Estimation of Unknown Inlet Temperature Profile Using an Improved Gbest-PSO*

PENG DING, MINGHAI XU AND DONGLIANG SUN+

College of Storage and Transportation and Architectural Engineering
China University of Petroleum, Hua Dong
Qing Dao, ShanDong, 266555 P.R. China

*State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources
North China Electric Power University
Beijing, 102206 P.R. China

In this study, an improved gbest-PSO is proposed to overcome the shortcoming of earlier convergence of classical gbest-PSO. Then the improved gbest-PSO is used to identify the unknown inlet temperature profile in a plate channel flow. The effects of measurement position and measurement error on the accuracy of prediction are studied thoroughly. Analysis of computational results of two test problems shows that the improved gbest-PSO proposed in this paper has an excellent smooth convergence characteristic. The local refine mechanism introduced in the improved gbest-PSO increases the opportunity of finding the global optimum greatly especially for high dimensional multimodal optimization problems. Accurate results are obtained even when the measurements contain a 10% noise. Consequently, the inverse convection heat transfer problem is successfully solved by the improved gbest-PSO.

Keywords: particle swarm optimization, inverse convection heat transfer, channel flow, global optimization, evolutionary computation

1. INTRODUCTION

Inverse analysis is very valuable when the direct measurements of data are impossible or the measuring process is very expensive. For examples, the determination of heat transfer coefficients and the heat loads acting on the outer surface of reentry vehicle, the estimation of unknown thermophysical properties of unknown materials, the prediction of the glass ribbon temperature in the float glass process, the determination of contact resistance and damage detection in the structure fields and so on.

It is well known that inverse problems are solved by minimizing an object function with some stabilization technique used in the estimation procedure. Generally speaking, the minimization of object functions is achieved by traditional mathematical programming techniques, such as the Levenberg-Marquardt method of Marquardt [1] and the conjugate gradient method pioneer by Alifanov [2].

The conjugate gradient method of Alifanov [2] belongs to the function estimation approach and it is very efficient when no a priori information is available on the function form of the unknown quantities. It appears to be one of the most successful and universal approaches for the construction of stable algorithms for solving inverse problems. Many works solve the inverse heat transfer problems by the conjugate gradient method of Ali-
fanov [2]. Huang and Yan [3] use a conjugate gradient method of minimization for estimating the temperature-dependent thermal conductivity and heat capacity per unit volume of a material. In another work, Huang and Ozisik [4] use a combination of conjugate gradient method and modified conjugate gradient method to solve the inverse problem of determining the space-wise variation of an unknown wall flux for laminar flow inside a parallel plate duct. Bokar and Ozisik [5] utilize the same method to estimate the time-wise variation of inlet temperature of a thermally developing, hydrodynamically developed laminar flow between parallel plates by utilizing transient temperature measurements from a single thermocouple located downstream of the entrance. In the work of Colaco and Orlande [6], the conjugate gradient method is used for the simultaneous identification of two unknown boundary heat fluxes in an irregularly shaped channel with laminar flow. In the work of Ding and Tao [7], the unknown space-dependent heat flux at the boundary of a circular pipe is identified using the Fletcher-Reeves conjugate gradient method. The effects of Re number on the performance of inverse algorithm are studied thoroughly. The inverse convection problems in turbulent channel flow are also studied with conjugate gradient method in the works [8-10].

However, the conjugate gradient method has its own drawbacks. First, it is a gradient based method; the gradient of the object function in the direction of the unknown parameters must be known a prior. But under some conditions, the derivatives of the object function can not be calculated. Second, it is a local optimization method; and the performance of the algorithm is strongly affected by the initial guess values.

