Enhancing the Firework Algorithm ecosystem for the reconstruction of the HTC function

Z Fried¹, Felde I² and Szénási S²
¹,²John von Neumann Faculty of Informatics Óbuda University Budapest, Hungary

E-mail: ¹fried.zoltan@stud.uni-obuda.hu

Abstract. The heat transfer coefficients reconstruction could be solved using several ways and configurations. The Inverse Heat Transfer Problems (IHCP) collects all kind of algorithms which try to solve the Heat Transfer Problems. These algorithms try to calculate the heat transfer coefficients inversely. There is some solution for solving this numerical method, but the bio-inspired solutions are better here even though they have some limitations because of this is a kind of genetic one. Using the Fireworks Algorithm (FWA), it is possible to introduce new methods, and modify the original FWA controlling system and parameters in an easy to adapt to this problem. Extensions are suggested with the major limitation of the genetic methods which solved by the proposed one, and with the faster computation of the new modified method.

1. Introduction

The heat transfer phenomena have been described by the principles of the thermal boundary conditions. These are taken place during the heat treatment process. The Heat Transfer Coefficient (HTC) function has been in time by the heat exchange between the examined surface and the surrounding medium. To solve the one-dimensional IHCP in the workpiece, we are using the time-varying HTC only, and the numerical solutions could approximate it. Subtracting an ambient temperature over the boundary temperature as a dimensionless temperature is a possible solution for a system which shifts the function procedure without any integral transforms. Determination of HTC will become a typical Inverse Heat Conduction Problem if using the boundary conditions [1–5]. The IHCP systems are using the thermal collected samples to predict the HTC functions of the measured data of the rod by simulation at the location of the rod. The IHCP solutions usually are using various optimization methods, where the objective function has been defined as the minimum of the differences between the recorded and calculated temperature data [6, 7]. The genetic algorithms are using successfully for solving many types of IHCP cases for example but most of the cases, the genetic algorithms can approximate one or more looked for results but are not able to calculate the expected result [8–10]. For this reason, hybridized solutions need to use to get the expected result. The classification of the result depends on the expectation. The Fireworks Algorithm (FWA) became popular in the last 10+ years due to its power versatility and flexibility. It makes a perfect balance between speed, convergence, cooperation and diversity. Many articles have been published to analyse the FWA technique in inverse heat conduction problems. They have been shown that FWA could reduce the time of processing, reach an expected result by an easy algorithms adaptation process for
the task opposite the similar classical methods. In this article an IHCP analysis of the time-dependent
and one-dimensional HTC(t) has been calculated [11–15]. The cooling curves were measured in the
centerline of the cylinder, and it has been applied to the algorithms to get the inverse heat transfer
computation. The objective function has been defined as the quadratic residual between the
measurements and the calculated temperatures, and it should be minimized at all. The optimization
technique has been implemented in CPU and into GPU architecture for parallelized computation for
better performance.

2. The inverse heat transfer problem
The temperatures of a cylindrical workpiece surface have been measured during the heat transfer
process at several p points inside the boundary. The variations of the temperature of the thermal
boundary conditions determine the solution of the IHCP [6, 7, 12]. Calling the $T_k^m$ as the measured
and $T_k^c$ as the calculated temperature at the measure points the objective function can be followed in
Eq. (1). The inverse heat conduction problem solution is received by minimizing the objective
function.

$$S = \sum_{k=1}^{p} (T_k^m - T_k^c)^2 \rightarrow \min$$

According to Eq. (1), the IHCP analysis has been transformed into a solvable numerical optimization
problem.

3. Thermal field calculation
The axisymmetric heat conduction model has been considered due to cylindrical workpiece to estimate
the temperature distribution. The heat capacity, density and thermal conductivity were assumed as a
function of the temperature respectively $C_p(T)$, $\rho(T)$ and $k(T)$. The Eq. (2) describes the mathematical
formulation of the nonlinear transient heat conduction problem.

$$\frac{\partial}{\partial r} (k \frac{\partial T}{\partial r}) + \frac{k}{r} \frac{\partial T}{\partial r} + \frac{\partial}{\partial z} (k \frac{\partial T}{\partial z}) + q_v = \rho \cdot C_p \cdot \frac{\partial T}{\partial t}$$

using the following boundary conditions Eq. (3) and Eq. (4):

$$T(r, z, t = 0) = T_0$$

and

$$k \frac{\partial T}{\partial r} |_{r=R} + k \frac{\partial T}{\partial z} |_{0 \leq z \leq Z} = \text{HTC}(z, t)[T_q - T(r, z, t)]$$

where

- R is the radius of the cylinder
- L is the length of the cylinder
- HTC(t, z) is the heat transfer coefficient
- k is the thermal conductivity
- $C_p$ is the heat capacity
- $r$ and $z$ are local coordinates
- t is the time
- $\rho$ is the density of the object
- $T_0$ is the initial temperature of the work-piece
- $T_q$ is the temperature of the cooling medium

and the z is constant if the calculation is one-dimensional.

