Bio-inspired methods for fast and robust arrangement of thermoelectric modulus

Ahmad Mozaffari*, Ali M. Goudarzi and Alireza Fathi

Department of Mechanical Engineering, Babol University of Technology, P.O. Box 484, Babol, Iran
E-mail: amozaffari10@yahoo.com
E-mail: goudarzi@nit.ac.ir
E-mail: fathi@nit.ac.ir
*Corresponding author

Pendar Samadian

Product Control Section, AAA Linen C.O., P.O. Box 34957, London, England
E-mail: pendar.samadian@gmail.com

Abstract: This paper aims to evaluate the ability of some well-known bio-inspired metaheuristics for optimal arrangement of thermoelectric cells mounted in a thermal component. In real life applications, proper arrangement of thermoelectric modules plays a pivotal role by maximising the generated electricity. However, some defects such as the increase in total maintenance cost is often associated with the use of thermoelectric cells. Hence, it is mandatory to contrive a policy which guarantees the maximum electricity generation while keeps the maintenance cost in lowest level. Here, authors use both adaptive neuro-fuzzy inference system (ANFIS) and experimental data to model the power generation and maintenance cost of thermoelectric cells. At the next step, they engage some famous bio-inspired metaheuristic algorithms, i.e., bee algorithm (BA), particle swarm optimisation (PSO) and the great salmon run (TGSR) to arrange the thermoelectric cells in a cost effective manner. The gained results indicate that the proposed algorithms are highly capable to find an efficient arrangement for thermoelectric cells within a rational duration. Besides, through independent runs, it is observed that metaheuristics show acceptable robustness for the current case study.

Keywords: thermoelectric arrangement; bio-inspired metaheuristics; engineering optimisation robustness analysis; neural networks.


Biographical notes: Ahmad Mozaffari was graduated from National Organization for Development of Exceptional Talents (NODET) in 2008. He received his BSc in Mechanical Engineering from Babol University of Technology (BUT), Iran in 2012. His research interests encompass engineering optimisation, applied and theoretical soft computing, natural and bio inspiration, intelligent computation and system identification. He proposed ‘mutable smart bee’ and ‘the great salmon run’ algorithms which have been successfully applied on some real-life optimisation problems. He is the author of 15 refereed journal papers and two conference papers.

Ali M. Goudarzi received his BE in Manufacturing and Production Engineering from Tabriz University, Tabriz, Iran in 1990. He fulfilled his graduate study (MTech) in Robotic Engineering at the University of Pierre and Mary Curie (Paris IV), Paris, France in 1995, and received his PhD in Production and Computer Engineering from Evry University, Evry, France in 2001. Since 2004, he has been working as a faculty member in the Department of Mechanical Engineering at Babol University of Technology (BUT), Babol, Iran. His research interests are in the fields of robotics, biomechanics, applied mechanics and renewable energy.

Alireza Fathi received his BSc and MSc in Mechanical Engineering from Sharif University of Technology, Tehran, Iran in 2000 and 2002, respectively; and PhD in Applied Design from the same university, in a common project with University of Waterloo, ON, Canada in 2007. He is
1 Introduction

In recent decades, bio-inspired metaheuristic algorithms indicated increasing contributions in several engineering tasks especially in complex systems handling and applied optimisation (Parpinelli and Lopes, 2011; Srivastava et al., 2012; Chatterjee et al., 2012). In the light of these contributions, several researchers have mainly concentrated on the constructive traits of these algorithms in order to boost their performance. Several cues such as statistical mathematics, parallel programming and natural inspiration have come to the aid of metaheuristic scientists. Whilst some reports addressed the application of novel natural inspired metaheuristics (Yang and Deb, 2012; Yang, 2008, 2009; Shah-Hosseini, 2009), other reports pursued different mathematical formulations to alleviate the complexity of stochastic exploration (Zhong et al., 2012; Mozaffari et al., 2012c). Among the natural inspired algorithms, ones that follow the theory of evolution are called evolutionary algorithms (EAs). After the proposition of EA concept, several evolutionary-based metaheuristics such as genetic algorithm (GA) (Goldberg, 2005), differential evolutionary algorithm (DEA) (Storn and Price, 1997), imperial competitive algorithm (ICA) (Khabbazi et al., 2009) and evolutionary strategy (ES) (Hansen and Kern, 2004) have been proposed in literatures. Swarm-based metaheuristics are other common types of metaheuristics which follow the interactivity and intelligence of swarms to optimise the problem at hand (Poli et al., 2007; Yang and Deb, 2009, 2010). Generally, there exist a wide variety of swarm-based algorithms which mimic the natural behaviour of insects and animals such as ants, glow-worms, fishes, bees, fireflies, frogs and etc. (Parpinelli and Lopes, 2011). There are also some other types of metaheuristics that were implemented based on the physical phenomena in nature. Simulated annealing (SA) (Kirkpatrick et al., 1983) is one of the most popular metaheuristic approaches inspired by the annealing happened in metals. Gravitation search algorithm (GSA) (Rashedi et al., 2009) is another heuristic algorithm that is based on the Newton’s law of gravity. Harmony search (HS) (Geem et al., 2001) is an optimisation method that imitates the music improvisation process where the musicians improvise their instrument pitches to gain a perfect state of harmony. Intelligent water drop (IWD) (Shah-Hosseini, 2009) is a natural inspired algorithm that simulates the physical processes occurring between water drops of a river and the soil of the river bed. There are also some other types of natural inspired algorithms that imitate different parallel evolutionary strategies in nature. The great salmon run (TCSR) (Mozaffari et al., 2012a) is a recent spotlighted metaheuristic inspired based on the annual migration of salmons in nature and different menaces laid behind their passage. The main concept behind the implementation of TCSR is to contrive an ensemble evolutionary strategy with a random population shuffling. Results prove that TCSR is highly capable to handle complex engineering problems (Mozaffari and Fathi, 2012).

