



An oppositional Salp Swarm: Jaya algorithm for thermal design optimization of an Organic Rankine Cycle

Mert Sinan Turgut¹ · Oguz Emrah Turgut²

Received: 4 September 2020 / Accepted: 21 December 2020 / Published online: 27 January 2021
 © The Author(s) 2021

Abstract

This study proposes a hybrid metaheuristic algorithm to tackle both single and multi objective optimization problems that are subjected to hard constraints. Twenty-four single objective optimization benchmark problems comprising uni-modal and multi modal test functions have been solved by the proposed hybrid algorithm (OPSSAJ) and numerical results have been compared with those acquired by some of the new emerged metaheuristic optimizers. The proposed OPSSAJ shows a significant accuracy and robustness in most of the cases and proves its efficiency in solving high dimensional problems. As a real-world case study, seventeen operational design parameters of an organic rankine cycle (ORC) operating with a binary mixture of R227EA and R600 refrigerants are optimized by the proposed hybrid OPSSAJ to obtain the optimum values of contradicting dual objectives of second law efficiency and Specific Investment Cost. A Pareto curve composed of non-dominated solutions is constructed through the weighted sum method and the final solution is chosen by the reputed TOPSIS decision-maker. The pareto curve and best-compromising result obtained by utilizing the OPSSAJ are compared with that of acquired by using nondominated sorting genetic algorithm II (NSGA-II) and multiple objective particle swarm optimization (MOPSO) algorithms. The multi-objective ORC design obtained with the OPSSAJ yields a significant improvement in thermal efficiency and cost values compared to designs found by the NSGA-II and MOPSO algorithms. Furthermore, a sensitivity analysis is performed to observe the influences of the selected design variables on problem objectives.

Keywords Jaya algorithm · Metaheuristic optimization · Organic Rankine Cycle · Salp Swarm algorithm · Zeotropic mixtures

List of symbols

Symbols

A	Total heat exchange surface (m^2)	F	Position of the food source
$Best$	Position of the best salp	h	Convective heat transfer coefficient (W/m^2K)
C	Cost (\$)	I	Irreversibility (W)
C	Randomly-generated number	$iter$	Current iteration
$CEPCI$	CEPCI coefficient	k	Thermal conductivity (W/m^2K)
D	Dimension size	lb	Lower bound
d	Tube diameter (m)	\dot{m}	Mass flow rate (kg/s)
		M	Center value, number of test functions considered

Supplementary Information The online version contains supplementary material available at (<https://doi.org/10.1007/s42452-020-04014-0>).

✉ Mert Sinan Turgut, sinanturgut@me.com | ¹Mechanical Engineering Department, Faculty of Engineering, Ege University, 35040 Bornova, Izmir, Turkey. ²Industrial Engineering Department, Faculty of Engineering and Architecture, Izmir Bakırçay University, 35665 Menemen, Izmir, Turkey.



SN Applied Sciences (2021) 3:224 | <https://doi.org/10.1007/s42452-020-04014-0>

<i>MAX</i>	Maximum value at each iteration
<i>mean</i>	Mean value:
<i>MIN</i>	Minimum value at each iteration
<i>maxIter</i>	Maximum number of iterations
<i>N</i>	Population size
\dot{Q}	Heat flux (W)
<i>p</i>	Pressure (Pa)
<i>R</i>	Thermal resistance (W/K)
<i>r</i>	Randomly-generated number
<i>S</i>	Matrix consists of the randomly-generated solutions, Entropy (J/K)
<i>t</i>	Time (s)
<i>T</i>	Temperature (K)
<i>U</i>	Overall heat transfer coefficient (W/m ² K)
<i>ub</i>	Upper bound
<i>v</i>	Velocity (m/s)
<i>w</i>	Position of the worst salp
\dot{W}	Rate of work transfer (W)
<i>X</i>	Position of the swarm member
η	Efficiency

Sub- and superscripts

<i>0</i>	Initial, ambient
<i>Best</i>	Best value
<i>cap</i>	Capital
<i>cond</i>	Condenser
<i>diff</i>	Difference
<i>evap</i>	Evaporator
<i>exp</i>	Expansion device
<i>fan</i>	Fan
<i>gen</i>	Generated
HEX	Heat exchanger
<i>in</i>	Inside
<i>max</i>	Maximum value
<i>min</i>	Minimum value
<i>out</i>	Outside
<i>pump</i>	Pump
<i>tot</i>	Total
<i>wf</i>	Working fluid
<i>worst</i>	Worst value

1 Introduction

Optimization involves successive iterative processes aiming to obtain the best decision variables among the set of candidate solutions for a particular problem. The quality of the solution is evaluated by the optimization objective function, which is to be maximized or minimized depending on the optimization purposes. The iterative search process is subjected to a predefined number of problem-specific restrictions called constraints, which can be mathematically formulated in terms of equality or inequality

expressions. There are several options for solving a typical optimization problem. Deterministic and stochastic optimization strategies are the most prominent methods between the available alternatives. Deterministic methods provide reliable tools for finding the global optimum point of the problem also claiming that the obtained solution is the global one. They strongly rely on the analytic information of the problem to probe the search space efficiently. The optimum solution is converged within a lower number of function evaluations in comparison with the stochastic solvers. These types of solvers are successful in solving the unimodal problems having linear search spaces. However, they experience difficulties in solving non-convex problems as local optimum entrapment is highly possible due to the singularities and non-differentiable points on the function space. One can conquer these characteristic drawbacks by modifying or hybridizing the algorithm [1].

Efficient design of an ORC becomes a hot spot research area for designers as well as researchers for many years. Many research studies have been devoted for thermodynamical design and optimization of ORC systems [2–4]. Some other completed studies cover experimental and theoretical applications of ORC along with its viable utilization in different type of hybrid power generation systems [5–7]. Besides, more and more studies about the ORC systems are finding their places in the literature every day [8–10]. Haghparast et al. [2] have investigated the impact of ejector geometry and power output capacity on cycle performance. A sensitivity study has been carried out by the researchers to find out the most significant parameters that have impact on cycle efficiency. The results of the parametric studies that have been conducted by the authors showed that the net power output increases as the ejector area ratio or secondary fluid mass flow rate rises; and decreases as the throat diameter, primary inlet pressure or primary inlet temperature increases. Furthermore, it has been concluded that area duct diameter is one of the most important geometrical parameter that effects the cycle performance. Chen et al. [3] have proposed a novel zeotropic ORC that the mixture composition can be adjusted based on the environmental temperature variation utilizing liquid-separation condensation. The optimization model has been developed to optimize the mixture composition with respect to the changing environmental temperature. A parametric optimization study has been performed and the results showed that the proposed ORC system significantly improved the output power compared to the conventional ORC in terms of 100 °C heat source. Feng et al. [8] have investigated and optimized a waste-heat regenerative ORC by utilizing classical and finite-time thermodynamics. The authors have accomplished to maximize the power output and thermal efficiency of the cycle by fixing total heat transfer area of all heat exchangers and

changing the superheat, mass flow rate of the working fluid and heat transfer area ratios of the heat exchangers. As a result of this study, the authors have found out that there exists an optimal ORC design with optimal superheat, working fluid mass flow rate and heat transfer area ratio values. Moreover, the power output and thermal efficiency of the cycle are increased by 25.22% and 12.16% after the optimization process finished. Baldasso et al. [9] have proposed a model to optimize the design of an ORC for waste heat recovery in maritime applications. The findings of the authors showed that the combination of minimum exhaust gas velocity and maximum engine backpressure acts as a constraint on the maximum amount of heat that can be drawn out from the waste heat recovery boiler. The authors have considered ten design variables and one objective function for the development of the optimization problem. The results have shown that the overall system fuel consumption can be lowered by 0.52–1.45 g/kWh by allowing higher backpressure levels on the engine. Liu et al. [10] have analyzed the performances of single—and dual-pressure ORCs with combined heat and compressed air energy storage modifications. The authors have developed thermodynamic models of the mentioned cycles and compared the performances of the cycles with each other. As a result of this study, the authors have come to the following conclusions. The dual-pressure ORC has produced more power compared to the single-pressure ORC. The optimum superheat temperature of the dual-pressure ORC with combined heat and compressed air energy storage modifications is much smaller compared to its single-pressure counterpart.

In this study, Salp Swarm Algorithm (SSA) [11] is hybridized with the Jaya algorithm (JAYA) [12] and a variant of opposition-based learning (OBL) [13] to improve the solution diversity and convergence rate. SSA is a swarm-based metaheuristic simulating the intelligent foraging and navigating behaviors of a salp chain in the ocean. Some of the recent applications of SSA can be exemplified as parameter extraction of PEM fuel cells [14], feature selection [15], and optimum allocation of power sources [16]. JAYA is a population-based algorithm with a very simple search mechanism and successfully applied to engineering design problems from various disciplines [17–19]. The optimization performance of the proposed hybrid is enhanced by utilizing a variant of OBL called quasi-opposition based learning (QOBL). OBL takes simultaneously advantage of a current estimate solution and its opposite point to improve the search capacity of the included algorithm. OBL concept has been incorporated into many metaheuristics such as Particle Swarm Optimization [20], Harmony Search [21] and Whale Optimization Algorithm [22]. SSA, JAYA and OBL methods have been coupled with each other in the literature before to solve

various optimization problems [23, 24], however, to the best knowledge of the authors, these three techniques have never been incorporated before. Besides the above mentioned successful applications of these algorithms on various optimization problems, they lack some algorithm-specific deficiencies similar to the other metaheuristics. For instance, SSA suffers from the slow convergence and local minima entrapment resulting from the redundant intensification of the fertile areas on the search space, which is occurred by the salp chain movement of the followers. Apart from the favorable advantages of being a parameter-free algorithm with an accelerated convergence rate, JAYA still has some tedious disadvantages such that the algorithm has only one manipulation scheme that may lead to obtaining candidate solutions with lower diversity. JAYA performs such iterative calculations that perturbed population individuals get closer to the best solutions while moving away from the worst ones. This tendency may occur a premature convergence due to the extensive exploitation of the potential best solutions. The hybridization of metaheuristic algorithms with OBL greatly increases the solution accuracy along with convergence speed. Thanks to the enhanced diversification produced by using the mirror point rather than the randomly generated point, OBL has the capability of reaching unvisited regions of the solution space [25]. Literature applications of QOBL [26, 27] reveal that using quasi-opposition instead of an opposition point enables us to acquire sample solutions much closer to the global optimum. Another strong point of QOBL has iteratively updated search regions that are dynamically shrunk throughout iterations. Updated extremum values of the search space pave the way for capitalizing on the historical knowledge of the previous generations and increase the possibility of finding better solution outcomes. This study hybridizes these three algorithms to eliminate their inherent structural drawbacks and benefit from their intrinsic advantages on the optimization framework called OPSSAJ. Optimization efficiency of the proposed OPSSAJ is firstly assessed on twenty-four benchmark problems composed of unimodal and multimodal test functions. Numerical results are compared and evaluated against those obtained from some recently emerged renowned metaheuristics. Comparative results show the superiority of the proposed algorithm over its constituent algorithms along with the challenging contenders. Then, OPSSAJ is applied to a highly complex real-world optimization problem of the multi-objective design of an organic ranking cycle (ORC). Finally, a sensitivity analysis is performed to observe the effects of seventeen design variables on the problem objectives.

This research study aims to provide an essential solution framework for optimum thermal design optimization for an Organic Rankine Cycle (ORC) working with a binary

mixture. This study also proposes a brand new variant of SSA in which basic mutation scheme of JAYA and rudimentary manipulation equations of another variant OBL scheme called QOBL are successfully integrated. This kind of integration entails a plausible balance in the exploration and exploitation phases of the proposed hybrid method. Then, this proposed hybrid method is applied to maintain a design optimization of an ORC. The foremost novelty occurring in this paper is that this is the first research study deeply investigating the thermal behavior of an ORC system taken into account of two conflicting but complementary designs of objectives of Specific Investment Cost and second law efficiency of the cycle. Several literature studies [28–30] have been accomplished up to now dealing with thermo-economic design optimization of ORCs running with binary mixtures. However, most of them consider a limited number of cycle parameters to be optimized and remaining parameters are taken constant during the iterations. This study offers a comprehensive examination of the influences of cycle parameters on considered conflicting design objectives. Moreover, limited research papers attempted to perform a comparative study between ORCs working with pure fluids and binary mixtures, which is detailly investigated in this work. Another significant contribution of this study is that this is the first consideration of the refrigerant mass flow rate and mass fraction of refrigerants in the binary mixture as design variables. Pareto curves for different cycle configurations are constructed and final optimum solution among the non-dominated solutions is chosen by the highly-reputed TOPSIS decision-maker [31]. Another novelty of this paper is that the improved solution accuracy and robustness which is resulted from the meticulously created synergy between JAYA and QOBL method. Proposed hybridization procedure leads to a significant boost in the probing mechanism of ordinary SSA by enhancing the exploration and exploitation capacities of the algorithm.