Recently, much attention has been drawn to the application of the computational intelligence techniques to the engineering optimization problems. These intelligence techniques are conceptually different from the traditional mathematical programming techniques. They are developed based on the modeling of biological and natural intelligence. These modern optimization methods mainly include genetic algorithm [11], and particle swarm optimization [12]. Compared with the conjugate gradient method, these modern optimization methods can search many possible solutions simultaneously without any gradient information and thus have the potential to give unbiased estimation. The basic idea of genetic algorithm is based on the principles of natural genetics and natural selection. The phenomena of reproduction, crossover, and mutation occurred in the natural selection process are modeled in the genetic algorithm for the search of optimum. There are some works in the open literatures concerning about solving inverse heat conduction problems using Genetic algorithm. In the work of Liu [13], a modified genetic algorithm is developed for solving the inverse heat conduction problem of estimating unknown plan heat source. Adili et al. [14] estimate the thermophysical properties of fouling deposited on internal surface of a heat exchanger tube using genetic algorithms. However, the performance of GA deteriorates apparently when the dimension of object function is high. On the other hand, the implementation of GA is relatively complex and there are many control parameters to tune before a specified problem to be solved.

The particle swarm optimization, abbreviated as PSO is originally proposed by Kennedy and Eberhart [12] in 1995. PSO is a stochastic optimization method, modeled on the social behavior of bird flocks and fish flocks. PSO is a population-based search procedure where each particle represents a candidate solution to the optimization problem. Each particle in the swarm adjusts its position according to the best positions found by itself and its neighbor, the final effect is that the swarm converge to an optimum. PSO is
very simple in concept and can be implemented in a few lines of computer code. There are few works aiming at solving inverse radiation problem with the PSO algorithm in open literatures. Lee et al. [15] perform an inverse radiation analysis for the estimation of the radiation properties for an absorbing, emitting, and scattering media with diffusely emitting and reflecting opaque boundaries. The repulsive particle swarm optimization (RPSO) is used for the identification of unknown emissivity, absorption and scattering coefficients. Also, the accuracy of estimated parameters and the computational efficiency are compared with results obtained from GA and classical PSO techniques.

However, to the author’s best knowledge, there are no works concerning the application of PSO to inverse convection heat transfer problems. In the work of Lee et al. [15], the dimension of the unknown quantities is no more than three. It is well known that the classical PSO performs badly when the dimension of unknown functions is high, since the classical PSO algorithm do not has a mechanism to jump out of the local optimum during the iteration process.

In this paper, an improved gbest-PSO is first proposed to overcome the shortcomings of classical gbest-PSO. Then, it is used to solve the inverse convection heat transfer problem of estimating unknown space-dependent inlet temperature profile.

2. MATHEMATICAL FORMULATION OF DIRECT PROBLEM

We consider a thermally developing, hydrodynamically developed laminar forced convection of the water through a parallel channel. Fig. 1 presents a schematic view of the present channel flow system. A heat flux \( q(x) \) is imposed on the bottom wall and insulation boundary condition at the top wall. Fluid enters the channel at a space-wise variation temperature of \( T_0(y) \). The governing equation for this problem is given by

\[
pC_{\mu}(\mu(y)) \frac{\partial T}{\partial x} = \frac{\partial}{\partial y} \left( \lambda \frac{\partial T}{\partial y} \right)
\]

with the boundary conditions:

\[
\lambda \frac{\partial T}{\partial y} \bigg|_{y=H} = 0
\]
\[-\lambda \frac{\partial T}{\partial y} \bigg|_{y=0} = q(x) \]  \hspace{1cm} (3)

\[T \big|_{y=0} = T_0(y) \]  \hspace{1cm} (4)

where \( \rho \) is the density, \( C_p \) is the capacity, \( \lambda \) is the thermal conductivity.

The fully developed velocity profile of a fluid in parallel channels is given by

\[u(y) = 6.0 u_m \frac{y}{H} \left(1 - \frac{y}{H}\right) \]  \hspace{1cm} (5)

where \( u_m \) is the average velocity. In this work, we take the height of channel \( H = 0.05 \text{m} \), \( \rho = 1.205 \text{Kg/m}^3 \), \( C_p = 1005.0 \text{J/(Kg} \cdot \text{K)} \), \( \eta = 0.0259 \text{W/(m} \cdot \text{K)} \). The length of channel is fixed for this investigation at a value of 0.2 meters. A 120 \times 15 uniform grid system is used to discretize the space domain.