The importance of developing and controlling thermal systems such as power plants that effectively use energy resources such as natural gas is apparent. One of the common tools in analysing and optimising the thermal systems like power plants derives from combining exergetic and economic properties of the flow stream in such systems. Exergetic and microeconomics forms the basis of thermoeconomics, which is almost known as exergoeconomics (Gorji-Bandpy and Mozaffari, 2012). Combining the second law of thermodynamic with economics (thermoeconomics) using availability of energy is one of the major objects that an engineer should apply for optimising thermodynamic systems. Its goal is to mathematically combine the second law of thermodynamic with the economic factors which predict the unit cost of product such as electricity and quantifies monetary loss due to irreversibility. Another crucial object in designing the operating parameters of a power plant is to achieve acceptable properties due to the first law of thermodynamic, e.g., obtaining maximum power, maximum efficiency and controlling the dependent parameters.
Bio-inspired algorithms have effectively come to the aid of thermodynamic scientists (Cui et al., 2012a, 2012b). Until now, several informatics and intelligent algorithms have been proposed to handle a variety of complex thermal systems. Mozaffari et al. (2012d) proposed a self-organise Pareto-based evolutionary strategy to optimise and control the operating parameters of Damavand power plant. Cammarata et al. (1998) formulate the objective function, the sum of capital investment and the operational and maintenance cost of a district heating network using exergoeconomic concepts. Esen et al. (2007) analysed the exergetic and energetic characteristics of a ground-coupled heat pump system with two horizontal ground heat exchangers using thermodynamical principles. Lee and Mohamed (2002) proposed a real-coded GA with a hybrid crossover operator for power plant control system design.

In this investigation, authors test the potential of metaheuristic algorithms for fast arrangement of thermoelectric cells in a condenser to maximise the generated power. Besides, they try to mitigate the maintenance cost of devised thermoelectric cells. In addition to these experiments, authors try to elaborate on the robustness of metaheuristics for optimal arrangement of thermoelectric cells. To achieve this aim, each metaheuristic conducts 30 independent optimisation procedures with random initial seeds.

The rest of paper is organised as follows. In Section 2, steps required for embedding the thermoelectric cells in the entry of condenser is scrutinised and the consequent mathematical formulations are given. Section 3 describes the details of the bio-inspired metaheuristics used in current research. In Section 4, the obtained results are discussed. Finally, Section 5 concludes the paper.

2 Problem definition

In our case study, thermoelectric cells are embedded through the entry of condenser to recover the heat loss in power plants. This is because the greatest energy loss occurs in the plants’ condensers, as the latent heat of condensation (Goudarzi et al., 2012). Thermoelectric power generation is adequate for utilising such low-grade heat and improving the total efficiency of power plants. Thermoelectric generation is a direct and clean heat-to-electricity conversion, and can be operated even if the temperature difference between the heat sources is small. Here, authors decide to utilise thermoelectric cells to retrieve the unused energy in condensers, as a thermal component, by means of thermoelectric conversion. Figure 1 shows framework of current design. As it can be seen, the considered heat and cool module is an aluminium block that perforated in its...
cross-section and it is installed in heat and cool situation of condenser’s input. The hot module entry of condenser is used so that the superheated vapour enters through one section and hot water exits from another side and moves through cool section. As we show in Figure 1, the cool water enters from the reverse side entry of hot module and exits from the reverse side output of hot module so for heat module, when superheated vapour passing through inside of heat aluminium module, a heat transformation between aluminium module occurs. This change of temperature is variable and proportional to the length of the hole. To find the variation of temperature through the aluminium module, Fluent and Gambit software were used.

Firstly, we meshed the aluminium module using Gambit software with 2,000 symmetrical mesh point in surface of module and inside of holes. Thereafter, we installed this formation to fluent software. To execute analysis, following boundary conditions are assumed:

1. upper and lower and lateral surfaces are insulated
2. the surface tangent of the models have convection heat transfer whit 25 air in free surfaces and have convection heat transfer with surface of modules.