In this article the material phase transformation does not calculated during the measuring process
and the latent heat is not generated ($q_v = 0$).
4. The Fireworks algorithm

The FWA model [16], by definition, contains a firework and several calculating points (sparks) around the firework. The process of the solution is based on the movement rules of the sparks and firework. In our implementation, the firework is a spark also. Each spark is moving around in the searching space, which is implemented in $\mathbb{R}^n$ space. The [15] article is used many times in this article. This article introduces a new statistical solution to determine a direction from the actual position of the spark, which brings a higher possibility to decrease the objective function.

The number of dimensions represents the number of points of the HTC function. Each spark represents a point in the searching space like $X_i = (x_{i1}, x_{i2}, \ldots, x_{iN})$ which can be transferred to an HTC function as well, and every spark could be a potential solution of the IHCP problem. The possible lowest value of the objective function is determining the solution. Every spark has a limited history for storing earlier calculated objective functions and the selected HTC function as the $\vec{x}_k$.

The next position of every spark is defined by

$$\vec{d} = \vec{A} \odot \vec{r}$$  \hspace{1cm} (5)

and

$$\vec{X}_{new} = \vec{X} + \vec{d}$$  \hspace{1cm} (6)

where $\vec{r}$ unit vector is the direction of the movement and it depends on the type of the spar, $\vec{A}$ is the amplitude vector (each dimension of the displacement vector has an amplitude value), $\vec{X}$ is the point of the spark in the searching space (vector), $\vec{d}$ is the displacement vector (by Hadamard product).

Every type of sparks have:

- a moving base which is defining the movement base position
- collect own history as storing all own information which was calculated the latest 2 interactions earlier. This information will only be used by a new type of spark in the 4.10 section and the amplitude correction coefficient in the 4.2 section
- the number of sparks in an iteration
- two moving profiles. These moving profiles determine the moving rules of the spark:
  - init state moving profile: this profile is only working in the init state and put the spark to the init position.
  - next state moving profile: this profile is using after the init state all-time

In this article, we have modified the ecosystem of standard FWA model in some cases. For example, in the standard algorithm, there are two types of spark [17, 18]. One of them is the ‘explosion spark’, and the other is the ‘gaussian spark’. We are using both of them, but the movement rules have been changed, and we are introducing three new types of sparks as ‘firework spark’, ‘best spark’ and ‘blind spark’.

4.1. Firework population count

In the standard FWA model, there are some firework populations running in the searching space parallelly. This is necessary to avoid the calculation to stuck in local minima and rerun the calculation many times. Moreover, the original definition describes that these populations can contact to each other. In our implementation, the count of the population has been calculated by Eq. (7).

$$\ln(F_w^{count}) = \log_{10}n$$  \hspace{1cm} (7)

where $F_w^{count}$ is the firework population count, the $n$ is the dimension.
4.2. Amplitude coefficient

This coefficient is calculated for controlling the amplitude value due to direction of fitness changes between the actual and last iteration. The controlling mechanism is based on the Fibonacci numbers. The Fibonacci numbers have been defined as

\[ F_0 = 0, F_1 = 1, F_q = F_{q-1} + F_{q-2} \]  

Define the \( o \) counter in \( \mathbb{N} \) which has minimum \(-20\) and a maximum \(20\) value. This \( o \) counter determines the \( q \) index from Fibonacci numbers Eq. (8) as \( q = |o| \) where \( o \) counter steps by \(+1\) or \(-1\) forward in each iteration depends on the type of spark as follows

- **firework spark**: the \( o \) counter is starting from 1 to 20. Every iteration the \( o \) is stepping by \(+1\) towards and if the \( o > 20 \) the \( o \) starting from 1 again.
- **random spark**: as same as the firework spark.
- **gaussian spark**: there is not any usage due to 4.7 section
- **best spark**: for selecting the next \( o \) number, the calculation is based on the historical statistical calculation [15]. The \( o \) is increasing by 1 if the fitness decreasing. If fitness does not change over 3 steps the \( o \) counter is increasing by 1, otherwise the \( o \) counter does not change. If the \( o > 20 \) or \( o < 1 \) the \( o \) set to 1.
- **blind spark**: in this case the \( o \) counter can be a negative number too. The \( o \) is increasing by 1 if the fitness decreasing and the \( o \) is decreasing by 1 if the fitness increasing. If fitness does not change over 3 steps the \( o \) counter is increasing by 1, otherwise the \( o \) counter does not change. If the \( o > 20 \) the \( o = 20 \) or if the \( o < -20 \) the \( o = -20 \).