\section*{3. INVERSE PROBLEM SOLVERS}

\subsection*{3.1 Classical Particle Swarm Optimization}

In the classical PSO algorithm of Kennedy and Eberhart \([12, 16]\), each particle flies in the multidimensional search space, the velocity of particle \( i \) is calculated as

\[V_{ij}(t+1) = wV_{ij}(t) + c_1 r_{1j}(t)[Y_{ij}(t) - X_{ij}(t)] + c_2 r_{2j}(t)[\hat{Y}(t) - X_{ij}(t)]\]  \hspace{1cm} (6)

where \( V_{ij}(t) \) is the velocity of \( i \)th particle in dimension \( j = 1, \ldots, n \) at time step \( t \) which represents a memory of the previous movement direction. \( X_{ij}(t) \) is the position of particle \( i \) in dimension \( j \) at time step \( t \). \( c_1 \) and \( c_2 \) are two positive acceleration constants. \( r_{1j}(t) \) and \( r_{2j}(t) \) are uniform distributed random values in the range \([0, 1]\), which introduce some stochastic elements to the particle swarm optimization algorithm. \( Y_{ij}(t) \) represents personal best position that the particle \( i \) has visited since the first time step. \( c_1 r_{1j}(t)[Y_{ij}(t) - X_{ij}(t)] \) termed as the cognitive component, it measures the distance between the particle’s current position and its past best position. The cognitive component has a tendency to adjust the particles’ movement direction to their own best positions. \( \hat{Y}(t) \) represents global best position at time step \( t \), it is the best position discovered by the whole swarm. \( c_2 r_{2j}(t)[\hat{Y}(t) - X_j] \) represents the social component, it drives each particle towards the best position found by the swarm. The inertia weight, \( w \) control how much the previous momentum will affect the new velocity.

The position of each particle is updated by adding the velocity \( V_{ij}(t+1) \) to the current position \( X_{ij}(t) \), i.e.

\[X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1).\]  \hspace{1cm} (7)

The particle swarm optimization of Eberhart and Shi \([12, 16]\) has been widely used
in many works [17-21]. The procedure of the particle swarm optimization (PSO) is described in the following:

1. Initialize the swarm and control parameters.
   (a) Set the number of particles, set the acceleration constants $c_1$ and $c_2$ and inertia weight $w$.
   (b) Set the initial positions of each particle $X_{ij}(0)$.
   (c) Set the initial velocity of each particle $V_{ij}(0)$.
   (d) Set the initial values of personal best position $Y_i$ and global best position $\hat{Y}$.

2. Evaluate the fitness of each particle and set the personal best position $Y_i$ and global best position $\hat{Y}$.
   (a) The personal best position is calculated as
   \[
   Y_i(t+1) = \begin{cases} 
   Y_i(t) & \text{if } f(X_i(t+1)) > f(Y_i(t)), \\
   X_i(t+1) & \text{if } f(X_i(t+1)) < f(Y_i(t)).
   \end{cases}
   \] (8)
   (b) The global best position is calculated as
   \[
   \hat{Y}(t) = \min\{ f(Y_1(t)), f(Y_2(t)), \ldots, f(Y_M(t)) \}. \] (9)
   where $M$ is the total number of particle in the swarm.