For cool aluminium module we have water with Top Dubai and that is assumed having a constant temperature.

Until now, several experimental and mathematical efforts have been made to find a reliable correlation between the variation of inlet-outlet streams and thermoelectric power generation. Here, the authors use the experimental data reported by Thermonamic Electronics Corporation and adaptive neuro-fuzzy inference system (ANFIS) to anticipate the power generation (W) and maintenance cost (\(\text{TC} \)) outlet (\(T_C\)) temperatures.

The schematic of ANFIS architecture is indicated in Figure 2. It can be seen that our FIS has two inputs (x, y) and one output (f). The considered rule set with two if-then rules of TSK type of FIS can be mathematically expressed as below (Mozaffari et al., 2012b):

\[ f = \sum_{i=1}^{m} p_{ij} + \sum_{k=1}^{n} Q_k f_k \]

where \(\mu_j\) represents the \(j^{th}\) membership function for the input \(X_i\) and \(O_{ij}\) is the output of node \(ij\). In this paper, the Triangular and Gaussian membership functions are selected based on trial and error. The mathematical formulations of these functions are given by:

\[ \text{Triangular: } \mu(X) = \max \left( \min \left( \frac{X-a}{b-a}, \frac{c-X}{c-b} \right), 0 \right) \]

(2)

\[ \text{Gaussian: } \mu(X) = \exp \left( -\frac{(X-c)^2}{a^2} \right) \]

(3)

where \(a\) and \(c\) are locate in the feet of the triangle and the parameter \(b\) locates in the peak.

\[ O^i = \sum_{k=1}^{n} \hat{O}_k f_k \]

where \(\hat{O}_k = \hat{W}_k f_k\)

(6)

where \(\hat{W}_k\) is the output of de-fuzzification layer and \(f_k\) represents the output of \(k^{th}\) TSK-FIS rule. Here we extend the rule for \(m\) input TSK-FIS:

Rule If \((X_1)\) is \(A_{i_1}\) and \((X_2)\) is \(A_{i_2}\) and...and \((X_j)\) is \(A_{i_j}\) then:

\[ f_k = \sum_{i=1}^{m} p_{ij} + r_k \]

(7)

where \((p_{ij}, r_k)\) is the parameter set. The parameters in this layer will be referred to as consequent parameters.

Layer 5: the last layer is also known as output layer in ANFIS structure. It includes a single labelled encircled (\(\Sigma\)) with function of summation.

\[ O^\Sigma = \sum_{k=1}^{n} \hat{W}_k f_k \]

(8)

where \(Q\) represents the overall output of the network.
The hybrid algorithm has been allocated to adjust the parameters of both input and output membership functions. During the learning process the performance of ANFIS can be estimated by root mean square (RMS). RMS can be defined as below:

\[
RMS = \sqrt{\frac{1}{n} \sum_{m=1}^{n} (y_{p,m} - t_{m})^2}
\]  

(9)

where \( n \) represents the patterns number, \( y_{p,m} \) submit the predicted value, \( t_{m} \) is the measured value of one data point \( m \) and \( t_{m} \) is the mean value of all the measured targets.

Figures 3(a) and 3(b) show the data reported by thermonamic database.

Based on the feedback of designed network, the power generation and maintenance cost can be implied as follow:

\[
\hat{W}_{\text{Thermoelectric}} = ANFIS\_Power\_Pred(T_H, T_C)
\]  

(10)

\[
\hat{C}_{\text{Thermoelectric}} = ANFIS\_Cost\_Pred(T_H, T_C)
\]  

(11)

Figures 4(a) and 4(b) depicts the resulted variation of power generation and maintenance cost as a function of \( T_H \) and \( T_C \). It can be seen that ANFIS succeed to identify the behaviour of thermoelectric cells.

Based on the abovementioned equations, the objective functions can be formulated as follow:

\[
F_1 = \text{Max}\{P = n \times \hat{W}_{\text{Thermoelectric}}\}
\]  

(12)

\[
F_2 = \text{Min}\{\hat{C} = n \times \hat{C}_{\text{Thermoelectric}}\}
\]  

(13)

where \( v \) is the matched load power output of one thermoelectric cell, \( \hat{C}_{\text{Thermoelectric}} \) denotes the maintenance cost of one thermoelectric cell and \( n \) represents number of thermoelectric cells used in our case study. For executing the optimisation, these functions are unified and form of single objective functions. As it can be inferred, the main
goal of current study is to find an optimum number of thermoelectric cells to make a balance between the system maintenance cost and power output. The admissible ranges of operating parameters are considered as following:

\[ 8 \leq T_C \leq 13 \] (14)
\[ 96 \leq T_H \leq 197 \] (15)

In the next section, we explain the structure of bio-inspired algorithms selected for optimal arrangement of thermoelectric cells.

