will automatically determine two other points, (2, 3) and (1, 4), in each coordinate direction, respectively. Second, another half of population, i.e., $N + 1$ solutions, is randomly generated. The entire population is then sorted according to their fitness. In this work, the performance function to evaluate the fitness represents the response variable in RSM. The best N solutions are saved for subsequent use and the best $N + 1$ solutions are fed into the modified simplex search algorithm to improve the $(N + 1)$ th solution in the rank. Additionally, if the true optimal solution of an N -dimensional problem is very far away from the starting point, a second expansion operator in the modified simplex search algorithm may be appropriate to improve the convergence rate. This operator will apply only after the success of an expansion attempt, and the second expansion point is calculated by the following equation:

$$P_{\text{second exp}} = \tau P_{\text{exp}} + (1 - \tau)P_{\text{cent}}, \quad (3)$$

where τ is the second expansion coefficient ($\tau > 1$). The choice of $\tau = 2$ has been tested with much success from early computational experience. Fig. 3 summarizes the modified simplex search method.

Joined by the N best chromosomes and the $(N + 1)$ th chromosome, the $N + 1$ worst chromosomes are improved by the real coded GA method (i.e., selection, 100% crossover and 30% mutation). In the selection procedure, the $N + 1$ best chromosomes are chosen as parents. The arithmetical crossover acts as the main operator with the 100% probability, and the secondary operator is the random mutation using the mutation probability of 30%. The results are then sorted again in preparation for repeating the entire run. Fig. 4 depicts the schematic representation of the proposed hybrid NM–GA algorithm and the overview of this approach is summarized in Fig. 5.

1. From ranked population, select the top $N + 1$ solutions.
2. Attempt reflection. If the reflection is accepted then attempt expansion, else attempt contraction or replace the worst fitness point with the reflection point.
3. Attempt expansion. If the expansion is accepted then attempt second expansion, else replace the worst fitness point with the expansion point.
4. Attempt second expansion. If the second expansion is accepted then replace the worst fitness point with the second expansion point, else replace the worst fitness point with the first expansion point.
5. Attempt contraction. If the contraction is accepted then replace the worst fitness point with the contraction point, else attempt shrinking for the entire simplex (except the best fitness point).

Fig. 3. The modified simplex search algorithm.

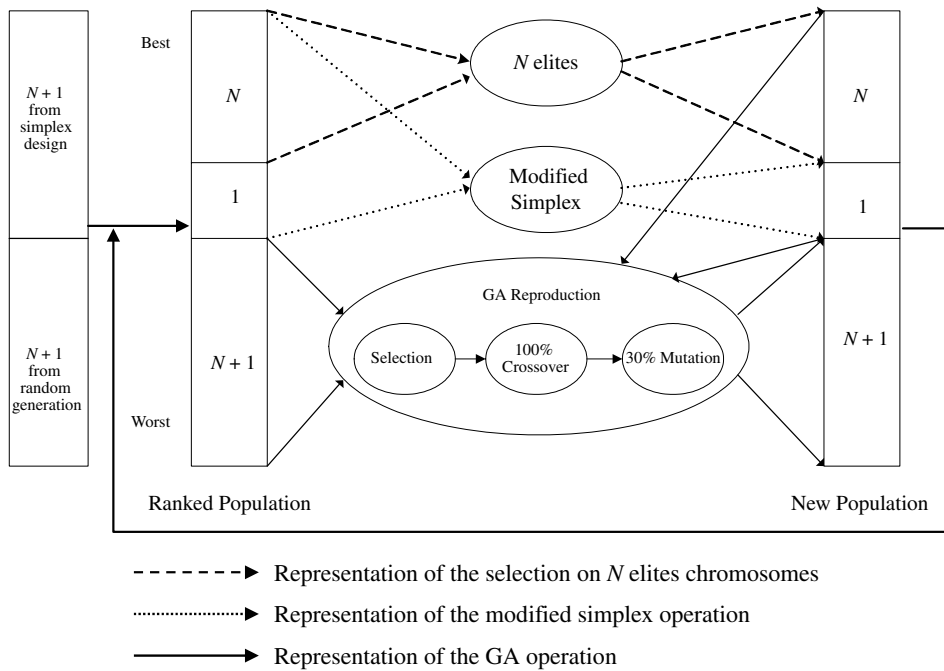


Fig. 4. Schematic representation of the NM-GA hybrid. [FX1] Representation of the selection on N elites chromosomes [FX2] Representation of the modified simplex operation [FX3] Representation of the GA operation.

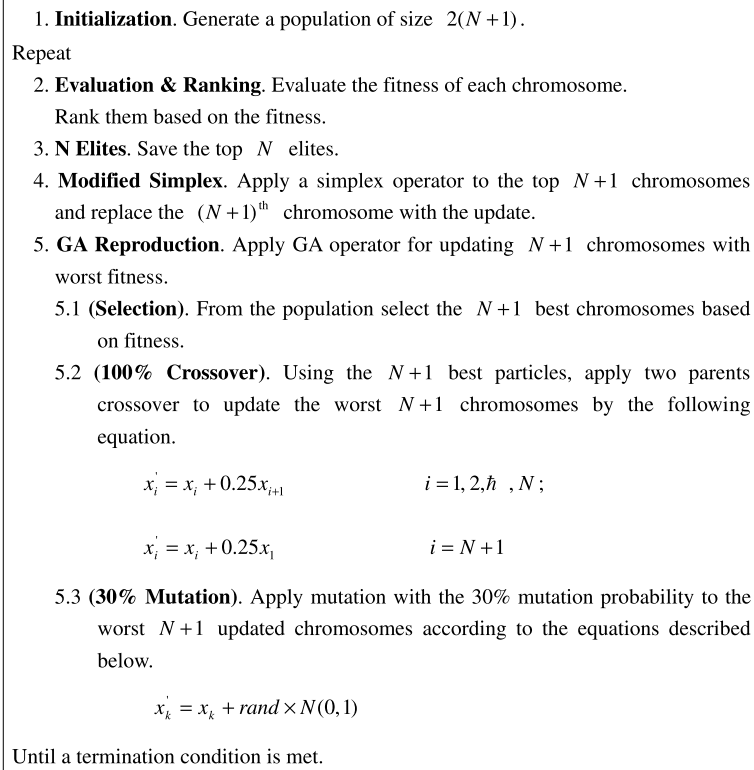


Fig. 5. The hybrid NM-GA algorithm.

3.5. Hybrid NM-PSO

The population size of this hybrid NM-PSO approach is set to $3N + 1$ when solving an N -dimensional problem. Similar to the hybrid NM-GA algorithm, the initial population is built by two approaches: $N + 1$ particles are constructed using the predetermined starting point and a positive step size of 1.0 in each coordinate direction, and the example of this approach has been demonstrated in the previous section. Other $2N$ randomly generated particles are divided into N pairs, i.e., one pair in each dimension. Each of these pairs is uniformly randomly generated corresponding to each dimension, and this approach can be illustrated as follows. In a two-dimensional problem, four points will be constructed, i.e., two in each dimension and the forms of these points can be represented as $(x_1, 0)$, $(x'_1, 0)$, $(0, x_2)$, $(0, x'_2)$, respectively.

A total of $3N + 1$ particles are sorted by the response values in RSM, and the best N particles are saved for subsequent use. The top $N + 1$ particles are then fed into the modified simplex search method to improve the $(N + 1)$ th particle and the scenario is the same with the one in Section 3.4. Joined by the N best particles and the $(N + 1)$ th particle, the worst $2N$ particles are adjusted by the modified PSO method. The procedure of the modified PSO algorithm mainly consists of selection, mutation for the global best particle and velocity update. The modified PSO method begins with the selection of the global best particle and the clustering best particles. The global best particle of the population is determined according to the sorted response values. The worst $2N$ particles are then divided into N clusters. Each cluster contains two particles. The particle with better response value in each cluster is selected as the clustering best particle. Velocity update in this hybrid PSO algorithm is performed as in Eq. (1) but the cluster best location (p_{cd}) is used to replace the personal best location of the particle (p_{id}). The illustration of selection operator is shown in Fig. 6.