The \( o \) is starting from 1 in the initialisation state of the algorithm. Based on the above definitions the \( A_{\text{coef}} \) is calculating by the Eq. (9) in every iteration

\[
A_{\text{coef}} = \begin{cases} 
F_o + F_{o-1} & \text{if } o > 0 \\
F_{|o|} + F_{|o|-1} & \text{if } o < 0 \\
\text{skip} & \text{if } o = 0 
\end{cases} 
\]  

4.3. Amplitude maximum and minimum value calculation

The \( \tilde{A}_{\text{max}} \) has been defined by Eq. (10) as the maximum value of the \( A_{\text{max}} \) value Eq. (11). In our terminology the \( A_{\text{max}} \) value is changing every iteration and there is a maximum value which can be depend on the size of searching space and the number of firework populations running in the same time at the same iteration. It is a theoretically maximum value of the \( A \) amplitude.

\[
\tilde{A}_{\text{max}} = \frac{X_{\text{max}} - X_{\text{min}}}{2 \cdot F_{\text{count}}} 
\]  

\[
A_{\text{max}} = \frac{\tilde{A}_{\text{max}}}{A_{\text{coef}}} \cdot \frac{\log_2(F)}{\log_2(F_{\text{max}})} 
\]  

where the \( F \) is the actual fitness value of the spark and the \( F_{\text{max}} \) is the maximum fitness value of the same spark in the whole calculation iteration. The \( A_{\text{min}} \) always has been calculated by \( A_{\text{max}} \) in Eq. (12)

\[
A_{\text{min}} = A_{\text{max}} \cdot \frac{3 - \sqrt{5}}{2} 
\]
4.4. Amplitude boundary calculation
The Amplitude is always $\geq 0$. If the new position run out of the searching space in the k dimension, the amplitude in that dimension will be reduced with an uniform distributed random value between 0 and 1, where 1 is exactly the new position when it is reaching the boundary as the following condition

$$A^k_{\text{new}} = \begin{cases} A^k_u & \text{if } X^k_{\text{new}} > X^k_{\text{max}} \\ A^k_l & \text{if } X^k_{\text{new}} < X^k_{\text{min}} \\ A^k_{\text{new}} & \text{if } X^k_{\text{min}} \leq X^k_{\text{new}} \leq X^k_{\text{max}} \end{cases} \quad (13)$$

where $A^k_u$ is

$$A^k_u = \text{rand}(0, 1) \cdot (X^k_{\text{max}} - X^k_{\text{new}}) \quad (14)$$

$A^k_l$ is

$$A^k_l = \text{rand}(0, 1) \cdot (X^k_{\text{new}} - X^k_{\text{min}}) \quad (15)$$

and k = 1...n dimension index and the $X^k_{\text{min}}$ and the $X^k_{\text{max}}$ is the searching space boundary minimum and maximum values in each dimension.

4.5. Iteration to live (ITL)
This value was developed to controlling iteration count between the borning and dying event. The main theory is if a spark can decrease the fitness value many times they need to set a high ITL credit and if a spark cannot decrease the fitness value they need to set a low ITL credit. For this reason, if a spark has been born, they have 10 ITL credits. The rule of the ITL credit has been described by the Eq. (16) If the ITL credit reaches the zero, the spark will have died. This calculation is based on the [15] article. The ITL credit is changing as follows

$$ITL_{i+1} = \begin{cases} ITL_i + 1 & \text{if } f(x_{i+1}) - f(x_i) < 0 \\ ITL_i - 1 & \text{if } f(x_{i+1}) - f(x_i) > 0 \\ ITL_i - \frac{1}{2} & \text{if } f(x_{i+1}) - f(x_i) = 0 \end{cases} \quad (16)$$

where i is the iteration number and the $f(x_i)$ is the fitness value of a spark in the $i^{th}$ iteration.