**Figure 4** The performance of ANN for predicting thermoelectric properties properties: power output and maintenance cost (see online version for colours)

3 Proposed bio-inspired metaheuristics

In this section, we yield a clear definition of bee algorithm (BA), particle swarm optimisation (PSO) and TGSR optimisation algorithms.

3.1 Bee algorithm

BA is a well-known metaheuristic algorithm inspired based on the foraging behaviour of honey bees in nature (Pham et al., 2004). This algorithm involves different types of control parameters. ‘\( n \)’ indicates the number of bees in the colony, ‘\( e \)’ determines the number of elite patches in each iteration. ‘\( \text{m} \)’ shows the number of patches with acceptable quality. ‘\( \text{nep} \)’ is the number of external recruited bees around the elite areas. These bees play an important role since they boost the ability of local search in BA. ‘\( \text{nsp} \)’ is a parameter used to calculate the number of external recruited bees in the algorithm. ‘\( \text{ngl} \)’ is a control parameter that evaluates the dominance of local search for recruited agents. At the beginning of the optimisation process, \( n \) scout bees are randomly initialised within the solution spans. In the next step, the fitnesses of each selected sights are evaluated. Bees with highest fitnesses are chosen as selected bees and their corresponding sights are selected for local neighbour exploration. Then, in next steps, BA performs local searches in the neighbourhood of the selected sights, assigning more bees to search near the best ‘\( e \)’ sights. After the local search process, for each patch only the bee with the highest fitness to form the next population. Finally, the remaining bees in the population are assigned randomly around the search space scouting for new potential solutions. In this state, required steps for one iterative exploration are fulfilled and BA goes through the next iteration. These steps are repeated until stopping criteria are met. The flowchart of BA is given in Figure 5.

**Figure 5** Flowchart of BA

Here, 50 bees have been hired to perform the optimisation procedure. We allocate 25 recruited bees to search nearby eight elite areas and also ten bees for exploiting around 20 acceptable sights.
3.2 Particle swarm optimisation

PSO algorithm which first developed by Kennedy and Eberhart (1995) is one of the most applicable swarm-based methods in optimisation and management of different complex engineering problems since it is simple in concept, easy to implement and computationally efficient. This method is inspired due to the social behaviour of birds flocking or fishes schooling where each of heuristic particles search the N dimensional solution space. At the step t, the i\textsuperscript{th} particle is defined as \( X(t) = (x_1(t), x_2(t), \ldots, x_N(t)) \) and the set of positions of m particles in a multidimensional space is identified as \( X = [X_1, \ldots, X_m] \). The fittest former position of the i\textsuperscript{th} particle is represented as \( P_i(t) = (p_{i1}, p_{i2}, \ldots, p_{in}) \). The best particle among all the particles in population is known as global model which is represent by the symbol g. The index of the best particle among all particles in a local searching zone is representing by the symbol l. At the time step t, the i\textsuperscript{th} particle velocity is defined as \( \dot{V}_i(t) = (v_{i1}(t), v_{i2}(t), \ldots, v_{in}(t)) \).

In classic PSO, the updating rule for global model’s velocity is mathematically expressed as:

\[
\dot{V}_i(t) = w \times V_i(t) - c_1 \times \text{rand} \times (P_{i\text{best}} - X_i(t)) \\
+ c_2 \times \text{rand} \times (P_{g\text{best}} - X_i(t)) \tag{16}
\]

Equation (16) is used to calculate a particle’s new velocity and distance of its current position from its own best experience (p\text{best}) and the group’s best experience (g\text{best}).

After updating the particle’s velocities, random walk is applied for updating particle’s position. This procedure is shown in equation (17):

\[
X_i(t) = X_i(t-1) + V_i(t) \tag{17}
\]

It should be noted that in all above equations, n represents the dimension \((1 \leq n \leq N)\), rand is a random number spanning to unity, w is the inertial weight and c\textsubscript{1} and c\textsubscript{2} are cognitive acceleration and social acceleration constants respectively.

The inertial weight is a linear decrementing function which mathematically defined as:

\[
w_t = w_{\text{max}} - \frac{w_{\text{max}} - w_{\text{min}}}{T} \times t \tag{18}
\]

Just like BA, in PSO, 50 heuristic agents are used to explore the searching space. \( w_{\text{max}} \) is equal to 1.47 and \( w_{\text{min}} \) decreases during the optimisation process. The flowchart of PSO is given in Figure 6.