To achieve the main goals of the research study, the following contributions are proposed,

- Improved version of the SSA called OPSSAJ is proposed to solve complex optimization problems in this research study.
- Hybridization of the JAYA with the QOBL is adapted into the leader phase of the SSA to diversify the swarm individuals, which has never been studied before in such utilization so far.
- The proposed method (OPSSAJ) is then applied to obtain the optimal design of an Organic Rankine Cycle working operating with a binary mixture of R227EA and R600. This is the first application of a metaheuristic algorithm on multi-objective optimization of a thermal cycle running with a binary mixture.

- Influence of optimum design parameters of the ORC cycle on conflicting but complementary problem objectives has been comprehensively analyzed, much deeper than most of the literature studies dealing with an ORC design.

The rest of the paper is organized as follows: Sect. 2 provides brief instruction on Salp Swarm Algorithm, Opposition-based Learning, and Jaya Algorithm. The essentials of the proposed hybrid are expressed in Sect. 3. Numerical results of the hybrid method along with the related discussions are provided in Sect. 4. Section 5 reports the single and multi-objective optimization results of ORC. Section 6 concludes this research study with remarkable comments.

2 Related work

2.1 Salp Swarm algorithm

Salp Swarm Algorithm (SSA) is a new bio-inspired metaheuristic algorithm developed by Mirjalili et al. [11]. SSA mimics the swarming behavior of salps during navigating in the deep sea. The movement of salps is modeled as salp chains that compose of two groups. The first group consists of the leader, and the second group is formed by the followers. The leader is located at the front of the chain and directs the whole chain. On the other hand, the followers go after the leader and at the same time they follow each other. The updated position of the leader in the n -dimensional search space is given as:

$$x_{1,j} = \begin{cases} F_j + c_1((ub_j - lb_j)c_2 + lb_j), & c_3 \geq 0.5 \\ F_j - c_1((ub_j - lb_j)c_2 + lb_j), & c_3 < 0.5 \end{cases} \quad (1)$$

where $x_{1,j}$ represents the j -th position of the first salp (leader), F_j is the position of the food source, and lb_j and ub_j denote lower and upper bounds of the j -th dimension, respectively.

The coefficients c_1, c_2, c_3 are random numbers. The coefficient c_1 is used to balance exploration and exploitation and defined as:

$$c_1 = 2e^{-\left(\frac{4 \times iter}{maxIter}\right)^2} \quad (2)$$

where $iter$ and $maxIter$ represent current iteration and the maximum number of iterations, respectively. The coefficients c_2 and c_3 are random numbers lying in the range $[0,1]$.

The position update rule of the followers obeys Newton's law of motion as:

$$x_{i,j} = \frac{1}{2}at^2 + v_0t \quad (3)$$

where $i \geq 2$ and $x_{i,j}$ shows the position of i -th follower salp in the j -th dimension, t denotes the time, v_0 is the initial speed, $a = v_{final}/v_0$ and $v = (x - x_0)/t$. Considering the discrepancy between iterations as equal to one and $v_0 = 0$, position update equation of the follower salps can be written as:

$$x_{i,j} = \frac{1}{2}(x_{i,j} + x_{i-1,j}) \quad (4)$$

where $i \geq 2$ and $x_{i,j}$ shows the position of i -th follower salp in the j -th dimension. The pseudocode of the SSA is given in Algorithm 1 [11].

Algorithm 1 Pseudocode of the SSA algorithm

initialize the population randomly

calculate the fitness of each agent

determine the best solution

while *termination criterion is not met do*

for *each salp* X_i **do**

if ($i \leq N/2$) **then**

update positions according to Eq. 1

else

update positions according to Eq. 4

end if

end for

check boundaries

update best solutions

end while

return *best solution and best fitness*

Extending this definition to higher dimensions, the opposite point can be defined as the following.

Definition (Opposite point) Let $P(x_1, x_2, \dots, x_n)$ be a point in the n -dimensional space, where $x_1, x_2, \dots, x_n \in R$ and $x_i \in [a_i, b_i] \forall i \in \{1, 2, \dots, n\}$. The opposite point is defined by its components as

$$\tilde{x}_i = a_i + b_i - x_i \quad (6)$$

where \tilde{x}_i represents opposite-point.

Based on the opposite point definition, quasi-opposite point is defined as a uniform random point generated between the center point and the opposite point. In the

2.2 Opposition based learning

There are seven widely used OBL approaches in the literature [32, 33]. These are, basic OBL [13], quasi-opposition [34], quasi-reflection [35], current optimum opposition [36], extended opposition [33], reflected extended opposition [33], and comprehensive opposition [33]. Among these seven variants of OBL, quasi OBL approach has enjoyed a visible position in the literature. It has been mathematically proved that the probability of the quasi-opposite point being closer to the unknown optimal solution is higher than being closer to the basic opposite point [34, 37]. Therefore, the present study employs quasi-OBL approach. Before giving details, basic definitions are introduced as follows.

Definition (Opposite Number) Let $x \in [a, b]$ be a real number. The opposite number of \tilde{x} is defined by

$$\tilde{x} = a + b - x \quad (5)$$

quasi-OBL, the opposite of each variable and center point is updated in each iteration dynamically. Because the search space is shrunk over the course of iterations, going through the initial static boundaries of variables might result in boundary violations and poor search capability. Therefore, the minimum and maximum values of each variable in the current iteration ($[MIN_j^t, MAX_j^t]$) are used to calculate center point M_j^t , which denotes the mean value of minimum and maximum values, respectively. By updating the boundary intervals in each step, the knowledge of previous generations is implicitly capitalized. The calculation of the quasi-opposite population is shown in Algorithm 2 [13].

Algorithm 2. Pseudo-code for quasi-OBL

```

1  for i = 1 to N do
2    for j = 1 to D do
3      OPXi,j = Mini,jt + Maxi,jt - xi,j
4      Mi,jt = (Mini,jt + Maxi,jt) / 2
5      if xi,j < Mi,jt then
6        QOPXi,j = Mi,jt + (OPXi,j - Mi,jt) × rand(0,1)
       else
7        QOPXi,j = OPXi,j + (Mi,jt - OPXi,j) × rand(0,1)
8      end if
9    end for
10 end for
    
```

2.3 Jaya algorithm

Jaya algorithm is a recent population-based metaheuristic algorithm developed by Rao [12]. The main rationale underlying the Jaya algorithm is that individuals try to escape from the non-promising solutions and tend towards the best solution. Java algorithm has a very simple search mechanism that is computationally efficient and straightforward. Jaya algorithm does not require any parameters pertaining to its search mechanisms. It only requires the knowledge of population size and number of generations. Main steps of the Jaya algorithm can be given as follows:

Step 1 Population initialization: Initial positions of individuals are randomly generated within the range of boundaries.

$$x_{i,j} = x_{\min,j} + rand(0, 1)(x_{\max,j} - x_{\min,j}) \tag{7}$$

where $x_{i,j}$ represents j th position value of the i th individual in the population, $rand(0, 1)$ denotes uniformly distributed random number lying between 0 and 1, $x_{\min,j}$ and $x_{\max,j}$ are the lower and upper bounds of the j th dimension, respectively. Once the initial population is formed, the fitness value of each solution vector is calculated. Then, the best and worst individuals are identified. Afterwards, each individual is subjected to repeated cycles of the search processes, which is described below, until the predefined termination criterion is met.

Step 2 Generating a new solution. The new solution vector is calculated as follows:

$$x'_{i,j} = x_{i,j} + r_{1,j}(x_{best,j} - |x_{i,j}|) - r_{2,j}(x_{worst,j} - |x_{i,j}|) \tag{8}$$

where $r_{1,j}$ and $r_{2,j}$ represent the j th dimension of the uniform random numbers, $x_{best,j}$ and $x_{worst,j}$ are the j th dimension of the best and worst solutions, respectively.

In Eq. 2, the term $(x_{best,j} - |x_{i,j}|)$ serves as guiding the solution towards the best solution and $(x_{worst,j} - |x_{i,j}|)$ is used to ensure that the new solution gets away from the worst solution.

Step 3 Checking for the boundaries: In this step, each dimension of the solution vector is checked against the allowable regions.

$$x'_{i,j} = \begin{cases} x'_{\min,j} & \text{if } x'_{i,j} < x'_{\min,j} \\ x'_{\max,j} & \text{if } x'_{i,j} > x'_{\max,j} \\ x'_{i,j} & \text{otherwise} \end{cases}, \forall i \in \{1, \dots, N\}, j \in \{1, \dots, D\} \tag{9}$$

Step 4 Giving solution acceptance decision: In this step, the fitness value of a new solution is compared with that of the current solution. If the new solution is better than the current solution, then the new solution is accepted, and the current solution is replaced by the new solution. Otherwise, the current solution remains in the population and the new solution is discarded.

$$X'_i = \begin{cases} X'_i, & \text{if } fitness(X'_i) \leq fitness(X_i) \\ X_i, & \text{otherwise} \end{cases} \tag{10}$$

where X_i and X'_i represent the current and new solutions, respectively.

3 Proposed algorithm

This section gives the basics of the opposition-based Salp Swarm Algorithm with the Jaya (OPSSAJ). In particular, the OPSSAJ method hybridizes quasi-opposition-based learning with the SSA and Jaya algorithms. There are three main steps in the proposed algorithm, which are the generation of the initial population, the movement of the salp chain, and updating the best solutions.

The procedural steps of the OPSSAJ method can be given as follows:

3.1 Generating initial population

The proposed algorithm starts with generating an initial population randomly. The initial population of salps is formed based on the predefined bounds, which are represented as $x_{\min,j}$ and $x_{\max,j}$ respectively. Mathematically speaking, the i th salp of the population $\vec{X}_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,D}\}$ is generated based on Eq. 11.

$$x_{ij} = x_{\min,j} + \text{rand}(0, 1)(x_{\max,j} - x_{\min,j}) \tag{11}$$

where x_{ij} denotes the value at the j th position of the i th salp.

After generating the initial population of salps, the neighboring solutions are identified within the search space.

3.2 Movement of the salp chain

In the canonical SSA, the first salp is regarded as the leader salp, and the rest of the salps follow the leader. This mechanism has been improved by hybridizing the SSA with the oppositional based learning and Jaya algorithms. The use of multiple leaders avoids trapping into local optima and improves the exploration capability of the algorithm.

In order to balance exploration and exploitation during the search, the leader salps randomly select either the oppositional-based learning or Jaya algorithm. In this study, quasi-opposition based learning has been employed, since the probability of the quasi-oppositional point being closer to the unknown optimal solution is higher than being closer to the opposite point [37]. To perform quasi-opposition-based learning, minimum, and the maximum values of each dimension are found based on the following equations:

$$\min_j = \min_{i \in \{1,2,\dots,N/2\}} \{x_{ij}\} \tag{12}$$

$$\max_j = \max_{i \in \{1,2,\dots,N/2\}} \{x_{ij}\} \tag{13}$$

The oppositional position vector is calculated as:

$$x_{ij}^o = \min_j + \max_j - x_{ij} \tag{14}$$

where x_{ij}^o represents the oppositional point of the j th position of the i th salp. Once the minimum and maximum values are identified for each dimension, medium value is calculated as:

$$\text{med}_j = (\min_j + \max_j) / 2 \tag{15}$$

where med_j denotes the mean value of the j th dimension.

Eventually, the quasi-oppositional position is calculated as follows:

$$x_{ij}^{qo} = \begin{cases} \text{med}_j + (x_{ij}^o - \text{med}_j) \times \text{rand}(0, 1), & \text{if } x_{ij} < \text{med}_j \\ x_{ij}^o + (\text{med}_j - x_{ij}^o) \times \text{rand}(0, 1), & \text{otherwise} \end{cases} \tag{16}$$

where x_{ij}^{qo} is the value of the j th dimension of the i th quasi-oppositional salp. On the other hand, if the leader salps perform movements based on Jaya algorithm, the indices of the best and worst solutions are calculated as:

$$\text{best} = \arg \min_{i \in \{1,2,\dots,N/2\}} \{f(\vec{X}_i)\} \tag{17}$$

$$\text{worst} = \arg \max_{i \in \{1,2,\dots,N/2\}} \{f(\vec{X}_i)\} \tag{18}$$

where best and worst represent the indices of the best and worst salps, respectively. Let x_{best} and x_{worst} denote the best and worst solution vectors among the first $N/2$ individuals in the population, respectively. The movement of leader salps, according to Jaya algorithm, can be given as:

$$x_{ij}^{\text{jaya}} = x_{ij} + r_{1j}(x_{\text{best},j} - |x_{ij}|) - r_{2j}(x_{\text{worst},j} - |x_{ij}|) \tag{19}$$

where x_{ij}^{jaya} is the value of the j th position of the i th salp as a result of Jaya move. The new position is accepted if the fitness of the new salp is better than the previous salp. Otherwise, the current salp is not replaced with the new salp. On the other hand, the follower salps move their positions following the salp chain movement, which is defined as:

$$x_{ij} = \frac{1}{2}(x_{ij} + x_{i-1,j}) \tag{20}$$

where i denotes the index of the followers and $i \geq (N/2) + 1$. Once the movements of the salp chains are completed, the fitness value of each salp is calculated. Then, the population is sorted based on the fitness values, and the leader and follower salps are identified. The population of salps is subjected to repeated cycles of the search processes until a termination criterion is met. The pseudo-code of the proposed algorithm is given in Algorithm 3.