4.6. Explosion spark
This type of spark position calculation is based on a uniformly distributed random number which can be −1 or 1 as a direction in each dimension and a uniformly distributed random number as the distance from firework spark position between the $A^k_{\text{min}}$ and $A^k_{\text{max}}$ in each dimension also. The $A^k_{\text{max}}$ and $A^k_{\text{min}}$ are based on the firework spark (see the 4.9 section) amplitude calculation.

4.7. Gaussian spark
This type of spark position calculation is based on a normal distributed random number as a distance from the actual position where the mean is 1, and stddev is 0.293. The direction is the position of the best spark or a uniformly distributed random selected another spark in the population.

4.8. Best spark
The best position means the lowest value of the objective function through the all process. This position has been held by this spark. This spark in every iteration is moving towards the position of spark, which has the best value of the objective function. Although the $A_{\text{max}}$ is not used in this type of spark, the value needs to calculate because the firework spark uses it.
4.9. Firework spark
Based on the [15] article the firework spark should be placed to close the Best spark in a random point. The $A_{\text{max}}$ calculation is modified as the 4.3 section describes as follow
\begin{equation}
A_{\text{max}} = A_{\text{Best min}} \cdot \frac{3 - \sqrt{5}}{2}
\end{equation}
where the $A_{\text{Best min}}$ is the $A_{\text{min}}$ of the best spark value.

4.10. Blind spark
This kind of spark has a complex displacement strategy and rules which are based on the [15] article. This is a kind of hill climbing attempt to minimize the objective function. The spark name comes from the way that the blind people walk from the top of the hill to valley. This spark uses its walk history to calculate the next direction of the way. This spark has been started from the parent $\vec{x}_{p k+1}$ position, and the spark inherits its parent spark history too. Any spark in the new position there are two possibilities between the last $\vec{x}_{k-1}$ and the new $\vec{x}_{k}$ position. If
\begin{equation}
f_p(x_k) < f_p(x_{k-1})
\end{equation}
then the spark stay in the new position and the o counter for $A_{\text{coef}}$ is increasing by 1. If
\begin{equation}
f_p(x_k) \geq f_p(x_{k-1})
\end{equation}
then the spark go back to the previous position in the next iteration and the o counter is decreasing by 1. The spark go forward if fitness speed [15] is decreasing but the o counter is decreasing also by 1. The $f_p(x_k)$ is the parent spark $\vec{x}_k$ position of k\textsuperscript{th} iteration of objective function.
In the first step of the blind spark uses its parent spark history. The first step direction and distance are as same as its parent spark moved towards in the latest iteration by
\begin{equation}
\vec{x}_{b|k+1} = \vec{x}_{p|k} - \vec{x}_{p|k-1}
\end{equation}
where $\vec{x}_{b|k+1}$ is the blind spark position in the (k + 1)\textsuperscript{th} iteration, $\vec{x}_{p|k-1}$ is its parent spark previous (k−1)\textsuperscript{th} position. To calculate the next position of blind spark in every further iteration are based on the following Eqs. which are using the Eq. (18) and Eq. (19):
\begin{equation}
\vec{x}_{k+1} = \vec{x}_{k} - \vec{x}_{k-1}
\end{equation}
\begin{equation}
\vec{x}_{b|k+1} = \begin{cases} 
\vec{r}(\vec{x}_{k+1} \pm (\frac{45}{2})^\circ) & \text{if } f(x_{k-1}) > f(x_k) \\
-\vec{x}_{k+1} & \text{if } f(x_{k-1}) \leq f(x_k)
\end{cases}
\end{equation}
where the $\vec{r}$ is an unified random vector around $\vec{x}_{k+1}$ in $\pm \frac{45}{2}$.
To avoid creating lots of or irrelevant new blind sparks it will not born if:
- the possible parent spark is on top of the hill while the best spark is in the valley
\begin{equation}
\frac{f(x^k_p)}{f(x^k_{\text{best}})} > 10
\end{equation}
and
\begin{equation}
\frac{f(x^k_p)}{T_N} - \frac{f(x^k_{\text{best}})}{T_N} > 1
\end{equation}
there is not free slot to store the new blind spark. The maximum numbers of blind sparks are defined as $2 \cdot (N_{e}^{\text{max}} + N_{g}^{\text{max}})$ where $N_{e}^{\text{max}}$ is the maximum number of explosion sparks and $N_{g}^{\text{max}}$ is the maximum number of gaussian sparks.

where the $T_{N}$ is the number of calculated temperature points on the workpiece, $f(x_{k}^{p})$ is the possible parent spark fitness value at stage $k$ and the $f(x_{k}^{\text{best}})$ is the best spark fitness value at stage $k$.