In the preliminary research, it is realized that in the original PSO method the particles' velocity updates highly depend on the global best particle. The velocity update for the global best particle can generate, at most, a tiny jump for further improvement such that the entire population is very unlikely to pull themselves out of the local optimum. Thus, if the global best particle is trapped around a local optimum, then all the other particles will also fly toward that local optimum. In order to remedy this situation, a modified 2/5 success rule of mutation heuristic borrowed from evolutionary strategy (ES) is employed. The position of the global best particle is mutated five times using a normal distribution $N(0, \sigma)$. The value of σ is adjusted based on the

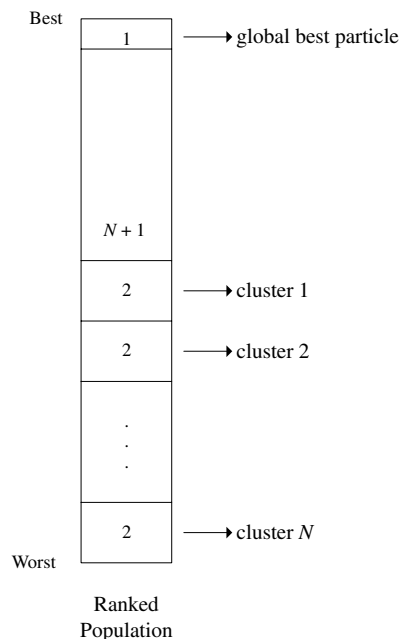


Fig. 6. The selection operator for the modified PSO.

Part I Selection. From the population select the global best particle and the clustering best particles.

Part II Mutation heuristic for the global best particle.

1. Define x^{old_gbest} as the global best particle, set the mutation success rate to $2/5$ and the mutation coefficient (λ) to 0.85 .
2. Generate 5 particles based on the position of the global best particle with variance σ according to the following equation

$$x_i^{new_gbest} = x^{old_gbest} + N(0, \sigma), \quad i=1, 2, 3, 4, 5.$$

3. For each mutation

{

if mutation success rate = $2/5$, then $\sigma^{new} = \sigma^{old}$;

if mutation success rate > $2/5$, then $\sigma^{new} = (1/\lambda)\sigma^{old}$;

if mutation success rate < $2/5$, then $\sigma^{new} = \lambda\sigma^{old}$.

}

4. Replace the old global best particle (x^{old_gbest}) with the new global best particle

found among the $x_i^{new_gbest}$.

Part III Velocity update. Apply velocity update to the $2N$ particles with worst fitness according to the original PSO method and the maximum velocity constrain (V_{max}) is ignored.

Fig. 7. Algorithmic representation of the modified PSO.

success (improvement) rate of five mutants. If the mutant outperforms the original global best particle, the global best particle is updated. Thereafter, the last step, velocity update, is performed based on the original PSO algorithm. The details of the modified PSO are described in Fig. 7.

The search process of this hybrid NM–PSO algorithm continues until a convergence stopping criterion is reached. Fig. 8 portrays the schematic representation of the proposed hybrid NM–PSO method and this algorithm is outlined in Fig. 9.

4. Computational experience of response surface optimization

In this section, the proposed hybrid methods will first be tested with four case studies for their applicability in RSM. The first three were described in Khoo and Chen (2001) and the last one was collected from MATLAB® toolboxes. Two of them (cases 1 and 4) belong to the nonlinear response surface models, rather than the first- and second-order regression models frequently used in response surface applications. A stopping criterion used here is based on the standard deviation of the objective function values over $N+1$ best solutions of the current population, as expressed by

$$S_f = \left[\sum_{i=1}^{N+1} (f(x_i) - \bar{f})^2 / (N+1) \right]^{1/2} < \varepsilon, \quad (4)$$

where $\bar{f} = \sum_{i=1}^{N+1} f(x_i) / (N+1)$ denotes the mean value of the objective function values over $N+1$ best solutions in the current population. In the four case studies, the tolerance $\varepsilon = 1 \times 10^{-7}$ is employed. The

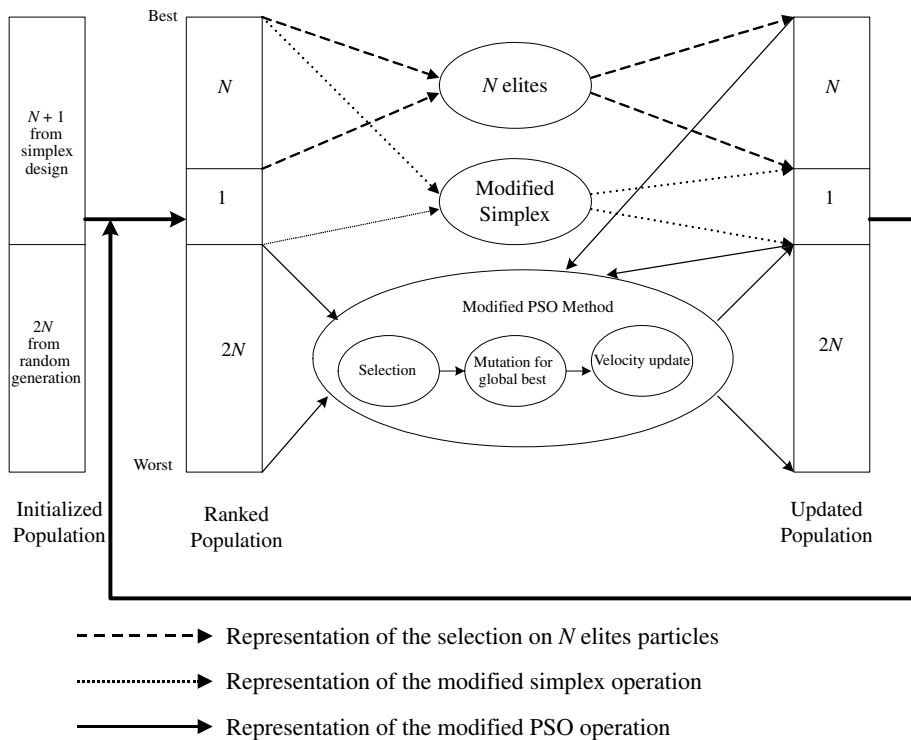


Fig. 8. Schematic representation of the NM-PSO hybrid. [FX4] Representation of the selection on N elites particles [FX5] Representation of the modified simplex operation [FX6] Representation of the modified PSO operation.

1. **Initialization.** Generate a population of size $3N + 1$.
- Repeat
 2. **Evaluation & Ranking.** Evaluate the fitness of each particle.
Rank them based on the fitness.
 3. **N Elites.** Save the top N elites.
 4. **Modified Simplex.** Apply a simplex operator to the top $N + 1$ particles and replace the $(N + 1)^{\text{th}}$ particle with the update.
 5. **Modified PSO.** Apply modified PSO operator for updating $2N$ particles with worst fitness.
- Until a termination condition is met.

Fig. 9. The simplex-PSO hybrid algorithm.

optimization task of GA, PSO, NM-GA and NM-PSO on each case study was repeated ten times. The population design for each algorithm is shown in Table 1. Only one run is generated for the NM algorithm because NM always converges to an identical optimum provided the same starting point. The decision on the number of simulation replications was made based on the limitation of computational time and computer disk space.

The algorithms were coded in MATLAB® 6.0 and run on a Pentium(R) IV 2.4 GHz PC with 512 MB memory. The performance measures contain the average global best solution (\bar{x}), average optimum response value (\bar{F}) and average number of iterations required ($\overline{\text{ITE}}$) over ten runs. The standard error statistics of the measures (S_x , S_F and S_{ITE}) are recorded as well. Before proceeding to the following optimization exercises, it needs to be emphasized that this research, however, primarily aims at how to search for an optimum operating setting for RSM problems under the assumption that the designed-experiment and model-building phase were carried out adequately.

Table 1

The populations design for the five algorithms

	NM	GA	PSO	NM-GA	NM-PSO
Population size	$N + 1$	$5N$	$5N$	$2N + 2$	$3N + 1$

4.1. Case study 1

Consider the Himmelblau function in Deb (1995), which is a two-dimensional, multimodal function. Suppose that the response function has been fitted to be

$$\hat{f} = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2 + 0.1 \times [(x_1 - 3)^2 + (x_2 - 2)^2], \quad (5)$$

where the optimization search is confined within the cuboidal region, $-6 \leq x_1 \leq 6$ and $-6 \leq x_2 \leq 6$. Among four local minima that are all inside the search boundary is the unique global minimum occurring at $x_1 = 3$ and $x_2 = 2$ with an optimal response value of zero (see the surface plot and the contour plot in Figs. 10 and 11). Other local minima take place at $(-2.7871, 3.1282)$, $(-3.7635, -3.2661)$ and $(3.5815, -1.8208)$ with response values of 3.4871, 7.3673 and 1.5044, respectively. Obviously, the predicted response function in Eq. (5) exhibits severer curvature than the quadratic model frequently encountered in common RSM practice, posing a challenging test-bed for response surface optimization. Hence, the objective of this case study is to try to locate the global optimum estimate of the “quartic” response surface as shown in (5) from various starting points.