Algorithm 3. The proposed OPSSAJ algorithm

1. *Determine parameters*
 2. *Initialize the population randomly*
 3. *Calculate the fitness of each agent*
 4. *Sort the population from best to worst*
 5. *Identify the leader and follower salps*
 6. **while** *termination criterion is not met* **do**
 7. **for each** *leader salp* **do**
 8. **if** $\text{rand}(0,1) < 0.5$ **then**
 9. // *Quasi-oppositional movement*
 10. Apply Eqs. 14-16.
 11. **Else**
 12. // *Jaya algorithm*
 13. Apply Eq. 19
 14. **end if**
 15. **end for**
 16. **for each** *follower salp* **do**
 17. // *Salp chain movement*
 18. Apply Eq. 20
 19. **end for**
 20. *Check boundaries*
 21. *Update the best solutions*
 22. *Sort the population from best to worst*
 23. *Identify the leader and follower salps*
 24. **end while**
-

3.3 Implementation of the proposed OPSSAJ algorithm

The step-wise description for the execution of the proposed OPSSAJ is given below.

Step 1 Initialize the set of algorithm parameters and define the objective function. Population size and dimension of the decision variables are respectively set to N and D . Objective function $f()$ is set and upper and lower bounds (UB and LB) of the search space are defined.

Step 2 Initialize the population within the prescribed search boundaries. Randomly generated solutions $S = [S]_{N \times D}$ can be described by the following matrix expression,

$$S_{1, \dots, N, 1, \dots, D} = \begin{bmatrix} S_{11} & S_{12} & \dots & S_{1D} \\ S_{21} & S_{22} & \dots & S_{2D} \\ \vdots & \vdots & & \vdots \\ S_{N1} & S_{N2} & \dots & S_{ND} \end{bmatrix}$$

Subsequently, the solution quality of each member is evaluated by $f = f(S_i)$, $i = 1, \dots, N$. Selected the best and the worst quality member based on their solution fitness.

Step 3 Sort the population from best to worst and apply randomly Jaya and Quasi-oppositional based learning for the best half ($N/2$) of the population, which are the leader salps. The pseudocode of this step is given in Algorithm 4.

Algorithm 4 Pseudocode of the Step 3

```

for the each leader salp in the best half do
  if rand([0, 1]) < 0.5
    Apply Quasi-opposition based solution update with Eqs. 14-16
  else
    Apply Jaya movement with Eq. 19
  end
end
end

```

Algorithm 5 Pseudocode of the Step 4

```

for the each follower in the worst half do
  Apply the chain movement defined by the expression given in Eq. 20
end
end

```

Step 4 For the worst half part of the population which are known as followers, apply the chain movement method given in Algorithm 5.

Step 5 Check the boundaries and restrict the violated solution into the predefined search ranges.

Step 6 Update the current solution.

Step 7 Repeat Step 3 to Step 6 until the termination criterion is reached.

Step 8 Output the best solution.

4 Numerical investigations over unconstrained benchmark suite

In this study, 24 benchmark functions comprising of multimodal and unimodal problems with 30 dimensions have been used to test the performance of the proposed algorithm. The statistical results obtained from the OPSSAJ method are compared with the state-of-the-art algorithms including colliding bodies algorithm (CB) [38], cuckoo search algorithm (CS) [39], fruit fly algorithm (FF) [40], tree-seed algorithm (TS) [41], spotted hyena algorithm (SH) [42], emperor penguin algorithm (EPA) [43], vibrating particles system algorithm (VPS) [44], butterfly algorithm (BF) [45], jaya algorithm [12], crow search algorithm [46]. The above-mentioned algorithms have essential parameters that are tuned employing a trial-and-error approach. The selection of the best parameters is highly dependent on the problem characteristics. In general, finding the best parameter is a time-consuming and tedious task that incurs a considerable amount of computation time. Table 1 reports the statistical results of the multimodal benchmark functions, which are collected after 50 independent replications. The termination criterion is set to 4000 function evaluations. The algorithms are developed in Java and computations are run on an Intel Core i5 computer with

3.00 GHz CPU and 6 GB RAM. The best, worst, mean, and standard deviation values of the multi-modal test functions are reported in Table 1. According to results, OPSSAJ method finds superior results than its counterparts in f_1 -Ackley, f_4 -Zakharov, f_5 -Alpine, f_7 -Csendes, f_9 -Salomon, f_{10} -Inverted cosine, and f_{12} -Pathological functions. On the other hand, BFA algorithm finds slightly better solutions in f_2 -Griewank, f_3 -Rastrigin, f_8 -Schaffer, and f_{11} -Wavy functions. BFA algorithm produces better results only in f_6 -Quintic function. Apart from the best solutions, OPSSAJ outperformed other optimizers in terms of robustness and solution accuracy. Analysis of the standard deviation values reveals that the OPSSAJ gives robust results with small standard deviations. On the other hand, the worst solutions obtained by OPSSAJ are usually better than those found by the second-best optimizer in comparison.

Figures 1, 2 depict the convergence charts of the compared algorithms for each multimodal test function. The x-axis represents the number of function evaluations, and the y-axis shows the best objective function value in the logarithmic scale obtained during iterations. The convergence performance of OPSSAJ is highly satisfactory in comparison with the remaining algorithms. The evolution history of the OPSSAJ method clearly shows that the algorithm reaches its optimum point gradually and consistently, while other algorithms stagnate at the early phases of the search, which is due to the lack of exploration capability. The OPSSAJ takes firm steps forward to the global optimum point as a result of a plausible balance between intensification and diversification. Table 2 gives the statistical results of unimodal test functions. OPSSAJ has found better solutions in f_{13} -Sphere, f_{15} -Schwefel 2.22, f_{16} -Schwefel 2.23, f_{18} -Brown, f_{20} -Powell, f_{21} -Sum of different powers, f_{22} -High conditioned elliptic, f_{23} -Sum squares, and f_{24} -Bent cigar functions. BFA algorithm has found better solutions in f_{10} -Stretched V Sine Wave functions. Jaya

Table 1 Statistical results for multimodal optimization test functions

	Best	SD	Mean	Worst
<i>f₁-Ackley</i>				
OPSSAJ	1.2123E-13	1.6167E-10	6.0991E-11	1.2654E-09
SSA	1.3161E-01	1.6412E-01	5.1954E-01	1.0925E+00
CB	7.0623E-01	5.8377E-01	2.0226E+00	3.6066E+00
CS	8.0625E-01	1.7801E-01	1.1003E+00	1.6204E+00
FF	2.4094E-03	7.1331E-06	2.4274E-03	2.4479E-03
TS	1.0242E-01	3.4091E-02	1.7184E-01	3.0227E-01
SH	2.0771E-06	1.1691E-04	1.1751E-04	5.3157E-04
EPA	4.7939E-03	5.0779E-01	2.6711E-01	2.7467E+00
VPS	7.3117E-01	3.0934E-01	1.2979E+00	2.1677E+00
BFA	4.3076E-14	3.0767E-15	4.8405E-14	5.0182E-14
JAYA	3.0637E-01	1.2647E-01	6.4801E-01	9.6289E-01
CROW	3.0342E-01	1.3601E-01	5.2771E-01	8.0193E-01
<i>f₂-Griewank</i>				
OPSSAJ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
SSA	2.1396E-02	6.3798E-02	1.2054E-01	3.0784E-01
CB	9.0104E-02	1.6965E-01	3.2266E-01	9.3721E-01
CS	1.4268E-01	1.1389E-01	3.2873E-01	6.5672E-01
FF	6.5072E-07	4.5682E-09	6.6387E-07	6.7349E-07
TS	1.0778E-03	1.6262E-02	7.1457E-03	1.3082E-01
SH	5.6212E-13	5.7263E-03	9.9875E-04	4.3321E-02
EPA	4.2177E-08	3.2763E-01	3.1394E-01	9.7193E-01
VPS	4.0810E-02	9.4332E-02	1.6761E-01	4.9993E-01
BFA	2.2204E-16	1.5666E-16	7.5615E-16	1.1102E-15
JAYA	2.3731E-02	1.8541E-01	1.5772E-01	6.7242E-01
CROW	1.2051E-02	1.8342E-02	4.1142E-02	1.0224E-01
<i>f₃-Rastrigin</i>				
OPSSAJ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
SSA	2.6386E+01	2.0964E+01	7.4444E+01	1.3947E+02
CB	7.1732E+01	2.5885E+01	1.2041E+02	1.9452E+02
CS	1.9024E+02	1.6324E+01	2.3472E+02	2.7539E+02
FF	2.0167E-03	1.1412E-05	2.1974E-03	2.2384E-03
TS	1.4592E+02	1.1842E+01	2.0034E+02	2.2174E+02
SH	1.2719E-08	3.6302E+01	3.3764E+01	1.6179E+02
EPA	1.8162E-01	1.3764E+02	1.6972E+02	4.9274E+02
VPS	5.1973E+01	4.2185E+01	1.2348E+02	2.3082E+02
BFA	1.7082E-13	9.8924E-14	3.4482E-13	6.8264E-13
JAYA	1.9562E+02	2.1028E+01	2.3692E+02	2.8969E+02
CROW	6.5615E+01	3.1696E+01	1.6953E+02	2.2863E+02
<i>f₄-Zakharov</i>				
OPSSAJ	1.7283E-21	1.6716E-14	4.0997E-15	1.4852E-13
SSA	1.0893E+01	1.0612E+01	3.0145E+01	6.3028E+01
CB	1.7032E+02	8.3073E+01	2.9483E+02	6.0242E+02
CS	1.7331E+02	1.7592E+02	3.8592E+02	9.5283E+02
FF	1.6982E-02	1.0116E-04	1.7221E-02	1.7972E-02
TS	2.8714E+01	8.4996E+01	5.0592E+01	7.2691E+01
SH	7.2512E-02	2.1582E+00	2.0362E+00	1.3864E+01
EPA	7.9958E+01	2.9564E+03	1.6294E+03	1.6671E+04
VPS	1.4294E+01	1.0284E+01	2.9567E+02	5.6821E+02
BFA	1.7482E-15	3.9275E+00	6.8614E-01	2.9224E+01
JAYA	4.0921E+01	1.7622E+01	8.8852E+01	1.3283E+02

Table 1 (continued)