3. Update the velocity vector for each particle.
   \[
   V_{ij}(t+1) = wV_{ij}(t) + c_1r_{1j}(t)[Y_i(t) - X_{ij}(t)] + c_2r_{2j}(t)[\hat{Y}(t) - X_{ij}(t)]
   \] (10)

4. Update the position vector for each particle.
   \[
   X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1)
   \] (11)

5. Repeat steps 2-4 until meets the stop criteria.

Generally speaking, there are two versions of PSO, namely the gbest-PSO and lbest-PSO in open literatures which differ in the size of neighborhoods. For the gbest-PSO, the neighborhood for each particle is the entire swarm; while in lbest-PSO, the neighborhood is only a part of the whole swarm. Gbest-PSO converges faster than lbest-PSO due to the large particle connection and fast information sharing in the swarm. However, it is often trapped in a local optimum instead of recovering the global optimal solutions especially when the objective function has a large number of dimensions. An improved gbest-PSO is proposed in the next section to overcome the premature convergence of gbest-PSO.

3.2 Improved Gbest-PSO

In the newly proposed gbest-PSO algorithm, a new velocity scheme is introduced to control the exploration and exploitation ability of particles. In the classical gbest-PSO, the velocity is very large at the beginning of searching process which may cause the par-
particle leave the boundary of search domain or miss the promising area. While at the final stage of searching process, the velocity is relative small which may cause the particle trapped in a local optimum. The main mission of the new velocity scheme is to limit all particles flying in the effective domain and to enhance the local refinement capability of particles. There two contradictory effects are achieved by six velocity reduction factors and a re-entry method. These six velocity reduction factors are employed to reduce the velocity of particles, they are $\text{factor}(1) = 1.0, \text{factor}(2) = 0.618, \text{factor}(3) = 0.5, \text{factor}(4) = 0.382, \text{factor}(5) = 0.191 \text{ and factor}(6) = 0.0891$. The reduced velocity is obtained by multiplying $V(t)$ with the velocity reduction factor.

In the searching process, if the sixth reduced velocity, i.e., $V(t) \times \text{factor}(6)$ will keep the particles in the search space, the following local refinement searching is implemented. The positions $X'(t+1)$ corresponding to the six reduced velocities $V(t) \times \text{factor}(1:6)$ are first computed, respectively. Then fitness values of these six new positions are evaluated, and the position with the smallest fitness value will be selected as the new particle position, the corresponding reduced velocity will be selected as the new particle velocity. This process is equivalent to adjusting the particle’s position to a better place nearby.

On the other hand, if the sixth reduced velocity $i.e., V(t) \times \text{factor}(6)$ pushes particle out of the search boundary, a re-entry method will be implemented. If a particle leaves the search space from one side, it will re-enter from the other opposite side of the search domain. Thus, all the particles, no matter its velocity values, will be maintained in the search space which makes the particles explore more regions in the search space.

The pseudo-codes of the new velocity scheme are as follows,

\[
X'(t+1) = X(t) + \text{factor}(6) \times V(t)
\]
\[
\text{fitness}_t = f(X'(t+1))
\]
\[
\text{if } X'(t+1) \text{ in the search space then}
// find the best reduced velocity //
\text{do } I = 1, 6
X'(t+1) = X(t) + \text{factor}(I) \times V(t)
\text{if } f(X'(t+1)) \leq \text{fitness}_t \text{ then } V(t+1) = \text{factor}(I) \times V(t)
\text{enddo}
X'(t+1) = X(t) + V(t+1)
\text{else}
// re-entry boundary is implemented //
X'(t+1) = X(t) + V(t)
\text{if } X'(t+1) < X_{\min} \text{ then}
X(t+1) = X_{\max} - \text{abs}(\text{mod}(X_{\min} - X', X_{\max} - X_{\min}))
V(t+1) = X(t+1) - X(t)
\text{elseif } X'(t+1) > X_{\max} \text{ then}
X(t+1) = X_{\min} - \text{abs}(\text{mod}(X' - X_{\max}, X_{\max} - X_{\min}))
V(t+1) = X(t+1) - X(t)
\text{Endif}
\text{Endif}
\]

According to the above description, the procedure of the improved gbest-PSO is:
1. Initialize the swarm and control parameters.
   (a) Set the number of particles, set the acceleration constants $c_1$ and $c_2$ and inertia weight $w$.
   (b) Set the initial positions of each particle $X_i(0)$.
   (c) Set the initial velocity of each particle $V_i(0)$.
   (d) Set the initial values of personal best position $Y_i$ and global best position $\hat{Y}$.
2. Calculate the personal best position $Y_i$ and the global best position $\hat{Y}$.
3. Update the velocity and position by the new velocity scheme.
4. Repeat steps 2-3 until meets the stop criteria.