3.3 The great salmon run

TGSR is a recent spotlighted metaheuristic in which the optimisation carry out through two independent operators (Mozaffari et al., 2012a). Each of these two operators deputes an independent salmon’s migration pathway. The first one belongs to the salmons that move through forested regions and mountain’s canyons. The other one belongs to the salmons which are passing through oceans, lakes and ponds. As it was mentioned in the former section, salmons choose their passage based on their instinct and without a meaningful inference. Each of these ways is incorporated with their own natural menaces. Commercial fishers concentrate on the ponds and ocean pathways while Grizzly bears hunt the salmons that pass through mountain’s canyons and forested regions. Each of these two main hunters utilises different techniques for hunting salmons with higher qualities. TGSR utilised all above steps to handle an optimisation problem. Along with its high ability for unimodal, multimodal and constraint numerical problems (Mozaffari et al., 2012a), it shows promising results in the case of real life engineering systems. Mozaffari and Fathi (2012) proved that TGSR successfully evolved the architecture of a monotonic ensemble artificial neural network (MEANN) for modelling the LSFF process (Mozaffari et al., 2012b). In the other work, they proposed a hybrid self-organised Pareto-based algorithm called (SOPEA) for handling the same problem. Consequent results indicate that while TGSR possesses a lower computational complexity compared to SOPEA, it can identify the LSFF system as well. This encourages the authors to engage the TGSR for a more demanding optimisation task i.e. optimal arrangement of thermoelectric cells. In the rest of this section, steps required for implementing the TGSR algorithm are given in details.

Figure 6 Flowchart of PSO

Initialisation: In TGSR algorithm, each potential solution delegates the salmon intensity in a region (amount of salmons in a sub group). In other words, region with higher salmon intensity yields a solution with higher fitness. The solutions are initialised stochastically spanning to the passage dominance (between lower bound and upper bounds). Equation (19) represents a procedure which is used to initialise random solutions with respect to the solution space.

\[
\text{Initial solution} = lb + \text{rand} \times (ub - lb) \tag{19}
\]

where \( lb \) and \( ub \) are the lower and upper bounds, respectively and rand is a random number spanning to 0 and 1 with a uniform distribution.
After initialising the solutions, optimisation procedure is started. At the beginning of the optimisation process, all of these initialised solutions (salmons sub groups) are prepared for their migration (iterative movements). It is obvious that each iteration cycle is equivalent to a natural migration phenomenon.

**Choosing pathways for migration:** Before migration, salmons choose their pathway based on their instinct. This suggests a stochastic shuffling control parameter for thrusting the salmon groups (initial solutions) in both pathways (evolutionary operators). Equation (20) formulates a mathematical form of this process.

\[
N_{p_i} = \left[ \mu \times P_s \right] \\
N_{p_i} = \left[ P_s - N_{p_i} \right]
\]

where \( N_{p_i} \) is the number of salmon groups passing through ocean and ponds, \( N_{p_i} \) is the number of salmon group passing through forested regions and mountain canyons, \( P_s \) is the number of all salmon groups which participate in the migration and \( \mu \) is a sharing factor that represents the salmon’s instinct. As seen, the proposed formulation is a strategy for shuffling the solutions stochastically. The results confirm the effectiveness of this shuffling in the diversification of the solutions.

After sharing process, these subgroups are entered in their pathways. They face different dangers while crossing these pathways. In the following, the details of the passage traversing are given.

**Crossing lakes and ponds:** In the first operator, the human hunting is simulated. Humans hire scout ships and commercial fishers to investigate the passage dominance (solution space).

The scout ships apply some arithmetic graphical search (an intelligent diversification methodology) to explore the passage as best as they can. This exploration has been mathematically modelled in equation (21).

\[
\begin{align*}
X_N &= X_F + \delta \left(t, \left( \alpha b - X_F \right) \right) \\
X_N &= X_F + \delta \left(t, X_F - \beta b \right)
\end{align*}
\]

where \( t \) represents the current iteration number, \( X_N \) represents a new detected region (new solution) and \( X_F \) shows the former region of the scout ship (former solution). \( \delta(x, y) \) is calculated using equation (22).

\[
\delta(x, y) = y \times \text{rand} \times \left(1 - \frac{x}{T}\right)^{\phi}
\]

where \( T \) is the number of the maximum iteration, \( b \) is a random number larger than 1 and \( \text{rand} \) is a random number spanning to 0 and 1 with a uniform distribution.

The remaining ships (known as commercial fishers) communicate with both scout ships and other active commercial fishers to find better areas (with higher salmon intensity) for hunting salmons. Then they congregate in the areas with higher intensity of salmons. Fisher groups often consist of two main hunter ships and one recruited ship. First, the main hunter groups find regions with an acceptable salmon intensity (solution fitness). After that, they inform the recruited agent to exploit nearby regions to find more intense areas (solution with higher fitness). This exploitation has been mathematically modelled in equation (23).

\[
X_R = \beta \times (X_{M1} - X_{M2}) + X_{M1}
\]

where \( \beta \) is a random number spanning to 0 and 1 with uniform distribution, \( X_R \) represents the new detected solution by the recruited agent, \( X_{M1} \) is the solution obtained by the first main hunter and \( X_{M2} \) is the solution obtained by the second one.