	Best	SD	Mean	Worst
CROW	4.9555E+01	1.3255E+01	8.5729E+01	1.1872E+02
<i>f₅-Alpine</i>				
OPSSAJ	1.3823E-14	1.5769E-09	4.6598E-10	1.4038E-08
SSA	1.4997E+00	1.4749E+00	3.9391E+00	7.6927E+00
CB	5.3878E-01	1.2764E+00	2.7113E+00	6.3262E+00
CS	9.6868E+00	2.1223E+00	1.5942E+01	2.1064E+01
FF	1.8229E-03	3.9183E-06	1.8324E-03	1.8432E-03
TS	4.9585E-02	4.7583E-01	4.9755E-01	3.6169E+00
SH	6.2665E-07	3.1188E-01	4.6621E-02	2.9545E+00
EPA	3.2464E-04	8.1311E+00	6.5692E+00	2.6852E+01
VPS	5.7313E-01	1.2407E+00	2.3157E+00	5.9194E+00
BFA	3.6408E-14	1.4098E-14	5.8321E-14	1.1118E-13
JAYA	1.6214E+00	2.4525E+00	5.2782E+00	1.2567E+01
CROW	1.9924E-01	1.3222E+00	1.4412E+00	6.8632E+00
<i>f₆-Quintic</i>				
OPSSAJ	8.0992E+01	4.1242E+00	9.5172E+01	1.0242E+02
SSA	2.7794E+01	1.5822E+01	5.5298E+01	1.1184E+02
CB	1.7321E+01	1.2973E+01	1.4324E+02	5.9642E+02
CS	6.4742E+01	1.0442E+01	9.5321E+01	1.2252E+02
FF	1.1614E+02	4.1455E-04	1.1672E+02	1.1679E+02
TS	5.0281E+01	1.2762E+01	8.2141E+01	1.1242E+02
SH	2.6499E+01	1.4251E+01	6.1742E+01	1.1973E+02
EPA	9.5918E+01	3.8624E+03	3.9074E+03	1.3241E+04
VPS	1.5512E+01	1.7284E+01	3.7312E+01	1.3642E+02
BFA	1.0142E+02	1.4421E+03	1.8331E+03	4.2908E+03
JAYA	7.3762E+01	1.1552E+01	1.0382E+02	1.2694E+02
CROW	1.6892E+01	1.8778E+01	4.9625E+01	1.0152E+02
<i>f₇-Csendes</i>				
OPSSAJ	3.9237E-91	4.1726E-71	4.3299E-72	4.0672E-70
SSA	3.4461E-03	1.1583E+00	5.5892E-01	6.4602E+00
CB	3.5671E+01	1.5972E+03	1.0762E+03	1.3472E+04
CS	4.3891E-01	4.5698E+00	5.2018E+00	2.4338E+02
FF	1.5322E-17	7.9271E-17	5.7927E-17	5.9082E-16
TS	8.7621E-01	3.0242E+00	3.8641E+00	1.4197E+01
SH	4.9623E-20	4.7752E-06	1.2764E-06	3.1017E-05
EPA	1.3319E-03	3.0162E+04	2.8925E+04	9.0962E+04
VPS	1.1642E+00	5.8251E+01	5.3726E+01	3.6973E+02
BFA	4.6991E+03	7.2381E+03	2.3678E+04	3.5079E+04
JAYA	9.9421E-02	1.6271E+01	5.6712E+00	1.4082E+02
CROW	1.6728E-01	1.8742E+01	7.5529E+00	1.7725E+02
<i>f₈-Schaffer</i>				
OPSSAJ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
SSA	7.7324E-03	4.2711E-03	1.8822E-02	2.9542E-02
CB	5.5655E-02	1.2142E-02	9.4927E-02	1.2845E-01
CS	3.4641E-02	1.5242E-02	7.4092E-02	1.1082E-01
FF	5.5958E-09	1.1024E-02	1.7244E-02	4.2841E-02
TS	1.0972E-02	2.7321E-02	1.7051E-02	2.3712E-02
SH	3.1219E-03	3.0452E-03	7.2115E-03	1.9642E-02
EPA	3.1971E-02	2.0042E-02	9.1241E-02	1.2881E-01
VPS	1.1263E-02	4.9261E-03	2.1852E-02	3.5018E-02
BFA	7.7714E-16	2.6898E-14	4.7218E-15	2.2026E-13

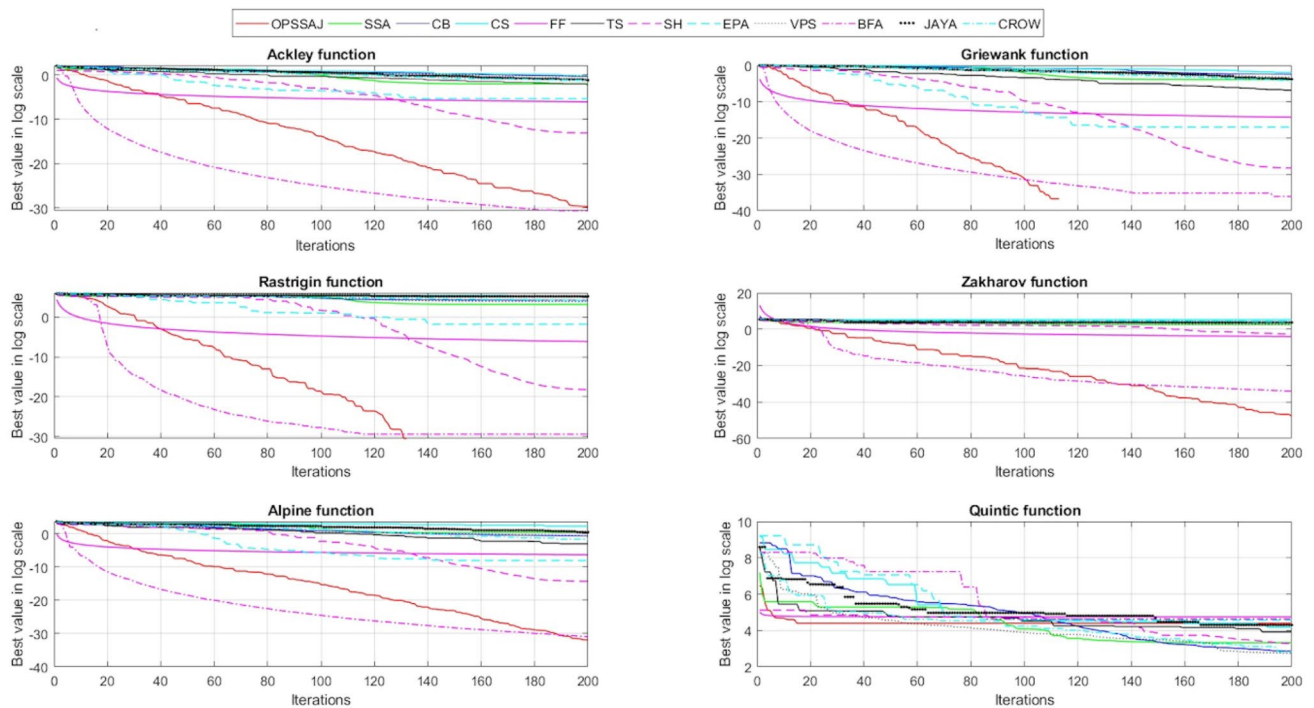
Table 1 (continued)

	Best	SD	Mean	Worst
JAYA	1.4792E-02	4.4492E-03	2.7782E-02	3.9193E-02
CROW	1.6743E-02	4.2472E-03	2.6879E-02	3.6521E-02
<i>f₉-Salomon</i>				
OPSSAJ	1.2707E-13	3.8526E-07	6.4998E-08	3.8762E-06
SSA	3.9982E-01	7.8912E-02	6.2861E-01	7.9921E-01
CB	9.9985E-01	1.2028E-01	1.3884E+00	1.6209E+00
CS	7.0041E-01	8.9261E-02	8.7556E-01	1.1008E+00
FF	1.1102E-01	4.1726E-01	1.3568E+00	1.9081E+00
TS	2.9991E-01	4.9273E-02	4.3929E-01	5.0293E-01
SH	1.9982E-01	7.8852E-02	3.4063E-01	4.9972E-01
EPA	4.0001E-01	3.0893E-01	1.2282E+00	1.7314E+00
VPS	4.9976E-01	7.6821E-02	6.4893E-01	7.9982E-01
BFA	1.5653E-13	2.9273E-02	9.2164E-02	1.4083E-01
JAYA	4.0021E-01	7.2612E-02	6.0924E-01	7.9726E-01
CROW	4.3514E-01	5.8761E-02	5.7721E-01	7.0863E-01
<i>f₁₀-Inverted cosine</i>				
OPSSAJ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
SSA	2.0231E+00	1.1719E+00	4.2981E+00	8.3214E+00
CB	2.9142E+00	4.9012E+00	1.1462E+00	2.8042E+00
CS	3.2647E+00	9.0513E-01	5.4492E+00	8.3752E+00
FF	1.4582E-04	6.6652E-07	1.4762E-04	1.4874E-04
TS	6.3182E-01	4.2671E-01	1.6927E+00	2.5402E+00
SH	1.0117E-10	1.9072E-06	6.1263E-07	1.1721E-05
EPA	8.4752E-05	1.0371E+01	2.4862E+00	7.4186E+01
VPS	4.6976E+00	2.6469E+00	9.4581E+00	1.6452E+01
BFA	3.5527E-15	1.3542E-15	5.4692E-15	9.7962E-15
JAYA	2.2525E+00	7.2276E-01	3.9937E+00	5.7532E+00
CROW	1.5097E+00	5.9721E-01	2.7682E+00	4.2789E+00
<i>f₁₁-Wavy</i>				
OPSSAJ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
SSA	3.0496E-01	8.8862E-02	5.1726E-01	7.1973E-01
CB	5.9529E-01	5.4173E-02	7.7352E-01	8.4781E-01
CS	7.4682E-01	1.9831E-02	8.1421E-01	8.5072E-01
FF	1.8456E-05	8.6669E-05	1.8627E-05	1.8817E-05
TS	5.7792E-01	2.9962E-02	6.7971E-01	7.2386E-01
SH	8.8081E-08	1.6064E-01	1.6489E-01	6.1572E-01
EPA	1.1582E-01	2.1529E-01	5.3725E-01	8.4561E-01
VPS	6.5073E-01	3.1465E-02	7.3298E-01	7.9642E-01
BFA	7.7715E-16	1.9902E-16	1.1312E-15	1.8873E-15
JAYA	6.7511E-01	2.3862E-02	7.5628E-01	7.9997E-01
CROW	6.2298E-01	3.1345E-02	7.1826E-01	7.6152E-01
<i>f₁₂-Pathological</i>				
OPSSAJ	8.3627E-24	5.5539E-17	6.3478E-18	5.6862E-16
SSA	3.9654E+01	2.1354E+02	3.6431E+02	1.0412E+03
CB	1.1082E+02	9.2462E+02	1.2223E+03	6.8124E+03
CS	1.8243E+02	1.2703E+02	4.1572E+02	7.2473E+02
FF	2.8964E-03	1.8643E-05	2.9347E-03	2.9714E-03
TS	1.0845E+00	1.1337E+00	2.9265E+00	6.9562E+00
SH	7.9364E-10	1.2101E-05	3.7261E-06	8.8793E-05
EPA	7.7525E-05	2.5268E+02	9.8651E+01	1.4471E+03
VPS	3.9771E+01	2.8461E+02	4.9575E+02	1.6073E+03

Table 1 (continued)

	Best	SD	Mean	Worst
BFA	6.4418E+03	5.4812E+03	1.9388E+04	2.8692E+04
JAYA	4.2733E+01	3.6971E+01	9.6812E+01	2.4799E+02
CROW	1.8774E+01	2.9574E+01	5.6810E+01	2.1782E+02

Bold values represent the best solution obtained by the optimization algorithm for corresponding problem

**Fig. 1** Evolution characteristics of the algorithms for the test functions f_1 - f_6

algorithm has managed to outperform other algorithms in only f_{17} -Schwefel 2.25 function. The Rosenbrock function is known to be one of the most challenging instances in which there are many long and carved valleys. OPSSAJ has found the solution of $2.7917E+01$ in the Rosenbrock function, which is much closer to the optimal solution in comparison with the results of other algorithms. The evolution histories of the algorithms for unimodal benchmark instances are depicted in Figs. 3, 4. As can be seen from the figures, OPSSAJ quickly converges to its optimum solution. In many instances, compared algorithms fall behind the OPSSAJ results in terms of solution quality and speed of convergence towards the global optimum point.

5 Thermodynamic modeling and optimization of Organic Rankine Cycle

5.1 Mathematical modeling of a thermal ORC system

The following assumptions have been made not only to simplify the optimization process but also to eliminate the excessive amount of the computational burden resulted from the iterative calculations [47].

- It is assumed that ORC operates under steady-state conditions and running fluid at the outlet of the condenser and evaporator is presumed to be saturated
- Leakages from the pipes and heat losses to the surroundings are neglected
- Shell and tube heat exchangers are considered for condenser and evaporator

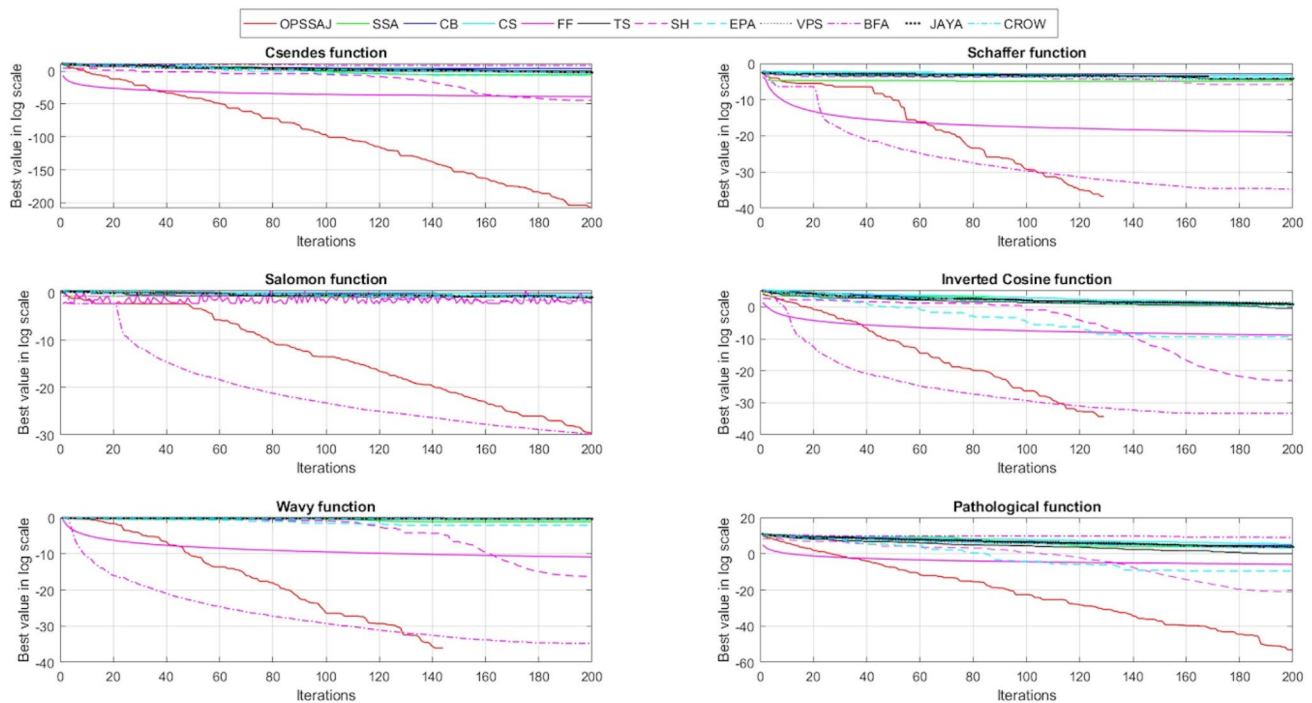


Fig. 2 Evolution characteristics of the algorithms for the test functions f_7 - f_{12}