The proposed hybrid methods were used to solve the Himmelblau function. To span extensive search space for achieving global optimality, the two hybrid methods would take the current global best particle as a pivot to generate additional candidate particles projected onto the remaining $2^N - 1$ quadrants. This special characteristic is designed for a check with a possible drastic mutation on the global best particle. For example in a two-dimensional case, if the global best particle so far is $(1, 1)$, then the hybrid methods will check $(-1, 1)$, $(1, -1)$ and $(-1, -1)$ to see whether the fitness in these three candidate points is better than the fitness of $(1, 1)$. If the fitness at point $(-1, 1)$ is better than the global best fitness, then the global best point will be mutated to $(-1, 1)$. Table 2 shows the optimization results returned by using the five methods while solving the Himmelblau function from five initial points (denoted by x_0), which are intentionally selected near the local minima.

As can clearly be seen from Table 2, NM always converges to the local minimum near the starting point, revealing its excessive dependency on starting points. The performance of GA and PSO indicates a mixture of some local and global optima over the ten runs. However, both hybrid methods could pinpoint the global optimum $(3, 2)$ with perfect accuracy (see S_x and S_F indicated in parenthesis) despite the choice of starting points. The hybrid NM-GA and NM-PSO methods converge quite fast to the global optimum (see the

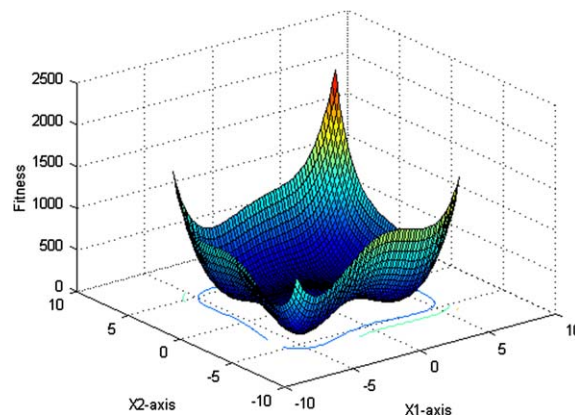


Fig. 10. The surface plot with contours of the Himmelblau function in case study 1.

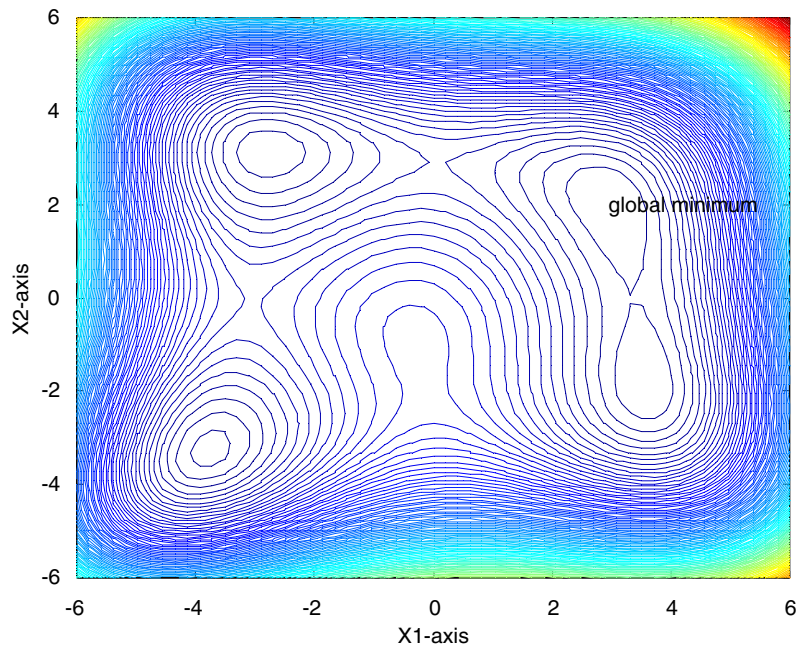


Fig. 11. The contour plot of the Himmelblau function in case study 1.

Table 2
Computational results on the Himmelblau function in case study 1

x_0	Method	$\bar{x} S_x$	$\bar{F} S_F$	$\overline{ITE} S_{ITE}$
(0, 0)	Optimal	3, 2	0	
	NM	2.9999, 2.0001	1.8837e-07	106
	GA	-0.6094, 1.0160 (3.2577, 2.6971)	3.0188 (2.7554)	51.2 (4.8259)
	PSO	2.4794, 1.7307 (1.8595, 1.2973)	0.4991 (1.1514)	71.3 (2.2136)
	NM-GA	3.0000, 2.0000 (0.0000, 0.0000)	2.6700e-08 (0.0000)	36.2 (2.8983)
(1, 1)	NM-PSO	3.0000, 2.0000 (0.0000, 0.0000)	2.8040e-08 (0.0000)	39.3 (11.1560)
	NM	3.0000, 2.0000	4.0806e-08	34
	GA	3.1744, 0.8538 (0.2809, 1.8456)	0.4513 (0.7267)	50.6 (9.9130)
	PSO	2.4213, 2.1128 (1.8300, 0.3568)	0.3487 (1.1027)	70.9 (4.9318)
	NM-GA	3.0000, 2.0000 (0.0000, 0.0000)	2.9787e-08 (0.0000)	33.7 (5.2715)
(-3, -3)	NM-PSO	3.0000, 2.0000 (0.0000, 0.0000)	2.9094e-08 (0.0000)	37.8 (5.7116)
	NM	-3.7635, -3.2661	7.3673	34
	GA	0.7620, 0.6656 (3.2854, 2.5285)	2.2342 (2.3005)	52.4 (6.0955)
	PSO	-1.2670, -0.9142 (3.3728, 2.8599)	4.4444 (2.6427)	77.4 (4.7188)
	NM-GA	3.0000, 2.0000 (0.0000, 0.0000)	3.3849e-08 (0.0000)	37 (4.6428)
(3, -1)	NM-PSO	3.0000, 2.0000 (0.0000, 0.0000)	3.6700e-08 (0.0000)	45.5 (9.0952)
	NM	3.5815, -1.8208	1.5044	31
	GA	1.5546, 0.4281 (3.0050, 2.4060)	1.7983 (1.3105)	47.5 (5.5428)
	PSO	3.2907, 0.0896 (0.3065, 2.0137)	0.7522 (0.7929)	73 (4.8534)
	NM-GA	3.0000, 2.0000 (0.0000, 0.0000)	4.0531e-08 (0.0000)	42.6 (12.2129)
(-2, 2)	NM-PSO	3.0000, 2.0000 (0.0000, 0.0000)	3.3263e-08 (0.0000)	36.9 (7.0781)
	NM	-2.7871, 3.1282	3.4871	31
	GA	1.2826, 0.9349 (3.0533, 2.3125)	1.7350 (2.4272)	50.9 (6.6072)
	PSO	2.4213, 2.1128 (1.8300, 0.3568)	0.3487 (1.1027)	72.4 (4.1419)
	NM-GA	3.0000, 2.0000 (0.0000, 0.0000)	3.5336e-08 (0.0000)	39.2 (6.6131)
	NM-PSO	3.0000, 2.0000 (0.0000, 0.0000)	3.3530e-08 (0.0000)	36.2 (12.2547)

performance statistics of \overline{ITE} and S_{ITE} in Table 1); the number of iterations required ranges approximately from 33 to 45. It should be noted that the GA result in Khoo and Chen (2001) also confirmed the global optimum at (3, 2), yet which was achieved at the 5825th iteration with the population size of 40. As evidenced by the computational results given here, the proposed hybrids show a great deal of promise as an optimization toolkit with higher accuracy, efficiency and robustness in solution quality for “complex” RSM problems.

4.2. Case study 2

The second case study is a typical, single-objective RSM problem discussed in Khoo and Chen (2001), where the bonding process in a microelectronics company was investigated. The measured response is the pull strength, with which in destructive testing the bonded leads are to be detached from the die. The mean pull strength is the average of numerous individual pull strengths of the leads in each component. The goal was to maximize the mean pull strength with three influencing variables: temperature, force and time. We follow Khoo and Chen’s function, the fitted response surface is

$$\hat{f}(x) = 73.89 + 12.91x_1 + 7.11x_2 + 2.56x_3 - 1.96x_1^2 - 1.01x_2^2 + 0.022x_3^2 + 0.36x_1x_2 - 0.068x_1x_3 - 0.52x_2x_3, \quad (6)$$

where \hat{f} is the predicted mean pull strength response (in gram force); x_1 is the bonding temperature (in °C) within the range [500, 580]; x_2 is the bonding force (in kg) within the range [11, 13]; x_3 is the bonding time (in ms) within the range [210, 250]. Apparently, the predicted response surface in (6) is a typical RSM optimization problem in a quadratic polynomial format. If \hat{f} is not a concave function or the stationary point is situated outside the allowable region, the constrained maximum point frequently resides at the perimeter of the feasible region. Note that, according to the coding convention in the factorial design, x_1 – x_3 in (6) stand for coded variables between -1 and 1 . Since the problem belongs to constrained optimization, thus all the algorithms include a mechanism to check constraint violation during optimization process.