Five multimodal benchmark functions are employed to examine the efficiency of the proposed EPSO. The test functions, parameter domain and global optimum for each test problem are listed as follows:

$$f_1 = \sum_{i=1}^{D} \sin(x_i) + \sin\left(\frac{2x_i}{3}\right), x_i \in [3, 13], \text{optimum: } -1.21598D$$

$$f_2 = \sum_{i=1}^{D} \sin(x_i + x_{i-1}) + \sin\left(\frac{2x_i}{3}\right), x_i \in [3, 13], \text{optimum: } -2D$$

$$f_3 = -\sum_{i=1}^{D} x_i \sin(10\pi x_i), x_i \in [-1, 2], \text{optimum: } -1.85D$$

$$f_4 = 418.9828D + \sum_{i=1}^{D} [x_i \sin(\sqrt{|x_i|})], x_i \in [-500, 500], \text{optimum: } 0$$

$$f_5 = \sum_{i=1}^{D} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2, x_i \in [-5.12, 5.12], \text{optimum: } 0$$

The initial position of each particle $i$ is random generated by

$$X_i(0) = X_{i,\min} + \text{rand}(0)(X_{i,\max} - X_{i,\min}). \quad (12)$$

Table 1 shows the computational results for $f_1$-$f_5$ with a high dimension of $D = 100$. The computation results obtained by using other evolutional algorithms [18, 22] are also shown in the table for comparisons.

An overview of Table 1 shows that the estimated results obtained by EPSO outperform all the competitors significantly. For function $f_5$, the EPSO gives a very accurate result of $-121.598$, the classical PSO converges to a local minimum of $-42.71184$ which

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>EPSO</th>
<th>PSO ($w = 0.1$)</th>
<th>TLPSO</th>
<th>IEA</th>
<th>OEGA</th>
<th>UEGA</th>
<th>OGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$</td>
<td>121.598</td>
<td>42.71184</td>
<td>121.598</td>
<td>120.44</td>
<td>89.36</td>
<td>116.71</td>
<td>121.598</td>
</tr>
<tr>
<td>$f_2$</td>
<td>194.04441</td>
<td>124.51563</td>
<td>161.1661</td>
<td>153.15</td>
<td>94.59</td>
<td>89.49</td>
<td>139.71</td>
</tr>
<tr>
<td>$f_3$</td>
<td>185.00</td>
<td>35.12958</td>
<td>137.2322</td>
<td>131.31</td>
<td>76.72</td>
<td>74.50</td>
<td>115.48</td>
</tr>
<tr>
<td>$f_4$</td>
<td>118.42695</td>
<td>32867.0368</td>
<td>2828.6935</td>
<td>8011</td>
<td>24645</td>
<td>24160</td>
<td>12294</td>
</tr>
<tr>
<td>$f_5$</td>
<td>0.00667</td>
<td>98.92136</td>
<td>93.1306</td>
<td>2081</td>
<td>96556</td>
<td>89514</td>
<td>5282</td>
</tr>
</tbody>
</table>
is very far away from the benchmark value of $-121.598$ with $w = 0.1$. The TLPSO of Chen [22] also gives an excellent estimation for this case. TLPSO is a very powerful PSO variant, a two-layer particle structure is employed in TLPSO. The particles are generated in top layer and bottom layer, respectively. Each global best position in each swarm of the bottom layer is set to be the position of the particle in the top layer. Therefore, the global best position in the swarm of the top layer influences indirectly the particles of each swarm in the bottom layer so that the diversity of the particles increases to avoid trapping into a local optimum. According to the computational results of Chen [22], TLPSO is superior to the other evolutionary algorithms in the ability of finding the global optimum solutions. The computational results obtained with IEA and some genetic algorithm variants [18] are also listed in the Table 1. All these algorithms only converge approximately to the global best position. For function $f_2$, EPSO outperforms all other algorithms and gives an estimation of $-194.04441$; TLPSO gives the second best result. For function $f_3$, the EPSO algorithm finds the exact optimum solution of $-185.00$. For function $f_4$, the estimated minimum by EPSO is $118.42695$ and it is very far from the $0.0$. But it is still far better than the results obtained from other algorithms. Again, the TLPSO gives the second best result of $2828.6935$. The classical PSO gives the worst result of $82867.0368$. For function $f_5$, EPSO algorithm gives the best solution of $0.00667$.