**Crossing mountain canyons and forested regions:** The second operator simulates the Grizzly bears hunting methodology. Similar to other animals, Grizzly bears communicate with each other to find a region with higher salmon intensity. Their hunting method is really simple. They always inform each other if they find an acceptable region. Then, the entire Grizzly bear groups approach the best region and search nearby areas. If they find an area with higher salmon intensity, they inform other bears. Otherwise, they leave the region and continue the local search. One of the main disadvantages of the bears hunting procedure is the lack of an independent diverse exploration. Bears hunting methodology is mathematically expressed in equation (24).

\[
X_B = \cos(\phi) \times (X_R - L_R) + L_R
\]

where \( X_B \) represents a new detected region, \( B_R \) is the best reported region by the hunting team, \( L_R \) is the current region in which the bears have decided to perform a local exploitation and \( \phi \) is an arbitrary angle spanning to 0 and 360 degrees. \( \cos(\phi) \) directs the bears to their destination. It is obvious that these animals perform an exploitation search with different radii and angle of attacks.

**Regrouping for spawning:** At the end of the migration, the survived salmons (current solution vectors) congregate in their destination for spawning. In TGSR, this natural event is simulated through a collection container. After salmons pass through their pathways (two evolutionary operators), the salmon subgroups (solutions) are collected in a unique container. In other words, the solutions are extracted from both operators and make a unique population. At this state, the algorithm has reached the end of the iteration.

The change in climate and urge for spawning are two main motivations which force the remaining salmons to begin another migration. Continuity of these permanent migrations turns the TGSR to a powerful iterative optimisation algorithm. The flowchart of TGSR algorithm is given in Figure 7.
Figure 7  Flowchart of TGSR

Set the controlling parameters: $\mu$, $b$, $prob$, population size ($P$), stopping criteria, number of variables, number of maximum iterations, solution space

Set TGSR iteration $= 0$; initialise the salmons subgroups (solutions) randomly in the solution space

Choosing pathways for migration based on $\mu$

Send the Grizzly bears for hunting salmons (this step improve the intensity); Compute the fitness of hunted salmons

Send all of the scout ships for scoping the solutions space (this step improves the diversification); Compute the fitness of hunted salmons

Recruit all of the commercial fis hers in the detected regions (apply local search); Compute the fitness of hunted salmons

Congregate the salmons for spawning

Stopping criterions are met?

Extract the global optimum solution

For current case study, 50 heuristic agents are used, sharing factor ($\mu$) is equal to 0.75 and $b$ is equal to 1.6.

4 Results and discussion

4.1 Comparing the performance of proposed algorithms

The controlling parameters of all optimisation algorithms are adjusted in accordance with the details in previous section and their original papers. To avoid a bias tuning, authors also exert several statistical tests to elaborate on the authenticity of these reported properties. All algorithms are executed in 30 independent runs with different initial random seeds on an ‘Intel Core 2 CPU 6600’ at 2.40 GHz and 2 GB RAM.

For executing the optimisation, these functions are unified and form a single objective function. Since both of the objective functions possessed an equal importance, authors use a same aggregation weight (0.5) for unifying objective functions. Besides, both of the objective functions are normalised in the span of unity for alleviating the probable bias optimisation that may occur for unbalance scale of objective functions. As a result, the optimisation yields an optimum solution that approximately address an
acceptable optimum solution for both power output and maintenance cost. The mathematical formulation of the proposed optimisation approach can be given as:

\[
\text{Objective Function } = -wF_1 + (1-w)F_2
\]  

(25)

For validating the effectiveness of the optimisation algorithm, we discuss the detailed computational data of these optimisation algorithms. The comparison metrics are measure of the quality (mean value), robustness (standard deviation), the success rate and the CPU time elapse. Mathematical definitions of these metrics are given as follows:

\[
\text{quality } = \frac{\sum_{i=1}^{n} \text{best sol}_i}{n}
\]  

(26)

\[
\text{robustness (Std.) } = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (\text{best sol}_i - \text{quality})^2}
\]  

(27)

\[
\text{CPU time } = \frac{\sum_{i=1}^{n} \tau_i}{n}
\]  

(28)

where \(n\) indicates the number of runtimes (30 in our case study), \(\tau\) deliberates the optimisation duration (seconds) for each independent run, \text{best sol} is a matrix that collects the global solutions obtained by optimisation algorithm and mathematically expressed as \([g_{\text{best1}}, \ldots, g_{\text{bestn}}]\).

Figure 8 depicts the general framework of current investigation. As it can be seen, metaheuristic optimisers act as supervising elements to find the optimum aggregation of power output and maintenance cost. At the very beginning of the process, a set of solutions are initialised in a random distributed style. After that, initial parameters enter in ANFIS phase to evaluate the corresponding power output and maintenance cost of embedded thermoelectric cells. The role of metaheuristics is to find the optimum operating parameters and its representative objective value.

Figure 9 depicts the mean performance (quality) of rival algorithms in 1,000 iterations through 30 independent runs. To avoid any probable discrimination and unwanted prejudice, all algorithms initialised identically.

As can be seen, at the end of optimisation, all of the algorithms find an approximately equal solution. The predominant difference may lay behind the convergence speed. As can be observed, BA consumes a higher number of iterative efforts to converge to global solution. Moreover, it seems that in this specific case, PSO shows a faster convergence speed when compared to TGSR.