The proposed hybrid methods were used to optimize the single response problem and the results are summarized in Table 3. Not surprisingly, all the five algorithms almost converged to the same maximum point on the boundary; that is, a bonding temperature of 580 °C ($x_1 = 1$), a bonding force of 13 kg ($x_2 = 1$) and a bonding time of 250 ms ($x_3 = 1$) with $\hat{f} = 93.2940$. Khoo and Chen (2001) obtained a similar but slightly worse solution ($\hat{f} = 93.1625$) via 1150 GA iterations. From Table 3, it shows that the hybrid NM–GA and NM–PSO converged much faster to the global optimum than NM, GA and PSO (see the \overline{ITE} and S_{ITE} statistics in the table). Based on the results displayed in this case, it is reasonable to allege that the NM–GA and NM–PSO hybrid methods are exceptionally efficient at locating optimum operating points of “standard” RSM problems (with the characteristic of monotone hill-climbing or walking down to valley on response surface).

4.3. Case study 3

This case study, a sequel to case study 2, was also considered in Khoo and Chen (2001), where an extra response variable “minimum strength” was included to create a multiple-responses problem. The goal was

Table 3
Computational results of case study 2

x_0	Method	$\bar{x} S_x$	$\bar{F} S_F$	$\overline{ITE} S_{ITE}$
(540, 12, 230)	Optimal	580, 13, 250	93.2940	
	NM	580, 13, 250	93.2940	20
	GA	580, 13, 250 (0.0, 0.0, 0.0)	93.2940 (0.0)	38.1 (16.21)
	PSO	580, 13, 244.2625 (0.0, 0.0, 5.3399)	92.7189 (0.5349)	176.3 (119.6393)
	NM–GA	580, 13, 250 (0.0, 0.0, 0.0)	93.2940 (0.0)	2.2 (0.6325)
	NM–PSO	580, 13, 250 (0.0, 0.0, 0.0)	93.2940 (0.0)	13.3 (4.3729)

Table 4
Computational results of case study 3

x_0	Method	$\bar{X} S_X$	$\bar{F}_1 S_{F1}$	$\bar{F}_2 S_{F2}$	ITE S_{ITE}
(540, 12, 230)	Optimal	580, 13, 250	93.2940	64.560	
	NM	580, 13, 250	93.2940	64.560	20
	GA	580, 13, 250 (0.0, 0.0, 0.0)	93.2940 (0.0)	64.560 (0.0)	42.8 (14.4207)
	PSO	580, 13, 249.8895 (0.0, 0.0, 0.03493)	93.2829 (0.0352)	64.5448 (0.0481)	180.5 (215.7119)
	NM-GA	580, 13, 250 (0.0, 0.0, 0.0)	93.2940 (0.0)	64.560 (0.0)	5.6 (1.8974)
	NM-PSO	580, 13, 250 (0.0, 0.0, 0.0)	93.2940 (0.0)	64.560 (0.0)	10.9 (4.7481)

to maximize both the measured responses of “mean pull strength” and “minimum strength” simultaneously. The fitted response surfaces established in Khoo and Chen (2001) were

$$\hat{f}_1(\mathbf{x}) = 73.89 + 12.91x_1 + 7.11x_2 + 2.56x_3 - 1.96x_1^2 - 1.01x_2^2 + 0.022x_3^2 + 0.36x_1x_2 - 0.068x_1x_3 - 0.52x_2x_3 \quad (7)$$

and

$$\hat{f}_2(\mathbf{x}) = 45.06 + 14.11x_1 + 6.56x_2 + 2.17x_3 - 1.69x_1^2 - 1.02x_2^2 + 0.14x_3^2 - 1.08x_1x_2 + 0.83x_1x_3 - 0.52x_2x_3, \quad (8)$$

where \hat{f}_1 is the predicted mean pull strength (in gram force); \hat{f}_2 is the predicted minimum strength (in gram force); x_1 – x_3 are the coded process variables as explained before. We follow Khoo and Chen’s (2001) overall pseudo-objective function to cope with the multiple-responses problem. For this case, it turns out to be univariate maximization on $(\hat{f}_1 + \hat{f}_2)$ with an equal weight placed on measured responses.

The hybrid methods were used to maximize $(\hat{f}_1 + \hat{f}_2)$ and the results are summarized in Table 4. All the five algorithms almost converged to the same maximum point on the boundary; that is, a bonding temperature of 580 °C ($x_1 = 1$), a bonding force of 13 kg ($x_2 = 1$) and a bonding time of 250 ms ($x_3 = 1$) with $\hat{f}_1 = 93.2940$ and $\hat{f}_2 = 64.560$. Khoo and Chen (2001) claimed a somewhat inferior solution ($\hat{f}_1 = 93.0604$ and $\hat{f}_2 = 64.2827$) via 582 GA iterations. Case study 3 resembles case study 2 in all performance statistics (see Table 4) that the hybrid algorithms converged much faster to the optimum solution on the boundary than the other three methods when the response surface to be optimized tends to be strictly increasing or decreasing in the feasible region.

4.4. Case study 4

In the last example, we would like to test the hybrid methods against another highly nonlinear model, which could occur “quite often” in response surface analysis (see Khuri & Cornell (1996)). For instance, Vohnout and Jimenez (1975) conducted a nonlinear response surface study, the objective of which is to develop methods for optimal utilization of tropical resources in livestock feeding. In their study, it was assumed that a nonlinear model of the “exponential” function form would adequately represent the relationship between the response variable and a set of process variables. In this case study, the fitted response function to be assumed is referred to as the “peaks function” demonstrated in MATLAB® toolboxes, which is a two dimensional, “exponential-type”, multimodal function as given below

$$\hat{f} = 3(1 - x_1)^2 \exp(-x_1^2 - (x_2 + 1)^2) - 10(x_1/5 - x_1^3 - x_2^5) \exp(-x_1^2 - x_2^2) - (1/3) \exp(-(x_1 + 1)^2 - x_2^2), \quad (9)$$

where the optimization search is limited in the cuboidal region, $-3 \leq x_1 \leq 3$ and $-3 \leq x_2 \leq 3$. Clearly, Eq. (9) belongs to the class of nonlinear regression model since the model is nonlinear in the parameters. In the compact region of interest, there exist two local minima and three local maxima. The unique global minimum occurs at (0.2282, −1.6256) with a response value of −6.511; the unique global maximum resides at (−0.0093, 1.5814) with a response value of 8.1062 (see the surface and contour plots in Figs. 12 and 13). The other local

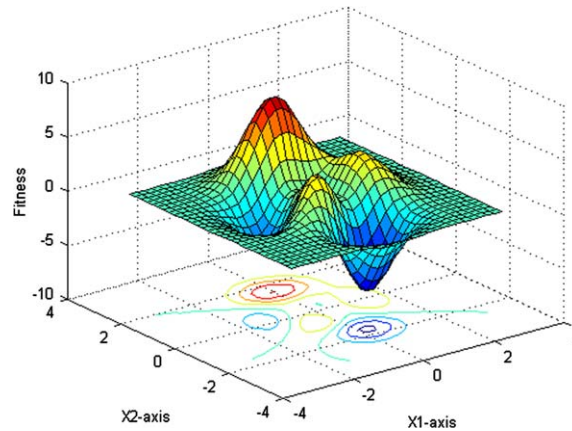


Fig. 12. A surface plot with contours of the peaks function in case study 4.