From an overview of table, the EPSO proposed in this paper overcomes the shortcomings of earlier convergence of other PSO algorithms for high dimensional multimodal optimization problems successfully. In the next section, the EPSO algorithm proposed in this paper is used to solve the inverse convection heat transfer problem.

4. INVERSE PROBLEM

For the inverse problem, the temperature profile at the inlet is regarded as unknown while the other parameters are considered as already known. We are going to identify the unknown temperature $T_0(y)$ by using the temperature measurements in the channel flow. Let the temperature measurements taken at some appropriate locations within the flow be denoted by $T_{i,\text{mea}}$ and let $T_{i,\text{est}}$ denote the solutions of the direct problem at the thermocouples position with current estimation of unknown temperature $T_0(y)$. The unknown temperature $T_0(y)$ can be resolved by requiring an exact equivalency between the measured temperature $T_{i,\text{mea}}$ and the calculated temperature $T_{i,\text{est}}$. But in practice, the inverse problem should be solved in a least square way due to the ill posed nature of the inverse problem. Then the inverse problem treated in this paper is defined as follows:

find a temperature $T_0(y)$ which minimizes the object function $J$ defined by

$$J = \sum_{i=1}^{I} \left[ T_{i,\text{mea}} - T_{i,\text{est}} \right]^2$$

(13)

where $I$ represents the total number of the measuring points. The minimization of object function $J$ is achieved by the classical gbest-PSO and the improved gbest-PSO algorithm proposed in this paper, respectively. The dimension of the unknown temperature profile is equal to the number of grid points in the $y$ direction, $D = 13$. 

5. COMPUTATIONAL RESULTS

To examine the performance of gbest-PSO for the solution of inverse problems, two test cases as shown in Fig. 2 are considered. Example (1) is a smooth function and it represents the easiest test; while Example (2) is a difficult function with 2 sharp corners. The accuracy of the estimation is quantified by the following averaged relative error $E$

$$E = \frac{\sum_{i=1}^{M} |T_{i,\text{est}} - T_{i,\text{exa}}|}{M} \times 100\%$$  (14)

where $M$ is the number of grid points in the $y$ direction, $T_{i,\text{est}}$ represents the heat flux estimated by gbest-PSO algorithm and $T_{i,\text{exa}}$ represents the exact heat flux.

The performance of the two gbest-PSO algorithms is compared with the same initial conditions and same control parameters. The initial conditions for the improved gbest-PSO and the classical gbest-PSO are as follows: the positive acceleration constants: $c_1 = 2.0, c_2 = 2.0$. The parameter domain of the unknown heat flux is set to $[0, 100]$. The initial value of the unknown temperature, i.e. the position of each particle $i$ is random generated by

$$X_i(0) = X_{i,\text{min}} + \text{rand}() (X_{i,\text{max}} - X_{i,\text{min}}).$$  (15)

The initial velocity of each particle $V_i(0)$ is set as follow

$$V_i(0) = \text{rand}() (X_{i,\text{max}} - X_{i,\text{min}}).$$  (16)

Both the initial values of personal best position $Y_i$ and global best position $\hat{Y}$ are set to $X_i(0)$.