Figure 10 depicts the robustness of these algorithms. It is clear that TGSR outperforms other methods in view of robustness since it has lower standard deviation (0.11) through 30 independent runs. In our case, it has been observed that BA could not show a robust behaviour, since the obtained standard deviation was greater than 1. On the other hand, PSO with standard deviation 0.53 showed acceptable robustness.

Another important metric in our case study is the CPU time elapse for efficient arrangement of thermoelectric cells. This metric aid the authors to judge about the speed of metaheuristics for optimal arrangement of thermoelectric modules. Figure 11 shows the bar diagram of CPU times for our case study.

---

**Figure 8** General framework of optimisation process
The last metric considered in our case study is the convergence of optimisation algorithms during the optimisation process. This metric indicates the ability of population-based optimisers to converge their heuristic agents in a proper region within the solution domain. To analyse the convergence rate of opponent methods, authors engage a same equation that was used before in one of their other researches. The mathematical formulation of convergence rate is given in below (Mozaffari et al., 2012c):

$$\text{Converge rate} = \frac{\text{best cost}}{\text{mean cost}}$$  \hspace{1cm} (29)

According to this equation, for a minimisation problem in which the objective function is transmitted within the unity, i.e., \((0, 1)\), it can be easily inferred that when the convergence rate is equal to 1, the maximum rate of convergence is obtained and when it is 0, heuristic agents diversely diffuse in solution space. Figure 12 is used to visualise this procedure and facilitate the understanding of this behaviour as well. As can be seen, in Figure 12(b), the solutions are dispersed within the solution space. The orange star represents the mean value of all heuristic agents and the black star delineates the best solution in current generation. As can be seen these position are a bit far from each other that means their values are not the same and consequently, the convergence rate is less than 1. On the other hand, Figure 12(a) indicates a condition where the solutions approximately converge in the vicinity of the best solution. Consequently, the mean value of solutions at this stage is relatively as same as the best obtain solution; hence, the convergence rate tends to reach its maximum value (i.e., 1).

Figure 13 illustrates the convergence rate of rival methods. For avoiding any discrimination, for all optimisation algorithms, the initial conditions are as same.

As can be seen, PSO show a higher convergence rate compared to other optimisation algorithms. At the very beginning of optimisation process, PSO’s heuristic agents are approximately diffused around the solution space. After some iteration, the heuristic agents tend to converge in an optimum region. In the case of BA algorithm, the convergence rate shows a smooth increment through the optimisation process. For TGRS algorithm, the scenario is slightly different. As it was mentioned in prior section, TGRS is a double population-based optimisation algorithm. This evokes an ensemble optimisation approach with a random population shuffling mechanism. The predominant feature behind the robustness of TGRS turns to random diffusion of heuristic agents through optimisation phases. As a result, one can be observed that the convergence of solution does not rise significantly during the optimisation process. However, it should be noted that TGRS is as effective as other optimisation approaches and the major difference returns to different exploration strategies.
In next section, along with above experiments and results, authors intend to prove the reliability and effectiveness of their method by comparing the gained results to an empirical effort for same problem. All the below experimental efforts have been done at Thermo Electric Lab., Babol University of Technology.

4.2 Experimental efforts for finding the optimum number of thermoelectric cells

Figure 14 depicts the overall schematic illustration of the experimental framework including the modules position and the mounted measurement tools.
Bio-inspired methods for fast and robust arrangement of thermoelectric modulus

The design aluminium modules have been built in original distances \((4,000 \times 1,500 \text{ mm}^2)\) and experienced in laboratory with original conditions. The temperatures and pressures were measured by proper gages. Along with real time gages, some other gages were setup at both inlet and outlet positions of the module. Then authors have installed 20 thermoelectric modules for one transverse arrangement and filled residue space by non-conducting heat materials (preventive of heat waste). At the next step, the generative power was measured using a common oscilloscope. Then second transverse arrangement with approximate 7 mm distance between each thermoelectric cells and approximate 14 mm distance between the extreme cell and the edge of each aluminium modules was installed and residue space was filled by non-conducting heat materials measure generative power. Same procedure was repeated for about 50 rows and the generative power was measured in each stage. Also we have measured temperatures at each row by temperature gages. At the end of, authors recorded results and calculate efficiently and cost of generative power in each step. As a logical tentative optimisation effort, we have chosen seven points and have shown temperatures of both hot and cool module and their resulted generative power and maintenance cost for finding optimal situations. The obtained experimental details are tabulated in Table 1.