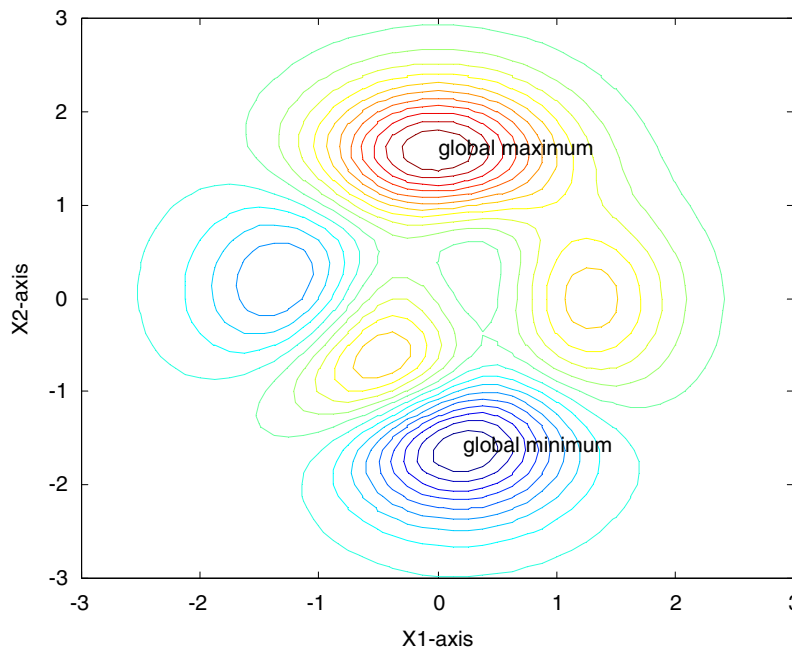


Fig. 13. A contour plot of the peaks function in case study 4.

minimum takes place at $(-1.3475, 0.2044)$ with a response value of -3.0498 ; the other two local maximum take place at $(1.2858, -0.0048)$ and $(-0.4601, -0.6292)$ with response values of 3.5925 and 3.7766, respectively.

The predicted response function in (9) is with curvilinear surface much more nonlinear than Eq. (5) since an exponential form is involved. For this complicated situation, we wish to find the global minimum and maximum separately from different starting points. The hybrid methods were used to search global optima of the peaks function. Tables 5 and 6 display the computational results returned by using the five approaches. Several initial points (denoted by x_0) were, again, selected near the local solutions to start the algorithms for the testing purpose.

As can be seen from Table 5, NM still maintained its consistency to attain an optimum solution near the starting point. When $x_0 = (-1, -1)$, GA and PSO could not guarantee the global maximum on each of ten runs; when $x_0 = (1, 0)$, PSO was trapped in a local maximum in some runs. In contrast, the two hybrid meth-

Table 5
Computation results on the peak function for searching the global maximum

x_0	Method	$\bar{x} S_x$	$\bar{F} S_F$	$\overline{ITE} S_{ITE}$
(0, 0)	Optimal	−0.0093, 1.5814	8.1062	
	NM	−0.0093, 1.5814	8.1062	36
	GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	33.7 (5.6184)
	PSO	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	66.1 (4.6056)
	NM–GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	32.6 (3.2042)
(0, 1)	NM–PSO	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	30.7 (2.9078)
	NM	−0.0093, 1.5814	8.1062	34
	GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	29.1 (4.7947)
	PSO	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	67.6 (3.1340)
	NM–GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	28 (1.4970)
(−1, −1)	NM–PSO	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	33.7 (8.5381)
	NM	−0.4601, −0.6292	3.7766	29
	GA	−0.0545, 1.3603 (0.1425, 0.6990)	7.6732 (1.3691)	27.5 (4.5277)
	PSO	0.0694, 0.3799 (0.6753, 1.0616)	5.4716 (2.2686)	66.8 (4.1402)
	NM–GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	35.7 (5.6970)
(1, 0)	NM–PSO	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	37.9 (4.5814)
	NM	1.2858, −0.0048	3.5925	32
	GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	28.2 (3.4897)
	PSO	0.5087, 0.9469 (0.6687, 0.8191)	6.3007 (2.3309)	65.2 (7.3606)
	NM–GA	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	32.8 (3.4577)
	NM–PSO	−0.0093, 1.5814 (0.000, 0.000)	8.1062 (0.000)	34.8 (2.7809)

Table 6
Computation results on the peak function for searching the global minimum

x_0	Method	$\bar{x} S_x$	$\bar{F} S_F$	$\overline{ITE} S_{ITE}$
(0, 0)	Optimal	0.2282, −1.6256	−6.5511	
	NM	0.2282, −1.6256	−6.5511	34
	GA	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	29.6 (3.3066)
	PSO	−0.2444, −1.0765 (0.7611, 0.8840)	−5.5007 (1.6913)	67.4 (4.2740)
	NM–GA	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	37.1 (5.3009)
(0, −1)	NM–PSO	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	43.5833 (15.2223)
	NM	0.2282, −1.6256	−6.5511	32
	GA	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	29.9 (4.0947)
	PSO	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	68.1 (2.7669)
	NM–GA	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	31.5 (2.6771)
(−1, 0)	NM–PSO	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	33 (4.4222)
	NM	−1.3475, 0.2044	−3.0498	34
	GA	−0.0868, −1.2595 (0.6643, 0.7717)	−5.8509 (1.4763)	30.3 (7.3492)
	PSO	−1.0323, −0.1615 (0.6644, 0.7716)	−3.7501 (1.4763)	65.2 (5.2451)
	NM–GA	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	32.3636 (3.8542)
	NM–PSO	0.2282, −1.6256 (0.000, 0.000)	−6.5511 (0.000)	39.3846 (5.9657)

ods could accurately identify the global maximum (−0.0093, 1.5814) in each run regardless of the starting points (see S_x and S_F in Table 5). The hybrid methods also exhibited a rather competitive edge on convergence speed, averaging about 33 iterations per run (see the performance statistics of \overline{ITE} and S_{ITE} in Table 5). For the minimization case, the computational results shown in Table 6 were analogous to those in Table 5. Three starting points were tried for every method. When $x_0 = (−1, 0)$, the pure GA and PSO algorithms could only find a local minimum in some runs; when $x_0 = (0, 0)$, PSO yielded inferior performance on solution quality. The optimization results shown here, combined with those in case study 1, point out explicitly that the pure GA and PSO approaches will be having trouble in “anchoring” the global optimum if the starting point is remote from it.

Computational experience gained on the preceding 4 case studies confirms the rich potential of our hybridization strategies combining GA and PSO with NM while applied to practical response surface optimization

problems. Meanwhile, the comparison report also points out an opportunity to improve the original GA and PSO algorithms for solving more “difficult” nonlinear optimization problems. However, RSM is only one possible application. To further examine the solving capability of these two hybrid algorithms extended to a more general framework of continuous optimization, an additional comprehensive test against several existing state-of-the-art algorithms is to be performed in the next section.

5. Computational experience of continuous nonlinear optimization

In this section, the efficiency and effectiveness of the two hybrid methods NM–GA and NM–PSO were evaluated in terms of 10 benchmark functions selected from Chelouah and Siarry (2003). The problem dimension ranges from 2 to 10 (see detailed function description in the Appendix A). In order to conduct fair computational tests, each benchmark problem was solved 100 times by using independent starting points randomly selected inside the pre-specified search domain of a hyper-rectangular region. The random selection procedure opted here is to reflect the fact reported in the open literature that the convergence of the NM algorithm relies heavily on the initial starting point. The stopping criterion applied in this comprehensive test is a bit different from Eq. (4) described in Section 4. To achieve quicker convergence, the algorithms (NM–GA and NM–PSO) will be terminated when either Eq. (4) with $\varepsilon = 1 \times 10^{-4}$ or a pre-specified maximum number of iterations $100 \times N$ is first reached.

5.1. The comprehensive test of NM–GA and NM–PSO

Following Chelouah and Siarry (2003), we evaluated the algorithms based on three performance measures, which had been collected from 100 independent minimizations (or runs) per test function. They are the rate of successful minimizations, the average of the objective function evaluation numbers, and the average error on the objective function. These performance criteria are defined below. Either one of the termination criteria is first reached, the algorithms stop and return the coordinates of a final optimal point, and the final optimal objective function value “FOBJ_{ALG}” (algorithm) at this point. We compared this result with the known analytical minimum objective value “FOBJ_{ANAL}” and deemed this solution to be “successful” if the following inequality holds:

$$|\text{FOBJ}_{\text{ALG}} - \text{FOBJ}_{\text{ANAL}}| < \varepsilon_{\text{rel}} |\langle \text{FOBJ}_{\text{INIT}} \rangle| + \varepsilon_{\text{abs}},$$

where $\varepsilon_{\text{rel}} = 10^{-4}$ (relative error), $\varepsilon_{\text{abs}} = 10^{-6}$ (absolute error) and $\langle \text{FOBJ}_{\text{INIT}} \rangle$ is an empirical average of the objective function values, calculated over 100 randomly selected, feasible points before running the algorithm. The relative error ε_{rel} is opted here to accommodate the actual error of the objective function value incurred by the algorithm itself; the absolute error ε_{abs} is allowed to account for the truncation error possibly resulting from computing machinery. The average of the objective function evaluation numbers is only accounted in relation to the “successful minimizations.” The average error is defined as the average of FOBJ deviation (or gap) between the best successful point found and the known global optimum, where only the “successful minimizations” achieved by the algorithm are taken into account.