Fig. 2. Various shapes of temperature functions $T(y)$ used to examine the performance of the inverse problem solvers.
We first consider an idealized situation in which there are no measurement errors. The computational results obtained by the improved gbest-PSO with only 5 particles are shown in Fig. 3 for example 1. Fig. 3 (a) shows the best fitness found by particles as a function of computational time. Thirteen measurement points are placed at every grid point along the y-axis with an x-coordinate of $x_{\text{mea}} = 4\Delta x$. The value of the best fitness is reduced to an order of $10^{-10}$ after only about 100 seconds in a very smooth way. It is evident that the improved gbest-PSO algorithm proposed in this paper can find better solutions consecutively which proves it has a strong anti-local trap capability and a good exploration capability. Fig. 3 (b) presents the estimated temperature profile obtained by the improved gbest-PSO, the benchmark result is also shown in figure for comparisons. The estimated temperature are in excellent agreement with the exact one with a relative error of $E = 0.0\%$. Figs. 3 (c) and (d) present the effects of measurement position on the accuracy of estimation. The measurement points are placed at $x_{\text{mea}} = 10\Delta x$ in this case. It is obvious that the prediction shows some deficiency near the bottom plate, since the measurement points in this region are placed in thermal boundary layer developing along the bottom plate, and they feel little about any changes occurred at the inlet. But the quality of the estimation can still be regarded as very good, the relative error is only 1.85%.
Fig. 4 shows the computational results obtained by improved gbest-PSO for example 2. Again, the improved gbest-PSO gives an excellent estimation when the measurement points are placed at $x_{\text{mea}} = 4\Delta x$ as shown in Fig. 4 (a). The accuracy of estimation decreases to 3.38% when the measurement points shift to $x_{\text{mea}} = 10\Delta x$. Large prediction errors are observed near the up plate and bottom plate simultaneously. For example 2, there is a sharp increase in the inlet temperature profile near $y = 0.035$, then the sensitivity coefficient in this region is decreased greatly by the heat transfer occurred in the $y$ direction.

The computational results for the two examples by the classical gbest-PSO are shown in Fig. 5. Figs. 5 (a) and (c) give the best fitness found by particles as a function of computational time. The convergence characteristic of the classical gbest-PSO is very different from that of the improved gbest-PSO as shown in Figs. 4 (a) and (b). The value of best fitness decreases in a step way. It is obvious that in the classical gbest-PSO, most of the time is used to fly around in the search domain, and it does not has a mechanism to
conduct further local search after a better position is found. The value of the best fitness dropped to an order of $10^3$ after 400 seconds. While it only use no more than 20 seconds for the improved gbest-PSO to reach the same order. The new velocity scheme introduced in the improved gbest-PSO algorithm not only has the ability to limit all the particles flying in the effective search domain, but also can deeply refine the local domain of each particle by the six velocity reduction factors. The smooth best fitness curve as shown in Figs. 4 (a) and (b) give a strong demonstration about the high efficiency characteristic of the improved gbest-PSO algorithm. The estimated results after 12000 seconds are shown in Figs. 5 (b) and (d) for examples 1 and 2, respectively. The estimated results by the classical gbest-PSO deviate from the exact one significantly.

Now, let us examine the effects of measurement error on the accuracy of the improved gbest-PSO. In practice, the temperature measurements always contain some degrees of measurement error. As real experimental data are not available, in this study, we generate the simulated measurement data by adding the random error to the exact temperature. Here, we assume the measurement error follows the normal distribution, then the simulated measurement data can be expressed in the following way as

$$T_{\text{mea}} = T_{\text{est}} + \sigma T_{\text{est}} \xi / 2.576$$  \hspace{1cm} (17)$$

where $\sigma$ is a normal distributed random number with zero mean and unit standard deviation, 2.576 comes from the fact that 99% of the population lying in the range of $\pm 2.576$. In this paper, two measurement errors are investigated, $\xi = 3\%$ and $\xi = 10\%$.