- Thermal analysis of the cycle disregards the effects of kinetic and potential energy changes
- Heat transfer calculations are made based on fully developed flow conditions

There are also some precautionary system design constraints taken such that evaporator pressure should not exceed 90% of the critical pressure of the running cycle fluid. All design constraints considered in the optimization problem are modeled as hard inequality constraints. Obeying the above-given assumptions, mathematical modeling ORC is constructed in step-by-step formulations. Mass and energy balance equations along with irreversibility generation formulations govern the basics of thermal modeling of ORC. Heat and mass balance equations are related to the first law of thermodynamics while irreversibility equations are concerned with the second law of thermodynamics. Mass balance is expressed by,

$$\sum \dot{m} = 0 \tag{21}$$

Energy balance in each cycle component is maintained by using the merits of first law and equated by

$$\sum \dot{Q} + \sum \dot{W} + \sum (\dot{m} \cdot h) = 0 \tag{22}$$

The irreversibility equation reflects the effects of entropy generation. If a substance changes its state from

a to b, irreversibility generation of the system is formulated by the following equation

$$I_{a \rightarrow b} = T_0 \cdot \left(S_b - S_a - \frac{\dot{Q}_0}{T_0} - \frac{\dot{Q}_{source(sink)}}{T_{source(sink)}} \right) = T_0 \cdot \Delta S_{gen} \geq 0 \tag{23}$$

The above equation can be applied to determine the current irreversibility rate of the working fluid flowing through the related cycle component. Based on the formulations given in the Electronic Supplementary Material, second law efficiency of the cycle, which is also one of the design objectives considered in this study, is calculated by the following expression

$$\eta_{II} = W_{net} / (W_{net} + \dot{I}_{tot}) \tag{24}$$

where \dot{I}_{tot} is the total irreversibility of the cycle.

Mathematical modeling of condenser and evaporator takes a great role in determining the overall efficiency and cost of the thermal cycle. It is known that the investment cost of the heat exchanger is directly related to the total heat exchange area, which is a function of the overall heat transfer coefficient of the system. Therefore, the utmost concern should be given to the proper use of the correct heat transfer correlation. Although there are plenty of correlations available in the literature for modeling two-phase flow heat transfer occurring in condenser and evaporator, they are correlated with their experimental conditions. For

Table 2 Statistical results for unimodal optimization test functions

	Best	SD	Mean	Worst
<i>f₁₃-Sphere</i>				
OPSSAJ	2.6796E-26	3.8134E-19	4.4885E-20	4.1191E-18
SSA	9.0285E-02	3.5214E-01	5.4261E-01	1.8145E+00
CB	1.7762E+00	4.8759E+00	8.0954E+00	2.4794E+01
CS	7.9651E-01	8.6287E-01	2.4763E-01	5.8002E+00
FF	1.0925E-05	5.3121E-08	1.1021E-05	1.1248E-05
TS	2.0212E-02	2.5691E-02	5.7483E-02	1.5619E-01
SH	1.8492E-10	2.1974E-07	9.6491E-08	1.5382E-06
EPA	5.2636E-06	3.8261E+00	1.3487E+00	1.8142E+01
VPS	4.4741E-01	1.7729E+00	3.2543E+00	8.7501E+00
BFA	2.2124E-16	2.2671E-16	5.1792E-16	1.3762E-15
JAYA	3.4867E-01	3.6054E-01	8.3764E-01	2.3964E+00
CROW	1.5964E-01	4.2462E-01	7.0352E-01	2.2991E+00
<i>f₁₄-Rosenbrock</i>				
OPSSAJ	2.7917E+01	1.3947E-01	2.8894E+01	2.9941E+01
SSA	7.0183E+01	1.2694E+02	2.5512E+02	1.0382E+03
CB	4.7576E+02	4.1482E+03	4.7781E+03	2.5448E+04
CS	2.1973E+02	1.8163E+02	5.7792E+02	1.2724E+03
FF	2.8661E+01	7.6695E-03	2.8654E+01	2.8701E+01
TS	1.5893E+02	9.3814E+01	2.9374E+02	6.6983E+02
SH	2.8234E+01	1.3087E+00	2.8486E+01	3.6917E+01
EPA	3.0162E+01	8.1472E+04	5.1867E+04	3.0271E+05
VPS	2.9371E+02	6.1374E+02	1.1212E+03	3.7592E+03
BFA	6.7262E+04	2.0211E+04	1.0887E+05	1.4918E+05
JAYA	9.5231E+01	7.9182E+01	2.3249E+02	4.7927E+02
CROW	1.3289E+02	1.2064E+02	3.3376E+02	9.5982E+02
<i>f₁₅-Schwefel 2.22</i>				
OPSSAJ	7.3617E-14	1.8451E-10	5.4337E-11	1.7274E-09
SSA	3.0689E+00	1.7412E+00	6.6802E+00	1.1754E+01
CB	1.2204E+00	3.5982E+00	6.8264E+00	2.4182E+01
CS	9.1020E+00	4.2764E+00	1.5134E+01	2.9246E+01
FF	1.8017E-02	4.1372E-05	1.8193E-01	1.8279E-02
TS	1.1974E-01	3.6342E-02	1.7348E-01	3.1194E-01
SH	9.4401E-08	6.4862E-06	2.8394E-06	5.5079E-05
EPA	1.8698E-04	3.6882E-02	2.4072E-02	1.8572E-01
VPS	1.2071E+00	1.5659E+00	4.2582E+00	8.2587E+00
BFA	7.7881E-14	4.3681E-14	1.5398E-13	2.7641E-13
JAYA	1.5789E+00	6.2974E-01	2.7974E+00	4.3428E+00
CROW	1.0224E+00	3.3561E-01	1.5397E+00	2.4292E+00
<i>f₁₆-Schwefel 2.23</i>				
OPSSAJ	1.0763E-127	1.8413E-85	1.6403E-86	2.0832E-84
SSA	8.9113E-05	3.8681E+00	1.5192E+00	3.1182E+01
CB	2.5724E+02	2.4824E+05	1.1986E+05	2.0314E+06
CS	3.3642E-01	1.5682E+01	1.0471E+01	9.1797E+01
FF	2.0282E-31	5.8756E-33	2.1409E-31	2.2989E-31
TS	5.8601E-01	4.7072E+01	4.1372E+01	2.5592E+02
SH	3.3619E-27	1.6052E-06	2.7617E-07	1.2462E-05
EPA	9.8838E-04	5.9418E+06	7.1457E+06	1.9470E+07
VPS	6.2033E+00	1.0842E+03	4.7521E+02	6.0523E+03
BFA	5.7554E+05	6.9241E+05	1.9652E+06	3.8054E+06
JAYA	4.4721E-02	8.0411E+01	1.8772E+01	7.5522E+02

Table 2 (continued)

	Best	SD	Mean	Worst
CROW	3.5398E-02	8.1122E+01	3.6254E+01	5.4215E+02
<i>f₁₇-Schwefel 2.25</i>				
OPSSAJ	2.3394E+01	7.8214E-01	2.6174E+01	2.7652E+01
SSA	3.2745E+00	7.1462E+00	1.7012E+01	4.2769E+01
CB	1.5639E+01	6.7521E+01	1.2021E+02	3.4988E+02
CS	2.2194E+01	1.0973E+01	3.7651E+01	8.4408E+01
FF	1.8672E+01	6.4442E-01	2.0331E+01	2.1671E+01
TS	6.4578E+00	7.7621E+00	1.8261E+01	5.2769E+01
SH	1.2863E+01	1.7347E+00	1.7711E+01	2.2114E+01
EPA	2.5361E+01	7.3081E+02	9.2868E+02	2.8313E+03
VPS	1.6782E+01	3.5092E+01	6.9072E+01	1.7257E+02
BFA	2.5814E+01	3.4571E+02	2.2198E+02	1.1158E+03
JAYA	1.3222E+00	2.8033E+00	4.4453E+00	1.6114E+01
CROW	6.0483E+00	1.2031E+01	1.9762E+01	5.7543E+01
<i>f₁₈-Brown</i>				
OPSSAJ	3.7291E-26	9.9143E-19	1.3124E-19	9.2393E-18
SSA	2.8873E-01	9.2515E+00	6.3834E+00	5.4208E+01
CB	1.0254E+01	3.4076E+11	3.1995E+10	3.6862E+12
CS	1.2149E+01	5.5341E+03	1.6238E+03	5.2148E+04
FF	2.1124E-05	1.0532E-07	2.1347E-05	2.1595E-05
TS	2.1452E-01	2.4023E-01	6.1393E-01	1.3729E+00
SH	7.4872E-10	1.8800E-06	5.7534E-07	1.2971E-05
EPA	7.3868E-06	7.0024E+24	8.1402E+23	6.1052E+25
VPS	1.6182E+00	9.1409E+00	1.2542E+01	6.4198E+01
BFA	1.4206E-16	1.4962E+10	3.4989E+09	9.7736E+10
JAYA	1.3138E+00	3.2282E+00	5.4321E+00	1.9851E+01
CROW	1.0675E+00	1.1834E+00	2.6983E+00	7.3983E+00
<i>f₁₉-Stretched V Sine Wave</i>				
OPSSAJ	2.5641E-07	5.6535E-06	4.5714E-06	3.0631E-05
SSA	2.5471E+01	3.5829E+00	3.4781E+01	4.3018E+01
CB	1.8062E+01	4.8289E+00	3.1824E+01	4.4172E+01
CS	3.0997E+01	3.1972E+00	3.9532E+01	4.7264E+01
FF	6.6092E+00	1.3594E+00	8.0307E+00	1.0852E+01
TS	1.2974E+01	1.4624E+01	1.6984E+01	1.9642E+01
SH	9.0642E-03	6.7521E-01	6.9972E-01	3.6521E+00
EPA	2.6821E-01	9.2164E+00	5.8164E+00	4.6864E+01
VPS	2.3012E+01	2.8364E+00	3.0624E+01	3.7521E+01
BFA	5.8242E-11	3.2917E-11	1.0682E-10	1.8572E-10
JAYA	2.2671E+01	3.4042E+00	3.0097E+01	3.8669E+01
CROW	1.3993E+01	2.5381E+00	2.0188E+01	2.4722E+01
<i>f₂₀-Powell</i>				
OPSSAJ	3.8087E-25	6.3463E-18	1.3432E-18	5.9178E-17
SSA	4.9691E+00	9.7162E+01	1.1124E+02	6.3589E+02
CB	3.2197E+02	2.1472E+03	3.2016E+03	1.1939E+04
CS	1.6792E+02	2.4994E+02	5.3972E+02	1.3836E+03
FF	1.1734E-03	5.4358E-06	1.1984E-03	1.2082E-03
TS	8.9601E+00	1.3826E+01	2.5681E+01	7.5824E+01
SH	2.8149E-08	1.5454E-04	6.4518E-05	8.4415E-04
EPA	5.5012E-02	3.6503E+04	1.9776E+04	1.4824E+05
VPS	2.0271E+02	6.3874E+02	8.5326E+02	3.4372E+03
BFA	2.8555E+04	1.3539E+04	5.9986E+04	8.9220E+04

Table 2 (continued)