For the first part of the comprehensive test, the computational results of NM–GA and NM–PSO over the 10 benchmark test functions were tabulated in Table 7. As can clearly be seen from the table, both hybrid algorithms exhibited perfect successful rates (100%) in every test instances of each benchmark function of problem size ranging from $N = 2$ to 10. Concerning the average of objective function evaluation numbers, NM–PSO requires approximately 38% less function evaluations than NM–GA for algorithmic convergence. This outcome coincides with a widely accepted understanding documented in the literature that swarm intelligence is generally more cost-effective than GA for solving continuous optimization problems. For both algorithms, the average function evaluation number increases as the problem size increases, so does the average gap between the best successful point found and the known global optimum. Through the comprehensive evaluation over these ten test functions, it has been discovered that the NM–GA and NM–PSO algorithms are extremely capable of solving general nonlinear optimization problems together with appreciable solution accuracy on objective function. In the situation presented here, the solution accuracy was at least down to four decimal places.

Table 7
Computational results of NM–GA and NM–PSO for 10 test functions

Test function	Rate of successful minimization (%)		Average of objective function evaluation numbers		Average gap between the best successful point found and the known global optimum	
	NM–GA	NM–PSO	NM–GA	NM–PSO	NM–GA	NM–PSO
RC	100	100	356	230	0.00004	0.00010
B2	100	100	529	325	0.00004	0.00000
GP	100	100	422	304	0.00002	0.00003
SH	100	100	1009	753	0.00002	0.00003
R ₂	100	100	738	440	0.00006	0.00005
Z ₂	100	100	339	186	0.00004	0.00000
H _{3,4}	100	100	688	436	0.00005	0.00012
S _{4,5}	100	100	2366	850	0.00016	0.00006
R ₅	100	100	3126	2313	0.00009	0.00004
R ₁₀	100	100	5194	3303	0.00020	0.00012

Next, the performances of NM–GA and NM–PSO are compared to other eight existing algorithms suitable for continuous optimization, including variants of GA, tabu search, simulated annealing, among others. Of particular interest to note in this comparison report is the CHA algorithm in Chelouah and Siarry (2003), using a different style of hybridization philosophy between GA and NM. These methods with references are listed in Table 8. For a detailed account, interested readers may refer to Chelouah and Siarry (2003).

The comparison results via eight test problems of $N \leq 5$ are exhibited in Table 9. For each test function and algorithm, average numbers of function evaluation over 100 runs are provided. The computational results of some algorithms are not available from the literature. The number in parenthesis indicates the ratios of experimental runs where the algorithm located the global solution instead of being trapped into a local solution. As seen from the table, CHA is the most competitive method among these eight existing algorithms. The compu-

Table 8
List of various methods used in the comparison

Method	Reference
Hybrid Nelder–Mead simplex method and genetic algorithm (NM–GA)	This paper
Hybrid Nelder–Mead simplex method and particle swarm optimization (NM–PSO)	This paper
Continuous hybrid algorithm (CHA)	Chelouah and Siarry (2003)
Enhanced continuous tabu search (ECTS)	Chelouah and Siarry (2000a)
Continuous genetic algorithm (CGA)	Chelouah and Siarry (2000b)
Enhanced simulated annealing (ESA)	Siarry et al. (1997)
Continuous reactive tabu search (CRTSmin.) minimum	Battiti and Tecchiolli (1996)
Continuous reactive tabu search (CRTSave.) average	Battiti and Tecchiolli (1996)
Taboo search (TS)	Cvijovic and Klinowski (1995)
INTEROPT	Bilbro and Snyder (1991)

Table 9
Average numbers of objective function evaluations by using 10 different methods to optimize 8 test functions of less than 5 variables (Chelouah & Siarry (2003))

Function	NM–GA	NM–PSO	CHA	CGA	ECTS	CRTS min.	CRTS ave.	TS	ESA	INTEROPT
RC	356	230	295	620	245	41	38	492	–	4172
GP	422	304	259	410	231	171	248	486	783	6375
B2	529	325	132	320	210	–	–	–	–	–
SH	1009	753	345	575	370	–	–	727	–	–
R ₂	738	440	459	960	480	–	–	–	796	–
Z ₂	339	186	215	620	195	–	–	–	15820	–
H _{3,4}	688	436	492	582	548	609	513	508	698	1113
S _{4,5}	2366 (1.0)	850 (1.0)	598 (0.85)	610 (0.76)	825 (0.75)	664	812	–	1137 (0.54)	3700 (0.4)

tational results of NM-PSO are very similar to those returned by CHA in terms of the average number of function evaluation, but NM-GA requires much more numbers of function evaluations for convergence. For the first 7 test functions where 100% of global optimality is obtained, the grand averages of function evaluations are 583 for NM-GA, 382 for NM-PSO, 314 for CHA, and 584 for CGA. The performance of NM-GA is close to that of CGA. Even though NM-PSO performs not as good as CHA in the average number of function evaluation, the ratios of global optimality generated by NM-GA and NM-PSO are much better than CHA in $S_{4,5}$. In light of the foregoing comparisons, NM-PSO remains quite competitive for every test functions to be a viable and promising solver for continuous nonlinear optimization. Furthermore, the possible refinement on the parameter selection in NM-GA and NM-PSO still remains unanswered. This type of comprehensive parameter analysis may be beneficial to expedite their convergences. It is particularly important to emphasize once again that this paper focuses primarily on the hybridization process of GA and PSO with respect to NM. The current, crude implementations of NM-GA and NM-PSO are on methodology; future implementations will include additional speed enhancements.

5.2. Graphical illustration of global convergence property of NM-GA and NM-PSO

In this section, a two-dimensional test function GP having four local minima and one global minimum (see the [Appendix A](#)) will be used to demonstrate why the proposed hybrid algorithms can find the global optimum efficiently. [Fig. 14](#) illustrates the performances of all five methods (i.e., NM, GA, PSO, NM-GA and NM-PSO) starting from the same initial point (0, 0) on the GP test function by plotting the best objective function attained versus the first 10 iterations for an individual optimization run. It can clearly be seen from the figure that the two hybrid algorithms NM-GA and NM-PSO yielded significantly larger objective function reduction than the other three methods from iterations 1–3. To look closely at the convergence property of each algorithm, [Fig. 15](#) shows 5 separate convergence performances by plotting the best objective function versus the number of iteration required until the stopping criterion was reached. The numbers of iteration for convergence are actually 29 for NM, 50 for PSO, 35 for GA, 25 for NM-PSO and 23 for NM-GA. However, performance assessment hinged merely on the number of iteration may not be fair since the total number of function evaluation is of most computation cost relevance. Typically, the pure local search method, such as NM, is less computational intensive in comparison with meta-heuristic methods, yet the capability of capturing possible global optimality is one of major weaknesses of the local search method.

[Fig. 16](#) depicts the search courses of the NM, PSO, GA, NM-GA and NM-PSO algorithms while optimizing the GP test function. From these tracks, it can be seen that the NM, GA and PSO methods began to approach a local optimum and then was rerouted to the global optimum, explaining why these three methods need more “iterations” (not function evaluations) for convergence, and the NM-GA and NM-PSO algo-

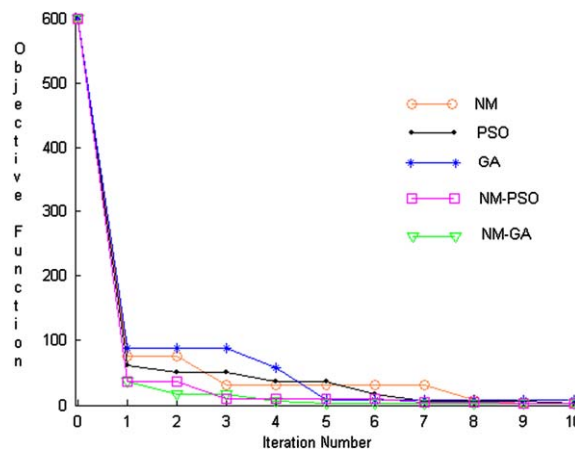


Fig. 14. Objective function value of GP test function versus iteration obtained by using five different algorithms.