Table 1  Experiment optimisation results

<table>
<thead>
<tr>
<th>Experiment no.</th>
<th>TL</th>
<th>TH</th>
<th>Power output (W)</th>
<th>Maintenance cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. (1 row)</td>
<td>12.60</td>
<td>197</td>
<td>354</td>
<td>1,760</td>
</tr>
<tr>
<td>2. (8 rows)</td>
<td>11.20</td>
<td>186</td>
<td>53,280</td>
<td>24,720</td>
</tr>
<tr>
<td>3. (15 rows)</td>
<td>10.40</td>
<td>158</td>
<td>75,000</td>
<td>30,200</td>
</tr>
<tr>
<td>4. (20 rows)</td>
<td>9.8</td>
<td>143</td>
<td>101,200</td>
<td>36,000</td>
</tr>
<tr>
<td>5. (29 rows)</td>
<td>9.4</td>
<td>131</td>
<td>127,600</td>
<td>43,270</td>
</tr>
<tr>
<td>6. (36 rows)</td>
<td>8.8</td>
<td>118</td>
<td>134,200</td>
<td>48,890</td>
</tr>
<tr>
<td>7. (45 rows)</td>
<td>8.2</td>
<td>96</td>
<td>142,300</td>
<td>52,900</td>
</tr>
</tbody>
</table>

Figure 15 depicts the estimated surfaces for both power output and maintenance cost. The plates were fitted using MATLAB software.

According to non-domination concept, it can be observed that most of the points do not dominate each other with respect to desire power output and maintenance cost. This is because the higher the power output is, the larger the maintenance cost become. However, by observing and expounding the surfaces and engaging a deliberate physical analysis, authors concluded that the modules characteristics and outputs at sixth experimental procedure (i.e., experiment with 36 rows) are logical. The characteristics gained through this experiment are shown in bold format in Table 1.

4.3 Comparing the physical properties of obtained solutions

Thermal properties of the power plant under the optimum parameters are tabulated in Table 2. For assurance on the effectiveness of the solutions obtained by bio-inspired computing, authors compare the results with those derived through experimental procedure expressed in previous section.

As can be seen, bio-inspired methods are much faster when compared to experimental optimisation procedure. Besides, it should be noted that, contrary to experimental procedure, the knowledge provided in ANFIS phase relax the expert user from any tedious physical analysis. Moreover, in previous sections, we proved that all three optimisation algorithms showed a high robustness (reliability) through 30 different optimisation processes. By taking a peek at the table, one can observe that the condition suggested by bio-inspired algorithms is much more desirable compared to the experimental solution. As can be seen, without a large increase in the maintenance cost, authors found a condition that significantly yields a higher power output.
The optimum solution suggests the use of 36 thermoelectric cells through the length of aluminium modules. It is observed that, these numbers of thermoelectric cells are deliberate since their resulted electricity power easily compensates their consequent maintenance cost. Figure 16 indicates the optimum arrangement of thermoelectric cells within the aluminium modulus.

**Figure 15** Obtained surfaces for power output and maintenance cost (see online version for colours)

![Obtained Experimental Surface](image)

---

**Table 2** Power plant’s properties with base and proposed structure

<table>
<thead>
<tr>
<th>Methodologies</th>
<th>Decision variables</th>
<th>Objectives</th>
<th>Procedure time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TL</td>
<td>TH</td>
<td>Power output</td>
</tr>
<tr>
<td>Experimental analyse</td>
<td>8.8</td>
<td>118</td>
<td>134200</td>
</tr>
<tr>
<td>BA+ANFIS</td>
<td>8.2</td>
<td>111.48</td>
<td>138700</td>
</tr>
<tr>
<td>TGSR+ANFIS</td>
<td>8.2</td>
<td>111.50</td>
<td>138600</td>
</tr>
<tr>
<td>PSO+ANFIS</td>
<td>8.2</td>
<td>111.33</td>
<td>138600</td>
</tr>
</tbody>
</table>
As it was mentioned, the number of thermoelectric cells is proportional to both length and width of aluminium module. In current design, we have used 7 rows of thermoelectric modules for transverse arrangement. As mentioned before, we considered 7 mm distance between each thermoelectric cells and about 14 mm distance between the extreme cell and the edge of each aluminium module. As it can be seen, the optimal arrangement dictates the use of 54 rows. So the dimension of aluminium module should be $3500 \times 5100$ mm$^2$. This amount a smaller module when compared to the module considered in experimental procedure. The temperature of output steam from the heat aluminium module is 111°C. This water enters to the condenser and mixes with output water from the cool aluminium module. This resulted to a heat recovery in condenser and the energy system.

5 Conclusions

In this study, authors test the potential of bio-inspired algorithms (as stochastic optimisers) for fast and reliable arrangement of thermoelectric modules. According to authors’ knowledge, such effort was not done before in literature and therefore the application of bio-inspired computing in this field is novel and welcome. For validating and elaborating on the appropriate performance of the proposed methods, authors also develop a precise experimental frame work at Thermo Lab. BUT. The results show that not only the algorithms are capable to arrange the cells in a reliable fashion, but also they consume a less computational time for the process. In the next work, authors would like to test the concept of multi-objective approaches for same optimisation process.

References


A. Mozaffari et al.