	Best	SD	Mean	Worst
JAYA	3.6492E+01	8.9075E+01	1.4655E+02	6.2074E+02
CROW	4.1927E+01	9.4228E+01	1.6615E+02	4.5072E+02
<i>f₂₁–Sum of different powers</i>				
OPSSAJ	1.6203E-40	1.6231E-15	2.3266E-16	1.8482E-14
SSA	7.0942E-02	3.0942E+02	5.6985E+01	3.8089E+03
CB	4.0067E+03	1.8081E+10	4.0824E+09	1.6193E+11
CS	6.2974E+01	1.6428E+04	1.1234E+04	7.3691E+04
FF	3.9512E-04	2.5906E-06	4.0008E-04	4.0824E-04
TS	6.3511E-01	8.9142E+00	7.1269E+00	5.1411E+01
SH	9.7366E-27	4.4982E-03	5.5766E-04	3.6826E-02
EPA	9.3712E-04	1.5928E+13	3.8351E+12	1.1149E+14
VPS	3.5571E-01	1.9025E+04	8.2056E+03	1.2367E+05
BFA	8.7194E-15	1.6933E+06	3.6079E+05	1.17334E+07
JAYA	1.9833E+01	8.8667E+01	1.6202E+02	5.6442E+02
CROW	6.8341E-02	7.6182E+01	1.6268E+01	5.8053E+02
<i>af₂₂–High conditioned elliptic</i>				
OPSSAJ	1.5861E-23	5.8762E-14	6.5481E-15	6.9576E-13
SSA	6.6464E+03	3.2309E+04	6.4013E+04	1.8981E+05
CB	4.7193E+03	6.1451E+04	4.0810E+04	6.0491E+05
CS	1.4586E+04	1.1024E+04	3.4762E+04	7.5878E+04
FF	7.0048E-01	5.9065E-03	7.1192E-01	7.2574E-01
TS	6.7429E+00	3.9221E+00	1.2559E+01	2.6552E+01
SH	1.5745E-09	7.2741E-06	2.7596E-06	4.3961E-05
EPA	1.9382E-02	1.0042E+03	3.9174E+02	5.8172E+03
VPS	1.1612E+03	4.7716E+03	6.1875E+03	2.9999E+04
BFA	1.0866E+06	5.6129E+05	2.1957E+06	3.3192E+06
JAYA	3.4983E+02	4.2683E+02	9.7821E+02	2.6543E+03
CROW	1.4558E+02	2.1342E+02	3.6792E+02	1.1785E+03
<i>f₂₃–Sum squares</i>				
OPSSAJ	2.0264E-26	1.0793E-18	2.0694E-19	1.0472E-17
SSA	3.4958E+00	1.1274E+01	2.0275E+01	6.2471E+01
CB	1.0463E+01	7.7115E+01	1.1873E+02	5.5892E+02
CS	1.1235E+01	8.7369E+00	3.2278E+01	6.1609E+01
FF	1.6437E-04	9.0975E-07	1.6649E-04	1.6844E-04
TS	2.6449E-01	2.2794E-01	6.2984E-01	1.2272E+00
SH	8.2987E-10	1.2020E-06	5.0778E-07	7.7481E-06
EPA	1.4451E-03	1.7834E+02	2.3234E+01	1.8684E+03
VPS	1.5552E+01	2.7618E+01	4.4867E+01	1.3395E+02
BFA	3.9801E-16	8.9991E-16	1.4802E-15	4.5692E-15
JAYA	2.9649E+00	5.4581E+00	1.0889E+01	3.5371E+01
CROW	1.0448E+00	4.0940E+00	7.3582E+00	2.0663E+01
<i>f₂₄–Bent cigar</i>				
OPSSAJ	4.8071E-22	3.6386E-13	4.2562E-14	4.4102E-12
SSA	5.6684E+04	2.7439E+05	4.1723E+05	1.8112E+06
CB	1.1688E+06	4.2751E+06	7.3259E+06	2.2087E+07
CS	9.3472E+05	5.7403E+05	2.0281E+06	3.6546E+06
FF	1.0507E+01	5.8265E-02	1.0642E+01	1.0804E+01
TS	9.1742E+03	1.6772E+04	3.5448E+04	1.1715E+05
SH	7.4047E-06	7.1615E-02	3.7451E-02	4.4021E-01
EPA	2.4883E+01	2.6622E+06	6.9538E+05	1.8231E+07
VPS	8.7924E+05	2.2994E+06	3.4812E+06	1.2380E+07

Table 2 (continued)

	Best	SD	Mean	Worst
BFA	7.4658E+07	1.0575E+07	1.1431E+08	1.3505E+08
JAYA	1.5501E+05	3.1942E+05	6.6452E+05	1.6251E+06
CROW	1.6395E+05	2.9961E+05	5.3021E+05	2.2501E+06

Bold values represent the best solution obtained by the optimization algorithm for corresponding problem

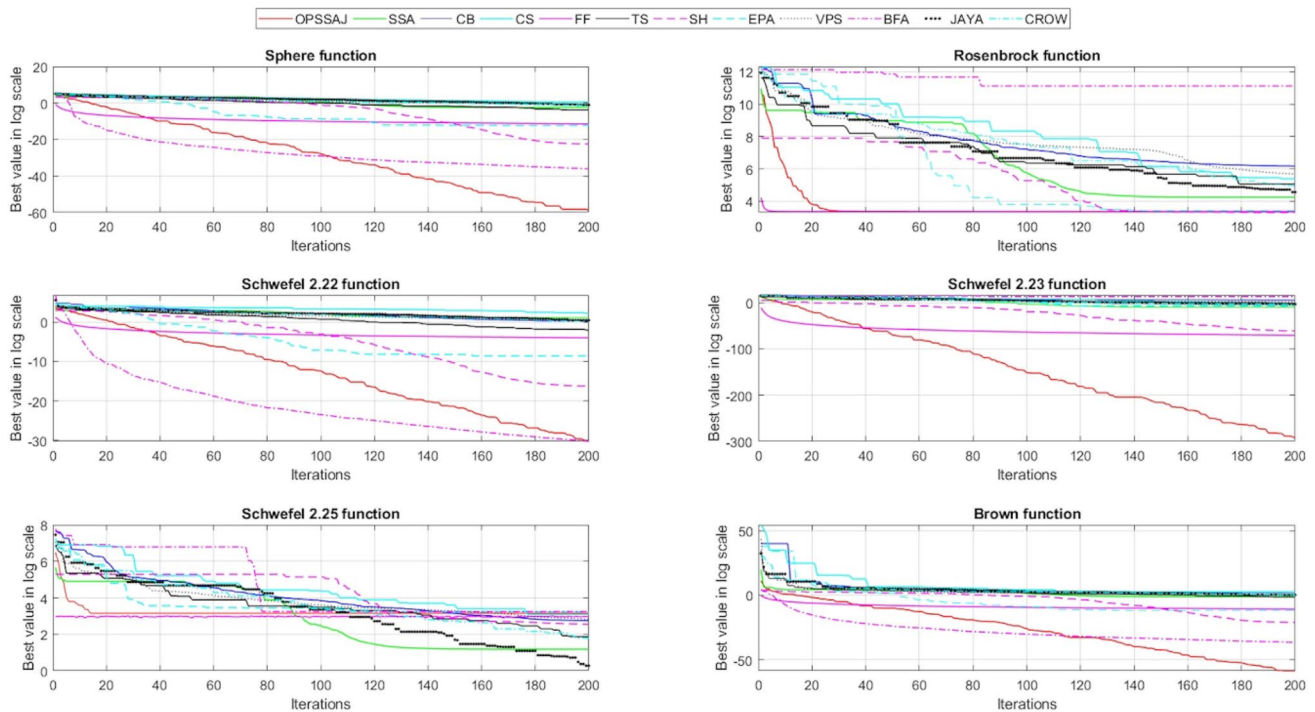


Fig. 3 Evolution characteristics of the algorithms for the test functions f_{13} - f_{18}

this reason, it is a huge task for a designer to use the correct heat transfer correlation for the governing operational conditions.

Gnielinski [48] correlation, which is a highly reputed equation for modeling single-phase flow, is used to evaluate the heat transfer mechanism taking place in the evaporator and condenser. Correlation developed by Thome [49] is a modified form of Gungor-Winterton [50] correlation which takes into account the contribution of binary mixtures on flow boiling mechanism and used for estimating the evaporative heat transfer coefficient. Florides et al. [51] correlation is used to calculate the total amount of heat transfer in the annulus flow, which takes place between outer and inner tube regions. Shah [52] correlation is very effective in predicting the in-flow condensation heat transfer rates and so is used in this study for thermal modeling of the condenser. Therminol VP-1 is used as a secondary fluid that conveys the available low grade heat to the working fluid of the system. Cooling water absorbs the rejected heat in the condenser and considered as a

secondary fluid for this process. Shell and tube type heat exchanger is used for condenser and evaporator and heat exchange area of each cycle component is calculated by the below given generic formulation

$$A = \frac{\dot{Q}}{U \cdot \Delta T_{LMTD}} \tag{25}$$

where A is the total heat exchange surface, Q is the imposed heat load on the heat exchanger, ΔT_{LMTD} is the logarithmic mean temperature difference between two streams, and U is the overall heat transfer coefficient computed by the following equation

$$U = \frac{1}{\left(\left(\frac{1}{h_{out}} \right) + R_{out} + \left(\frac{d_{out}}{2k} \right) \ln \left(\frac{d_{out}}{d_{in}} \right) + R_{in} + \left(\frac{d_{out}}{d_{in}} \right) \left(\frac{1}{h_{in}} \right) \right)} \tag{26}$$

Applying reliable capital cost correlations to cycle components is also very crucial in obtaining the best possible thermoeconomic design outcomes. However, this

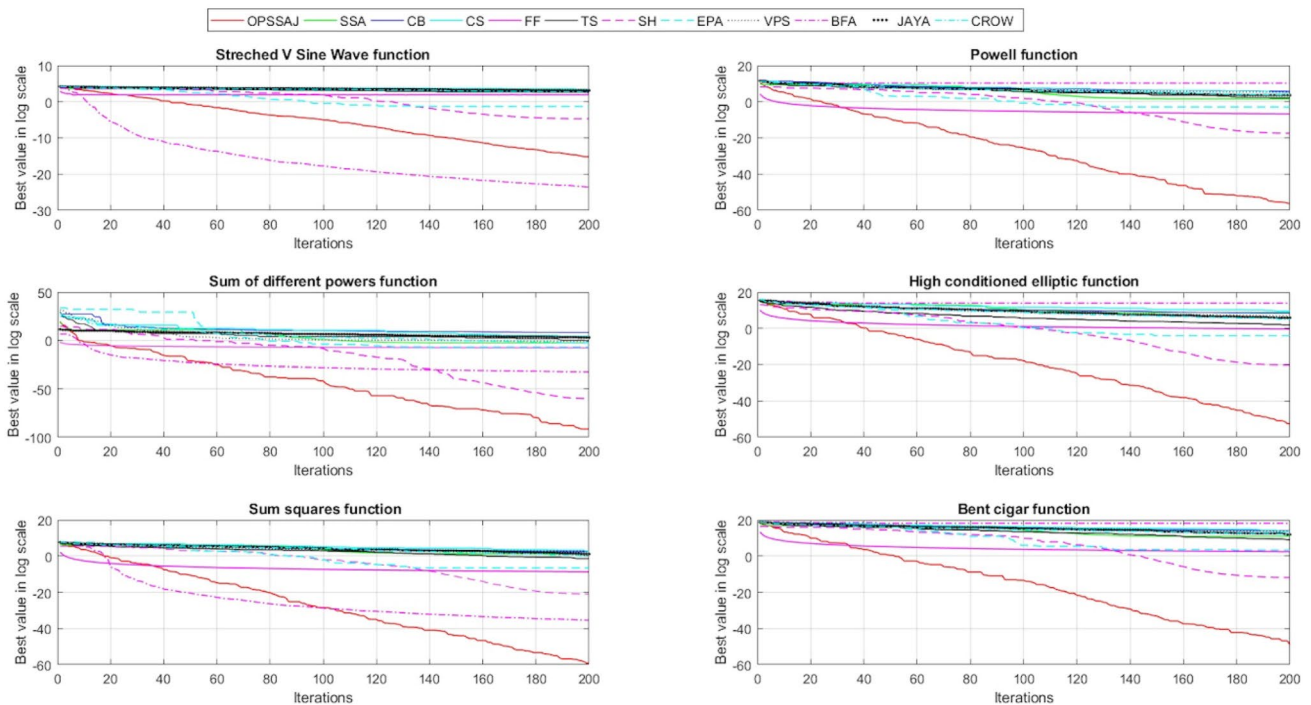


Fig. 4 Evolution characteristics of the algorithms for the test functions f_{19} - f_{24}

important issue has not been carefully analyzed in most of the literature studies regarding ORC design. This is because most of the available cost model is correlated for a very specific experimental region of temperatures and pressures. Economic information of cycle components is generally not conveyed by the component manufacturers, however affordable data from different sources is sufficient enough to develop reliable cost correlations. Among the various type of cost correlations available in the literature, the most reputed alternative is the chemical engineering plant cost index (CEPCI) whose updated model parameters have been reported in Chemical Engineering publication since 1963 [53]. Most of the published studies in the literature use the correlations developed by Turton et al. [54] to estimate the equipment investment cost of ORC components and apply the CEPCI index to each component to obtain their updated and current cost values.