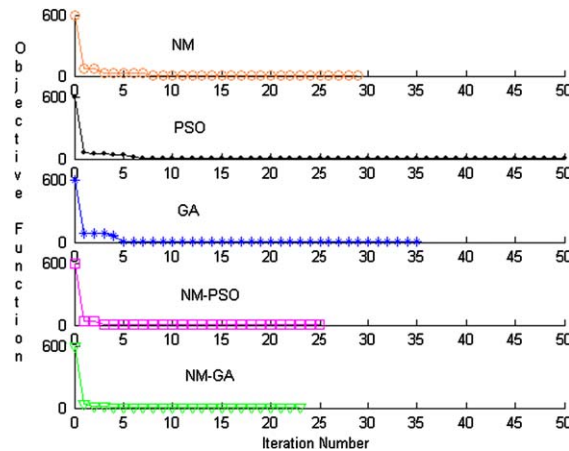


Fig. 15. Individual convergence performance for five different algorithms.

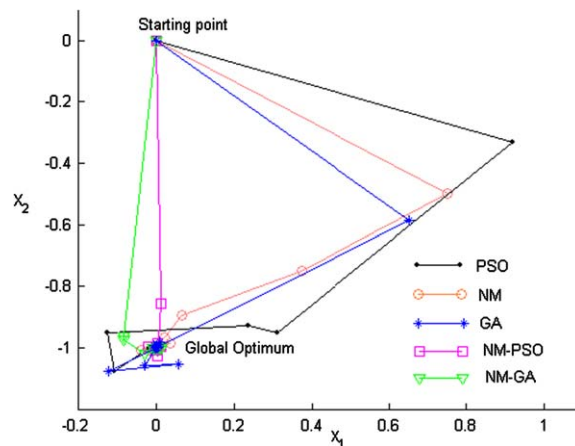


Fig. 16. Search paths of PSO, NM, GA, NM-PSO and NM-GA while minimizing GP test function using (0, 0) as the starting point. Note that the global optimum occurs at (0, -1).

algorithms were already near the neighborhood of the global optimum after one single jump. This observation may explain partly why the two proposed hybrid algorithms perform better than the pure GA and PSO methods for continuous nonlinear optimization. To sum up, the computational results presented in Section 5 strongly suggest that the two proposed hybrid approaches combine harmonically the advantages between the local search and meta-heuristic optimization mechanisms. Even compared to the best heuristic to date, the hybrid NM-PSO sustains competitiveness on solving reliability and quality efficiency.

6. Conclusions

This paper demonstrated the possibility and potential of integrating NM with GA or PSO for locating the global optima of nonlinear continuous variable functions, particularly on highly nonlinear response surface models. The proposed hybrid methods were validated using four case studies. Two of these four case studies belong to the nonlinear response surface models. Case study 1 involved solving the Himmelblau function and the results showed that the hybrid methods were able to reach the global minimum under different initial starting points and exhibited satisfactory convergence speed. In case study 2, a single response problem was assumed and the results showed that the hybrid methods converged faster to the global optimum than

NM, GA and PSO. In case study 3, the hybrid methods were used to handle a multiple-responses and multi-constraints problem and produced fairly similar results as in case study 2. In case study 4, the hybrid methods were used to search the global maximum and global minimum of the peaks function individually, the results produced were fairly similar to those as in case study 1.

The proposed hybrid NM–GA and NM–PSO algorithms were then tested on some difficult nonlinear continuous functions from the literature. Both algorithms were successful to reach the global optimum in all instances over all runs. Considering the computational expense, NM–PSO showed very competitive performance to the best heuristic in the literature while NM–GA also performs comparably. These observations lead us to conclude that the proposed hybrid NM–GA and NM–PSO are indeed effective, reliable, efficient and robust at locating best-practice optimum solutions for continuous variable function optimization problems, particularly for linear and nonlinear response surface problems.

Acknowledgements

We would like to thank the Special Issue Editor, Dr. Cathal Heavey at University of Limerick, Limerick, Ireland, and three anonymous referees for their constructive comments on an earlier version of this paper. Dr. Fan is partly supported by National Science Council, via grant NSC 93-2213-E-155-009.

Appendix A. List of test functions

Branin RCOC (RC) (2 variables):

$$RC(x_1, x_2) = (x_2 - (5/(4\pi^2))x_1^2 + (5/\pi)x_1 - 6)^2 + 10(1 - (1/(8\pi))) \cos(x_1) + 10;$$

search domain: $-5 < x_1 < 10$, $0 < x_2 < 15$;

no local minimum;

3 global minima: $(x_1, x_2)^* = (-\pi, 12.275)$, $(\pi, 2.275)$, $(9.42478, 2.475)$;

$$RC((x_1, x_2)^*) = 0.397887.$$

B2 (2 variables):

$$B2(x_1, x_2) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1) - 0.4 \cos(4\pi x_2) + 0.7;$$

search domain: $-100 < x_j < 100$, $j = 1, 2$;

several local minima (exact number unspecified in literature);

1 global minimum: $(x_1, x_2)^* = (0, 0)$; $B2((x_1, x_2)^*) = 0$.

Goldstein and Price (GP) (2 variables):

$$GP(x_1, x_2) = [1 + (x_1 + x_2 + 1)^2 \times (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)];$$

search domain: $-2 < x_j < 2$, $j = 1, 2$;

4 local minima; 1 global minimum: $(x_1, x_2)^* = (0, -1)$; $B2((x_1, x_2)^*) = 3$.

Shubert (SH) (2 variables):

$$SH(x_1, x_2) = \left\{ \sum_{j=1}^5 j \cos[(j+1)x_1 + j] \right\} \times \left\{ \sum_{j=1}^5 j \cos[(j+1)x_2 + j] \right\};$$

search domain: $-10 < x_j < 10$, $j = 1, 2$;

760 local minima; 18 global minima: $SH((x_1, x_2)^*) = -186.7309$.

Rosenbrock (R_n) (n variables):

$$R_n(\mathbf{x}) = \sum_{j=1}^{n-1} [100(x_j^2 - x_{j+1})^2 + (x_j - 1)^2];$$

- 3 functions were considered: R_2 , R_5 , R_{10} ;
 search domain: $-5 < x_j < 10$, $j = 1, \dots, n$;
 several local minima (exact number unspecified in literature);
 1 global minimum: $\mathbf{x}^* = (1, \dots, 1)$; $R_n(\mathbf{x}^*) = 0$

Zakharov (Z_n) (n variables):

$$Z_n(\mathbf{x}) = \left(\sum_{j=1}^n x_j^2 \right) + \left(\sum_{j=1}^n 0.5jx_j \right)^2 + \left(\sum_{j=1}^n 0.5jx_j \right)^4$$

- 3 functions were considered: Z_2 , Z_5 , and Z_{10} ;
 search domain: $-5 < x_j < 10$, $j = 1, \dots, n$;
 several local minima (exact number unspecified in literature);
 1 global minimum: $\mathbf{x}^* = (0, \dots, 0)$; $Z_n(\mathbf{x}^*) = 0$

Hartmann ($H_{3,4}$) (3 variables):

$$H_{3,4}(\mathbf{x}) = - \sum_{i=1}^4 c_i \exp \left[- \sum_{j=1}^3 a_{ij}(x_j - p_{ij})^2 \right];$$

- search domain: $0 < x_j < 1$, $j = 1, 2, 3$;
 4 local minima: $\mathbf{p}_i = (p_{i1}, p_{i2}, p_{i3}) = i$ th local minimum approximation; $f(\mathbf{p}_i) \cong -c_i$;
 1 global minimum: $\mathbf{x}^* = (0.11, 0.555, 0.855)$; $H_{3,4}(\mathbf{x}) = -3.86278$.

i	a_{ij}			c_i	p_{ij}		
1	3.0	10.0	30.0	1.0	0.3689	0.1170	0.2673
2	0.1	10.0	35.0	1.2	0.4699	0.4387	0.7470
3	3.0	10.0	30.0	3.0	0.1091	0.8732	0.5547
4	0.1	10.0	35.0	3.2	0.0381	0.5743	0.8827

Shekel ($S_{4,n}$) (4 variables):

$$S_{4,n}(\mathbf{x}) = - \sum_{i=1}^n [(x - \mathbf{a}_i)^T (\mathbf{x} - \mathbf{a}_i) + c_i]^{-1};$$

$$\mathbf{x} = (x_1, x_2, x_3, x_4)^T; \mathbf{a}_i = (a_i^1, a_i^2, a_i^3, a_i^4)^T;$$