Fig. 6 displays the computational results when the input data contains a measurement error of $\xi = 3\%$, the thermal sensors are places at $x_{\text{mea}} = 4\Delta x$. The prediction is very good in general; the relative error $E$ takes a value of 1.39% for example 1 and 1.54% for example 2, respectively. Fig. 7 shows the results of the inverse analysis for the two test cases with a measurement error of $\xi = 10\%$. The computational results exhibits some noise randomly scattered around the exact temperature profile. But, the recovered profile still can be regarded as very acceptable. Fig. 8 studies the effects of measurement position on the accuracy of estimation with $\xi = 10\%$. As expected, the prediction is less accu-
INLET TEMPERATURE ESTIMATION USING GBEST-PSO

(a) Estimated temperature profile for example 1, $x_{\text{mea}} = 4\Delta x$. (b) Estimated temperature profile for example 2, $x_{\text{mea}} = 4\Delta x$.

Fig. 6. Computational results obtained by the improved gbest-PSO, $\xi = 3\%$.

(a) Estimated temperature profile for example 1. (b) Estimated temperature profile for example 2.

Fig. 7. Computational results obtained by the improved gbest-PSO, $\xi = 10\%$, $x_{\text{mea}} = 4\Delta x$.

(a) Estimated temperature profile for example 1. (b) Estimated temperature profile for example 2.

Fig. 8. Computational results obtained by the improved gbest-PSO, $\xi = 10\%$, $x_{\text{mea}} = 10\Delta x$.
rate when the measurement position is shifted to $x_{\text{mea}} = 10\Delta x$, there is more large oscillation scattered around the exact profile. But overall, the improved gbest-PSO still gives physical reasonable results.

6. CONCLUSION

In this paper, an improved gbest-PSO is proposed to overcome the earlier convergence of classical gbest-PSO. A new velocity scheme is introduced in the improved gbest-PSO algorithm which not only can limit particles flying in the effective searching domain but also enhance the local refine capability of the particles. An inverse convection heat transfer problem of estimating the unknown inlet temperature profile in a plate channel flow is solved with the improved gbest-PSO and the classical gbest-PSO. Two test examples are used to examine the performance of the PSO algorithm as an inverse problem solver. Analysis of the computational results shows that the improved gbest-PSO has a powerful anti-local trap capability, and the estimated unknown temperature can converge to the exact one in a very smooth way when no measurement errors exist in input data. While the classical gbest-PSO can not give any significant results even with errorless input data. The effects of the measurement errors on accuracy of the estimation are also studied. It is shown that reasonable results can be obtained with the improved gbest-PSO algorithm proposed in this paper.

REFERENCES

8. C. K. Chen, L. W. Wu, and Y. T. Yang, “Estimation of time-varying inlet tempera-
Peng Ding (鹏) received his Ph.D. degree in Department of Power and Thermal Engineering from Xi’an Jiaotong University, and his B.S. degree in 2009 and 2005, respectively. He is a Professor of the Department of Thermal Engineering of Petroleum University of China. His areas of research include computational method for fluid flow and heat transfer, reduced order model, inverse heat transfer.

Minghai Xu (徐海) received his Ph.D. degree in Department of Power and Thermal Engineering from Xi’an Jiaotong University, and his B.S. degree in Petroleum engineering from Petroleum University of China in 1989. He is a Professor of the Department of Thermal Engineering of Petroleum University of China. His areas of research include enhanced heat transfer, heavy oil thermal recovery and computational method of fluid flow and heat transfer.

Dongliang Sun (孙) received his Ph.D. degree from the Department of Energy and Power Engineering of Xi’an Jiaotong University in 2009. He is a Lecturer of the Department of Renewable Energy of North China Electric Power University. His areas of research include numerical heat transfer, enhanced heat transfer and multi-phase flow.