5. Comparing the proposed FWA and the original FWA

To compare the above proposed FWA and the standard FWA, we define a theoretical HTC* function with 50 points in 60 sec in the time axes (Fig. 1). The workpiece radius was 6.25cm. The new type of FWA used 50 explosion sparks and 20 gaussian sparks. The maximum number of blind sparks were 140. The original FWA used 150 explosion sparks and 60 gaussian sparks. The iterations were stopped at 1000 in case of proposed FWA and 10000 in case of standard FWA. The simulation was run ten times. The shape of the calculated HTC has been checked at the end of each run, and that ran was selected where the shape was the best. Therefore Fig. 2 shows the calculated HTCs. Fig. 3 shows the convergence speeds. This figure shows how the fitness value decreasing depends on the iteration. The ‘+’ line shows the proposed FWA, the ‘o’ line shows the standard one. The Fig. 4 shows a zoom into the Fig. 3 near the (0,0) corner. The ‘-’ line is the theoretical HTC*, the ‘+’ line shows calculated HTC based on the proposed FWA and the ‘o’ line shows calculated HTC based on the standard FWA. We think that the standard FWA could produce as same as HTC shape like the proposed one, but it takes much more time and computation effort.

Figure 1. Theoretical HTC function

Figure 2. The theoretical HTC and Calculated HTC functions by standard and proposed FWA

Figure 3. Fitness values by standard and proposed FWA

Figure 4. Fitness values by standard and proposed FWA (zoomed version)
6. Conclusions

The Fireworks Algorithm for estimating the Heat Transfer Coefficient was prepared. After analysis of the Fireworks Algorithm has turned out, there is possible to integrate a new type of sparks or modify the exists one. The measurement of the prepared FWA shows that the inverse thermal analysis by a numerical process speed could be fast, and the accuracy could be precise. Comparing the prepared and the original FWA shows that the proposed FWA is more accurate than the original. We need to continue the investigation to make the algorithm more precise and faster.

Acknowledgements

We acknowledge the financial support of this work by the Hungarian State and the European Union under the EFOP-3.6.1-16-2016-00010 project and Hungarian-Japanese bilateral Scientific and Technological (TÉT 16-1-2016-0190) project. The authors of this work also acknowledge the financial support of the Hungarian State for supported by UNKP-19-3-I-OE-35 new national Excellence program of the ministry for innovation and technology.

References

[1] Heinrich B and Hofmann B B J V B B S C C R j 1987 ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik 67 212–213
[2] Ivanov V V and Salomatov V V 1965 9 63–64 ISSN 0022-0841
[3] Ivanov V V and Salomatov V V 1966 11 151–152 ISSN 0022-0841
[4] Kozlov V N 1970 18 100–104 ISSN 0022-0841
[5] Postol’nik Y S 1970 18 233–238 ISSN 0022-0841
[6] Alifanov O 1994 Inverse heat transfer problems International series in heat and mass transfer (Springer-Verlag) ISBN 9783540536796
[7] Özisik M N and Orlande H R B 2000 Inverse Heat Transfer: Fundamentals and Applications (Taylor &Francis)
[8] Ozisik M 2007 International Journal of Heat and Mass Transfer 50 1706–1714 ISSN 0017-9310
[9] Kim K W and Baek S W 2004 Numerical Heat Transfer, Part A: Applications 46 367–381
[10] Felde I and Szénási S 2016 Int. J. Microstructure and Materials Properties 11 288–300
[11] Chiba R 2018 Mathematical Problems in Engineering 2018 1–11
[12] Chantasiriwan S 1999 42 4275–4285 ISSN 0017-9310
[13] Tian N, Zhu L and Lai C H 2014 8 233–248 ISSN 1748-3018
[14] Fried Z, Szénási S and Felde I 2018 (IEEE) pp 000099–000104 ISBN 978-1-7281-1118-6 ISSN 2380-8586
[15] Fried Z, Szénási S and Felde I 2019 (IEEE) pp 000305–000308 ISBN 978-1-7281-0685-4
[16] Tan Y, Yu C, Zheng S and Ding K 2013 4 39–70 ISSN 1947-9263
[17] Zheng S, Janecek A, Tan Y, 2013 Enhanced Fireworks Algorithm (IEEE) pp 002069–002077 ISBN 978-1-4799-0454-9
[18] Tan Y 2015 Fireworks algorithm ISBN 978-3-662-46352-9