- 3 functions $S_{4,n}$ were considered: $S_{4,5}$, $S_{4,7}$ and $S_{4,10}$;
 search domain: $0 < x_j < 10$, $j = 1, \dots, 4$;
 n local minima ($n = 5, 7$ or 10): $\mathbf{a}_i^T = i$ th local minimum approximation: $S_{4,n}((\mathbf{a}_i^T)) \cong -1/c_i$;
 $S_{4,5}(n = 5)$ 5 minima with 1 global minimum: $S_{4,5}(\mathbf{x}) = -10.1532$

i	\mathbf{a}_i^T				c_i
1	4.0	4.0	4.0	4.0	0.1
2	1.0	1.0	1.0	1.0	0.2
3	8.0	8.0	8.0	8.0	0.2
4	6.0	6.0	6.0	6.0	0.4
5	3.0	7.0	3.0	7.0	0.4
6	2.0	9.0	2.0	9.0	0.6
7	5.0	5.0	3.0	3.0	0.3
8	8.0	1.0	8.0	1.0	0.7
9	6.0	2.0	6.0	2.0	0.5
10	7.0	3.6	7.0	3.6	0.5

References

- Battiti, R., & Tecchiolli, G. (1996). The continuous reactive tabu search: blending combinatorial optimization and stochastic search for global optimization. *Annals of Operations Research*, 63, 53–188.
- Bilbro, G. L., & Snyder, W. E. (1991). Optimization of functions with many minima. *IEEE Transaction on System, Man and Cybernetics*, 21, 840–849.
- Box, G. E. P., & Draper, N. R. (1987). *Empirical model-building and response surfaces*. New York: Wiley.
- Box, G. E. P., & Lucas, H. L. (1959). Design of experiments in nonlinear situations. *Biometrika*, 46, 77–90.
- Box, G. E. P., & Wilson, K. G. (1951). On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society, Series B*, 28, 1–45.
- Brandstatter, B., & Baumgartner, U. (2002). Particle swarm optimization-mass-spring system analogon. *IEEE Transactions on Magnetics*, 38, 997–1000.
- Chelouah, R., & Siarry, P. (2000a). Tabu search applied to global optimization. *European Journal of Operational Research*, 123, 30–44. Special issue on combinatorial optimization.
- Chelouah, R., & Siarry, P. (2000b). A continuous genetic algorithm designed for the global optimization of multimodal functions. *Journal of Heuristics*, 6, 191–213.
- Chelouah, R., & Siarry, P. (2003). Genetic and Nelder–Mead algorithms hybridized for a more accurate global optimization of continuous multimodal functions. *European Journal of Operational Research*, 148, 335–348.
- Cvijovic, D., & Klinowski, J. (1995). An approach to the multiple minima problem. *Science*, 267, 664–666.
- D'Angelo, A., Gastaldi, M., & Levialdi, N. (1998). Performance analysis of a flexible manufacturing system: a statistical approach. *International Journal of Production Economics*, 47–59.
- Deb, K. (1995). *Optimization for engineering design: Algorithms and examples*. New Delhi: Prentice-Hall of India.
- Draper, N. R. (1963). Ridge analysis of response surfaces. *Technometrics*, 5, 469–479.
- Eberhart, R.C. & Shi, Y. (2001). Tracking and optimizing dynamic systems with particle swarms. *Proceedings of the 2001 Congress on Evolutionary Computation*, Seoul, Korea, 94–97.
- Fan, S.-K. S. (2003a). Implementation of ridge analysis on quadratic response surfaces. *Journal of Statistical Computation and Simulation*, 73, 675–684.
- Fan, S.-K. S. (2003b). A different view of ridge analysis from numerical optimization. *Engineering Optimization*, 35, 627–647.
- Fan, S.-K. S., Liang, Y. -C., & Zahara, E. (2004). Hybrid simplex search and particle swarm optimization for the global optimization of multimodal functions. *Engineering Optimization*, 36, 401–418.
- Goldberg, D. E. (1998). *Genetic algorithms in search, optimization and machine learning*. New York: Addison-Wesley.
- Holland, J. (1975). *Adaptation in natural and artificial systems*. Ann Arbor, MI, USA: The University of Michigan Press.
- Hu, X., & Eberhart, R.C. (2001). Tracking dynamic systems with PSO: where's the cheese?. *Proceedings of the Workshop on Particle Swarm Optimization*, Indianapolis, IN, USA.
- Janikow, C.Z., & Michalewicz, Z. (1991). An experimental comparison of binary and floating point representation in genetic algorithms. *Proceedings of the International Conference on Genetic Algorithms*, San Diego, CA, USA, 31–36.
- Kennedy, J., & Eberhart, R.C. (1995). Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks*, Piscataway, NJ, USA, 1942–1948.
- Kennedy, J., & Shi, Y. (2001). *Swarm intelligence*. San Diego, CA, USA: Academic Press.
- Khoo, L. P., & Chen, C. H. (2001). Integration of response surface methodology with genetic algorithms. *The International Journal of Advanced Manufacturing Technology*, 18, 483–489.
- Khuri, A. I., & Cornell, J. A. (1996). *Response surfaces: Designs and analyses* (2nd ed.). New York: Marcel Dekker.
- Myers, R. H. (1971). *Response surface methodology*. Boston: Allyn and Bacon.
- Myers, R. H., & Montgomery, D. C. (2002). *Response surface methodology: Process and product optimization using designed experiments*. New York: Wiley.
- Nash, S. G., & Sofer, A. (1996). *Linear and nonlinear programming*. New York: McGraw-Hill.
- Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7, 308–313.
- Nybergh, K., Marjamäki, T., & Skarp, E. (1997). Modeling the growth of PECVD dilicon nitride films for crystalline dilicon solar cells using factorial design and response surface methodology. *Proceedings of the IEEE Photovoltaic Specialists Conference*, Piscataway, NJ, USA, 143–146.
- Olsson, D. M., & Nelson, L. S. (1975). The Nelder–Mead simplex procedure for function minimization. *Technometrics*, 17, 45–51.
- Parkinson, J. M., & Hutchinson, D. (1972). An investigation into the efficiency of variants of the simplex method. In F. A. Lootsma (Ed.), *Numerical methods for non-linear optimization* (pp. 115–135). London: Academic Press.
- Renders, J. M., & Flasse, S. P. (1996). Hybrid methods using genetic algorithms for global optimization. *IEEE Transaction on Systems, Man, and Cybernetics-Part B: Cybernetics*, 26, 243–258.
- The SAS System for Windows Release 6.12 Copyright © 1989–1996 (Cary, NC: SAS Institute Corporation).
- Scotti, M. V., Malik, Z., Cheung, P. Y. K., & Nelder, J. A. (2000). Optimization of full-custom logic cells using response surface methodology. *Electronics Letters*, 36, 14–16.
- Siarry, P., Berthiau, G., Durbin, F., & Haussy, J. (1997). Enhanced simulated annealing for globally minimizing functions of many continuous variables. *ACM Transactions on Mathematical Software*, 23, 209–228.
- Sondergeld, L., & Voss, S. (1999). Cooperative intelligent search using adaptive memory techniques. In S. Voss, S. Martello, I. H. Osman, & C. Roucairol (Eds.), *Meta-heuristics: Advances and trends in local search paradigms for optimization* (pp. 297–312). Kluwer.

- Spendley, W., Hext, G. R., & Himsworth, F. R. (1962). Sequential application of simplex designs in optimization and evolutionary operation. *Technometrics*, 4, 441–461.
- Vohnout, K. & Jimenez, C. (1975). Supplemental by-product feeds in pasture-livestock feeding systems in the tropics. In E.C. Doll and G.O. Mott (Eds.), *Tropical Forages in Livestock Production Systems*. ASA Special Publication No. 24, Madison, WI: American Society of Agronomy, 71–82.
- Yen, J., Liao, J. C., Lee, B., & Randolph, D. (1998). A hybrid approach to modeling metabolic systems using a genetic algorithm and simplex method. *IEEE Transaction on Systems, Man, and Cybernetics-Part B: Cybernetics*, 28, 173–191.
- Yoshida, H., Kawata, K., Fukuyama, Y., Takayama, S., & Nakanishi, Y. (2000). A particle swarm optimization for reactive power and voltage control considering voltage security assessment. *IEEE Transactions on Power Systems*, 15, 1232–1239.
- Zahara, E. (2003). *A study of Nelder–Mead simplex search method for solving unconstrained and stochastic optimization problems*. Ph.D. dissertation, Department of Industrial Engineering and Management, Yuan Ze University, Taoyuan County, Taiwan, Republic of China.
- Zahara, E., & Fan, S. K. (2004). Real-coded genetic algorithm for stochastic optimization: A tool for recipe qualification of semiconductor manufacturing under noisy environments. *International Journal of Advanced Manufacturing Technology*, 25, 361–369.