Based on the proposed economic model, specific investment cost (SIC) rate is the ratio between total ORC system cost and net power output which is expressed by the following equation [55]

$$SIC = \frac{C_{cap} \cdot CRC + C_{mi}}{W_{net} \cdot h} \tag{27}$$

where C_{cap} is the capital cost accounting for the investment cost of the ORC elements of the evaporator, condenser, expander, fans, pump and working fluid.

$$C_{cap} = (C_{evap,bm} + C_{exp,bm} + C_{pump,bm} + C_{cond,bm}) \frac{CEPCI_{2019}}{CEPCI_{1996}} + C_{fan,bm} \left(\frac{CEPCI_{2019}}{CEPCI_{2011}} \right) + C_{wf,bm} \tag{28}$$

wherein $CEPCI_{2019} = 619$, $CEPCI_{2011} = 582$, and $CEPCI_{1996} = 382$ [56]. In Eq. (29), CRC represents the capital recovery cost which is calculated by

$$CRC = \frac{i(1+i)^{LT_{plant}}}{((1+i)^{LT_{plant}} - 1)} \tag{29}$$

where i symbolizes the interest rate which is considered to be 8% in this case; LT_{plant} is the plant life time and assumed to be 15 years of operation; h stands for the working hours of the ORC system per annual year and presumed to 8000 h. Insurance and management cost of the system is expressed by C_{mi} and estimated by [57]

$$C_{mi} = 0.165 \cdot C_{cap} \tag{30}$$

Cost correlations for expanders are expressed as

$$\log C_{exp} = K_{1,exp} + K_{2,exp} \log(W_{Exp}) + K_{3,exp} (\log(W_{Exp}))^2 \tag{31}$$

$$C_{exp,bm} = C_{exp} \cdot F_{Exp,bm} \tag{32}$$

Investment cost correlations for the pump are formulated by

$$\log C_{pump} = K_{1,pump} + K_{2,pump} \log (W_{pump}) + K_{3,pump} \log (W_{pump})^2 \quad \log C_{fan} = K_{1,fan} + K_{2,fan} \log \dot{V} + K_{3,fan} (\log \dot{V})^2 \quad (43)$$

$$P_{diff} = P_{evap} - P_{cond} \quad C_{fan,bm} = C_{fan} \cdot F_{fan,p} \cdot F_{fan,bm} \quad (44)$$

$$\log F_{pump,p} = C_{1,pump} + C_{2,pump} \log (p_{diff}) + C_{3,pump} \log (p_{diff})^2 \quad \text{For working fluid cost} \quad C_{wf,bm} = c_{wf} \cdot m_{wf} \quad (45)$$

$$F_{pump,bm} = B_{1,pump} + B_{2,pump} \cdot F_{m,pump} \cdot F_{pump,p} \quad \text{where } c_{wf} \text{ is the unit price of the working fluid in } \$/\text{kg} \text{ and } m_{wf} \text{ is the mass of the working fluid circulating through the cycle in kg.}$$

$$C_{pump,bm} = C_{pump} \cdot F_{pump,bm} \quad (37)$$

For calculation of heat exchanger (HEX) cost in the cycle (valid for both condenser and evaporator)

$$\log F_{HEX,p} = C_1 + C_2 \log (p_{HEX} - p_0) + C_3 \log (p_{HEX} - p_0)^2 \quad (38)$$

$$F_{HEX,bm} = B_{1,HEX} + B_{2,HEX} \cdot F_{m,HEX} \cdot F_{HEX,p} \quad (39)$$

$$\log C_{HEX} = K_{1,HEX} + K_{2,HEX} \log (A_{HEX}) + K_{3,HEX} \log (A_{HEX})^2 \quad (40)$$

$$C_{HEX,bm} = C_{HEX} \cdot F_{HEX,bm} \quad (41)$$

For the investment cost of the fan,

$$\log F_{fan,p} = C_{1,fan} + C_{2,fan} \log p_{fan} + C_{3,fan} (\log p_{fan})^2 \quad (42)$$

5.2 Discussion on the optimization results

Numerical modeling of the thermal system and the proposed OPSSAJ algorithm are both developed in Java environment. Due to the intrinsic random behavior of the hybrid metaheuristic OPSSAJ optimizer, simulations are run for 50 times for each optimization objective and the best results are collected and evaluated in terms of statistical analysis. The developed program is run at quad-core Intel i5-4460 @ 3.20 GHz with 16.0 GB RAM on a desktop computer. Table 3 reports the single objective optimization results of the ORC system running with different single working fluids obtained by OPSSAJ. For the cycle operating with R227EA refrigerant, it is seen that minimum obtained SIC value is 7529.457 (\$/W) while the maximum second law efficiency of the cycle 0.431 when these

Table 3 Single objective optimization results of the ORC system obtained with OPSSAJ

Designvariables	R227EA		R600		R227EA/R600	
	SIC min	Sec. eff. max	SIC min	Sec. eff. max	SIC min	Sec. eff. max
Condenser outlet temperature (K)	303.150	303.150	303.202	303.263	303.153	303.150
Evaporation pressure or temperature (MPaorK)	2.289	2.289	1.479	2.100	373.117	373.149
Superheat temperature (K)	29.998	17.168	6.379	19.921	0.002	15.000
Evaporator pinch temperature (K)	19.999	8.000	14.216	8.014	19.498	8.000
Condenser pinch temperature (K)	14.995	5.000	14.965	5.091	14.999	5.000
Condenser outertube diameter (m)	0.015	0.015	0.015	0.015	0.015	0.015
Condenser shell diameter (m)	0.250	0.499	0.276	0.496	0.281	0.499
Condenser baffles spacing (m)	0.217	0.497	0.182	0.489	0.457	0.499
Condenser number of tube passes	4	1	6	1	4	2
Condenser tube arrangement	Triangular	Triangular	Triangular	Triangular	Square	Square
Evaporator outertube diameter (m)	0.015	0.015	0.015	0.017	0.015	0.015
Evaporator shell diameter (m)	0.250	0.311	0.251	0.486	0.250	0.499
Evaporator baffle spacing (m)	0.150	0.499	0.157	0.485	0.150	0.499
Evaporator number of tube passes	8	1	8	1	8	2
Evaporator tube arrangement	Triangular	Square	Triangular	Square	Triangular	Square
Refrigerant mass flow rate(kg/s)	0.799	0.400	0.298	0.111	0.323	0.250
Mass fraction (R227EA/R600)	–	–	–	–	0.200	0.590
Second law efficiency	0.316	0.431	0.363	0.496	0.358	0.473
Specific investment cost (\$/W)	7529.457	54,137.679	5351.095	19,378.383	5295.332	24,905.764

conflicting objectives are separately optimized. Design variables of superheat temperature and evaporator pinch temperature hit the maximum allowable limits when SIC is minimized. Temperature mismatch between the heat source and evaporator increases with increasing superheat and pinch-point temperatures. This increase leads to a decline in the heat exchange area of the evaporator which entails a decrease in SIC rates. Similar tendencies can be observed for the condenser and heat sink temperature relationship. Outer tube diameters of the condenser and evaporator hit the prescribed lower bound which is 0.015 m. This is because when tube diameter values decrease, convective heat transfer coefficient of the running cycle fluid increases. This increase consequently gives rise in the net power output of the cycle, resulting in a decrease in SIC values. Triangular tube arrangement is observed for evaporator and condenser when SIC is minimized. Condenser and evaporator shell diameters reach the minimum allowable bounds. As shell diameters decrease, the cross-sectional area of the normal to secondary fluid flow decreases. This decrease results in an increase in secondary fluid flow velocity and consequently gives rise to overall heat transfer coefficient rates. Enhancement in heat transfer rates resulted from the increased convective heat transfer coefficient values increases the net power generation rates and decreases SIC of the cycle. Optimum refrigerant (R227EA) mass flow rate obtained by OPSSAJ is at its higher limit, which is 0.799 kg/s. One can easily see the relationship between the refrigerant mass flow rate and single and multi-phase convective heat transfer correlations reported in the Electronic Supplementary Material. It is observed from the correlations that convective heat transfer coefficient of the running fluid is directly influenced by the increasing flow rates, which eventually enhances the power generation of the ORC system. It is also observed that baffle spacing in the condenser and evaporator is closer to minimum allowable bound rather than its maximum. When ORC system running with R227EA is optimized to obtain maximum second law efficiency of the cycle, evaporator and condenser pinch point temperatures reach their minimum value. A thermodynamic system with minimum entropy generation is the system having higher exergetic (second law) efficiency. It is a thermodynamic principle that entropy of a system increases with the increasing temperature differences between the surroundings. Therefore, it is logical to obtain the minimum allowable value of these design variables, which reduces the temperature mismatch between two mediums and thereby increasing the exergetic efficiency of the cycle. Superheat temperature in the evaporator is obtained 17.168 K. Baffle spacing in the condenser and evaporator reaches its upper limits of 0.499 m. Triangular and square pitch arrangements are respectively

obtained for the condenser and evaporator. The optimum number of tube passes in the evaporator and condenser is found to be 1. Single optimization results of the ORC running with R600 working fluid indicates that minimum SIC and maximum second law efficiency values are correspondingly 5351.095 (\$/W) and 0.496. These objective function values are respectively 28.9% lower and 15.1% higher than those found by the cycle running with R227EA. Enhancement in the numerical value of these cycle performance indexes results from the intrinsic thermophysical characteristics of R600 refrigerant. The inclination of the design variables of ORC operating with R600 is in accordance with that of running with R227EA, except one case which is the superheat temperature in the evaporator when SIC is minimized. 6.379 K superheat is obtained which is closer to its lower limit contrary to the evaporator superheat of 29.998 K obtained by the cycle running with R227EA. It is also interesting to see that the design variable of condenser temperature reaches its minimum value for both cycles running with R227EA and R600. The net power output of the cycle is the direct function of the mass flow rate of the running refrigerant stream while it has no significant influence on second law efficiency rates as it is understood from Eq. (24). Regarding single objective optimization results obtained by the proposed hybrid algorithm reported in Table 3, Specific Investment Cost rate of the cycle gets its minimum value when mass flow rates hit the maximum value between the allowable value for ORC operated with R227EA refrigerant. A moderate mass flow rate of 0.363 kg/s, which is close to the minimum value between the defined search span, is obtained in the case of SIC minimization of ORC with R600 refrigerant. As respective Specific Investment Cost and second law efficiency rates acquired for R227EA and R600 refrigerants are comparatively evaluated, it is seen that ORC running with R600 is much more favorable to that of operated with R227EA because of its compatible thermal characteristics. In most of the related literature studies concerning with thermal design of ORC working with a binary mixture, the fluid composition is taken constant value and generally is not considered as a design variable. Parametric optimization is applied to observe the variational effects of mixture composition on the problem objectives. However, this type of optimization procedure has several shortcomings [28]. A significant drawback is that singularities can be observed in calculating the objective function values when the mixture composition is varied within the defined interval. This deficiency in calculation reduces the credibility and reliability of the obtained optimization results. For this important reason, mixture composition is considered as an accompanying decision variable along with remaining cycle design parameters. ORC with mixture fluid shows much better thermoeconomic performance in comparison

Table 4 Single objective optimization results of the ORC system obtained with NSGA-II

Designvariables	R227EA		R600		R227EA/R600	
	SIC min	Sec. eff. max	SIC min	Sec. eff. max	SIC min	Sec. eff. max
Superheat temperature (K)	27.156	17.823	7.355	19.412	0.218	13.527
Evaporator pinch temperature (K)	19.999	8.000	14.204	8.214	19.481	8.000
Condenser pinch temperature (K)	14.995	5.000	14.911	5.147	14.999	5.000
Condenser outertube diameter (m)	0.015	0.015	0.015	0.015	0.015	0.015
Condenser shell diameter (m)	0.250	0.499	0.282	0.491	0.295	0.497
Condenser baffles spacing (m)	0.287	0.412	0.141	0.420	0.418	0.496
Condenser number of tube passes	2	1	6	1	2	2
Condenser tube arrangement	Triangular	Triangular	Triangular	Triangular	Square	Triangular
Evaporator outertube diameter (m)	0.015	0.015	0.015	0.016	0.015	0.015
Evaporator shell diameter (m)	0.250	0.372	0.273	0.488	0.250	0.421
Evaporator baffle spacing (m)	0.150	0.499	0.163	0.488	0.150	0.418
Evaporator number of tube passes	4	2	6	1	8	4
Evaporator tube arrangement	Triangular	Triangular	Square	Square	Triangular	Square
Refrigerant mass flowrate(kg/s)	0.712	0.425	0.305	0.095	0.312	0.309
Mass fraction (R227EA/R600)	–	–	–	–	0.217	0.483
Second law efficiency	0.317	0.417	0.362	0.494	0.357	0.459
Specific Investment Cost (\$/W)	10,129.215	54,139.581	5763.108	19,380.506	6472.029	24,920.106

Table 5 Single objective optimization results of the ORC system obtained with MOPSO

Designvariables	R227EA		R600		R227EA/R600	
	SIC min	Sec. eff. max	SIC min	Sec. eff. max	SIC min	Sec. eff. max
Condenser outlet temperature (K)	303.482	303.623	303.479	303.508	303.496	303.602
Evaporation pressure or temperature(MPaorK)	2.418	2.317	1.512	2.125	373.125	373.147
Superheat temperature (K)	29.942	18.106	3.342	17.513	0.285	13.057
Evaporator pinch temperature (K)	19.296	9.147	14.308	8.519	14.269	9.157
Condenser pinch temperature (K)	14.257	5.563	14.951	5.287	14.572	6.107
Condenser outertube diameter (m)	0.015	0.015	0.015	0.015	0.015	0.015
Condenser shell diameter (m)	0.263	0.427	0.258	0.419	0.272	0.499
Condenser baffles spacing (m)	0.287	0.352	0.175	0.448	0.468	0.499
Condenser number of tube passes	4	2	4	2	2	1
Condenser tube arrangement	Square	Square	Square	Triangular	Square	Square
Evaporator outertube diameter (m)	0.015	0.015	0.015	0.017	0.015	0.015
Evaporator shell diameter (m)	0.258	0.308	0.258	0.474	0.250	0.499
Evaporator baffle spacing (m)	0.163	0.457	0.163	0.457	0.150	0.499
Evaporator number of tube passes	6	2	6	2	6	2
Evaporator tube arrangement	Triangular	Square	Triangular	Square	Square	Square
Refrigerant mass flow rate (kg/s)	0.682	0.328	0.291	0.099	0.328	0.255
Mass fraction (R227EA/R600)	–	–	–	–	0.259	0.574
Second law efficiency	0.315	0.431	0.362	0.493	0.357	0.464
specific investment cost (\$/W)	8146.109	54,137.679	5546.107	19,379.587	5439.057	24,912.062

with the cycle operating with pure fluids of R227EA and R600. The maximum exergetic efficiency of the ORC with mixture fluid is correspondingly 4.6% lower and 9.7% higher than that of the cycle operating with R600 and R227EA. Minimum SIC acquired by R227EA/R600 mixture

is respectively 29.6% and 1.1% lower than that of the R227EA and R600 pure fluids. In this sense, working with mixture fluid leads to lower capital cost rates entailing a smaller physical scale of ORC plant compared to pure fluids. Evaporator pinch point and superheat temperatures

Fig. 5 Non dominated solutions obtained by the OPSSAJ, MOPSO and NSGA-II for ORC running with R600 refrigerant

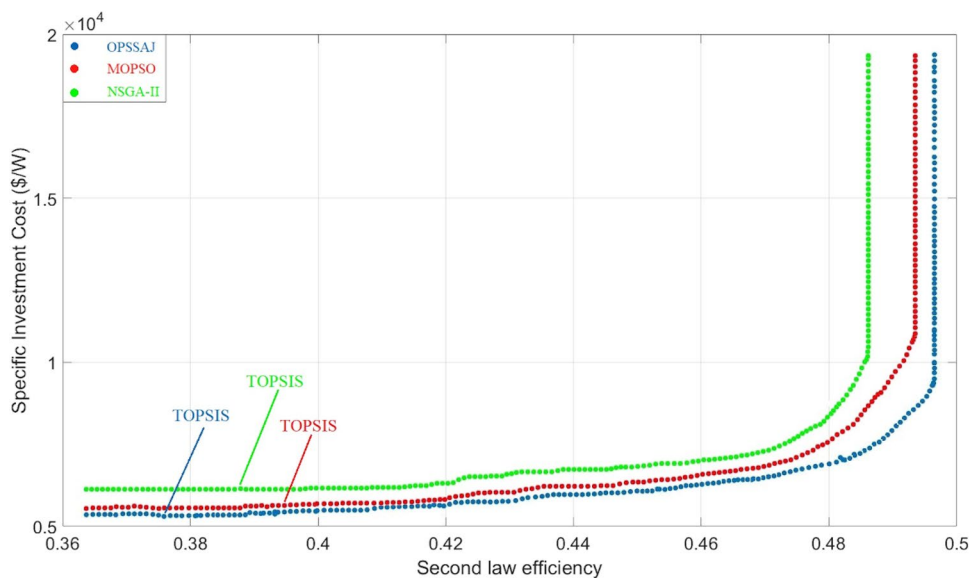
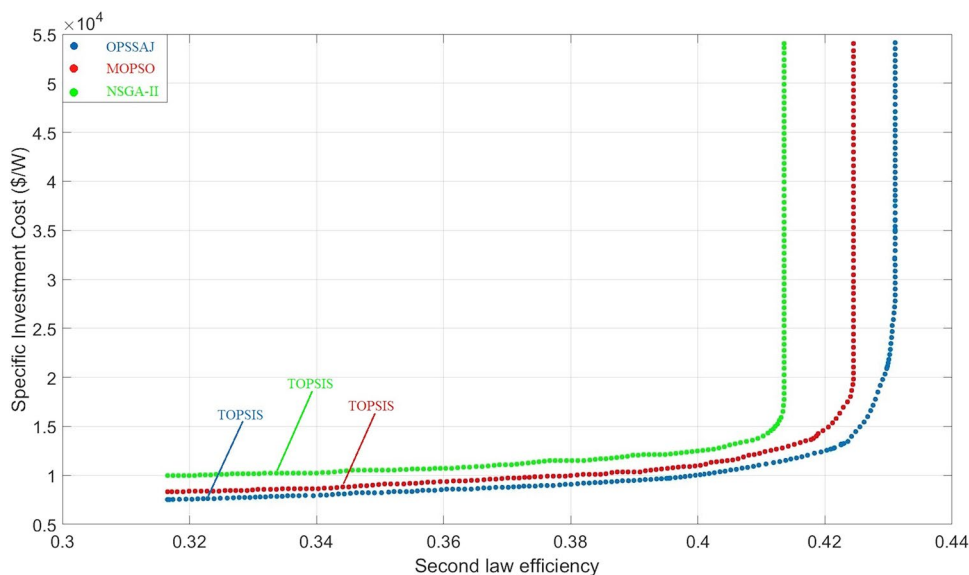


Fig. 6 Pareto optimal solutions obtained by OPSSAJ, MOPSO and NSGA-II for ORC operating with R227EA working fluid



are correspondingly 8.0 and 15.0 K when second law efficiency is maximized. Besides, the optimum condenser temperature of the mixture fluid is obtained 303.150 K, which is its minimum allowable value, in this optimization case. Based on the Carnot principle of thermodynamics higher temperature difference between evaporator and condenser yields better exergy efficiency. Increasing evaporator temperatures results in an increased enthalpy difference in the expander. This increase causes a reduction in mass flow rates of the mixture fluid. The effect of variations in enthalpy difference and mass flow rate entails an increase in the net power output of the cycle. It is noted that the mass fraction of R227EA in the mixture is found respectively 0.590 when exergetic efficiency is maximized and 0.20 when SIC is minimized. Tables 4 and 5

respectively report the single-objective optimization results of the ORC system for the NSGA-II and MOPSO algorithms. By looking at the results, it is realized that OPSSAJ outperformed the other two competing algorithms for the single-objective optimization task.

Figure 5 shows the Pareto solutions found by the NSGA-II [58], MOPSO [59] and OPSSAJ algorithms for the base ORC system running with R600 working fluid. Each sample solution on the Pareto curve represents a trade-off and no solution is superior to the other on the frontier. It is seen that the final solution selected by TOPSIS from the alternative sample points on the curve is much closer to minimum SIC rather than maximum second law efficiency. Moreover, the pareto curve generated by the OPSSAJ algorithm has more desirable solutions compared to that of

Fig. 7 Pareto solutions for the ORC cycle employing R227EA/R600 refrigerant mixture

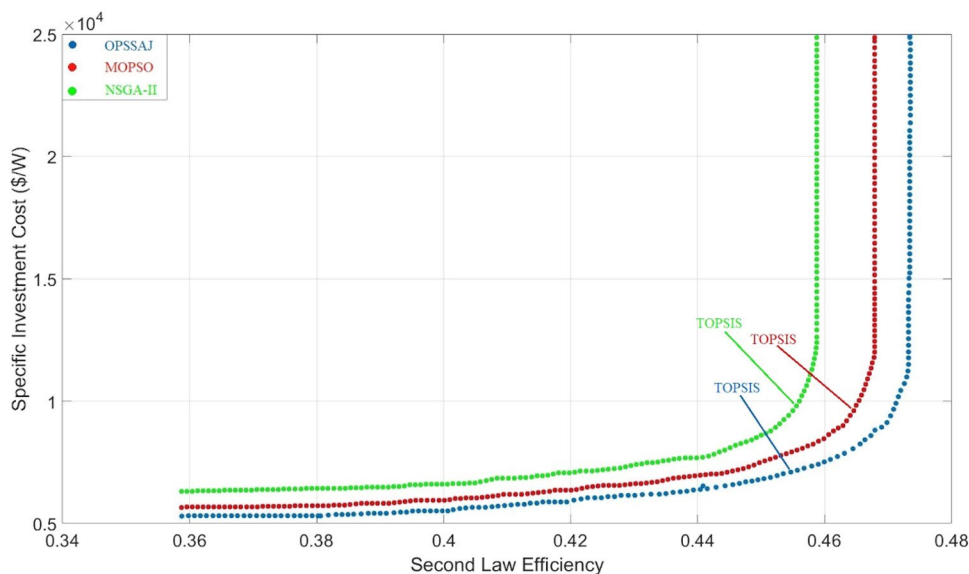


Table 6 Optimal values of cycle parameters and conflicting design objectives obtained with the OPSSAJ algorithm

Designvariables	R227EA	R600	R227EA/R600
Condenser temperature (K)	303.151	303.150	303.150
Evaporation pressure or temperature (MPa or K)	2.289 (MPa)	1.673 (MPa)	373.149 (K)
Superheat temperature (K)	29.999	0.0608	0.0002
Evaporator pinch temperature (K)	19.993	14.856	8.000
Condenser pinch temperature (K)	13.341	14.989	5.000
Condenser outer tube diameter (m)	0.015	0.015	0.015
Condenser shell diameter(m)	0.250	0.281	0.270
Condenser baffles spacing (m)	0.237	0.168	0.499
Condenser number of tube passes	4	6	4
Condenser tube arrangement	Triangular	Triangular	Square
Evaporator outer tube diameter (m)	0.015	0.015	0.015
Evaporator shell diameter (m)	0.250	0.250	0.250
Evaporator baffle spacing (m)	0.015	0.150	0.150
Evaporator number of tube passes	8	8	8
Evaporator tube arrangement	Triangular	Triangular	Triangular
Refrigerant mass flow rate(kg/s)	0.799	0.282	0.250
Mass fraction (R227EA/R600)	–	–	0.212
Second law efficiency	0.3228	0.3758	0.4491
Specific investment cost (\$/W)	7636.917	5499.612	7025.801

NSGA-II and MOPSO optimizers. Referring to the final optimum solution on the Pareto curve constructed by OPSSAJ and obtained by TOPSIS, optimal values of exergetic efficiency and Specific Investment Cost are respectively 0.3758 and 5499.612 (\$/W). Optimal solutions attained by TOPSIS for SIC and exergetic efficiency are correspondingly 2.7% higher and 24.2% lower compared to the optimum ORC system running with single R600 refrigerant when design objectives are separately optimized. Figure 6 depicts the distribution of the non-dominated solutions on the Pareto curve for ORC running with R227EA. TOPSIS

results of ORC for OPSSAJ with R227EA for second law efficiency and SIC are respectively 0.3228 and 7636.917 (\$/W). The optimization results indicate that SIC and exergetic efficiency rates are correspondingly increased by 1.4% and 2.1% compared to that of the optimum solutions obtained from single-objective optimization. Furthermore, it is seen that the Pareto curve constructed by OPSSAJ has better solutions compared to that of NSGA-II and MOPSO algorithms. Figure 7 shows the non-dominated solutions of the conflicting objectives residing on the Pareto curve for the ORC system with the R227EA/R600 mixture refrigerant.

Table 7 Optimal values of cycle parameters and conflicting design objectives obtained with the NSGA-II algorithm

Designvariables	R227EA	R600	R227EA/R600
Condense rtemperature (K)	303.184	303.182	303.183
Evaporation pressure or temperature (MPaorK)	2.429 (MPa)	1.757 (MPa)	373.147 (K)
Superheat temperature (K)	25.179	0.2786	0.0018
Evaporator pinch temperature (K)	17.460	14.752	8.000
Condenser pinch temperature (K)	15.863	14.416	5.000
Condenser outer tube diameter (m)	0.015	0.015	0.015
Condenser shell diameter (m)	0.250	0.296	0.250
Condenser baffles spacing (m)	0.289	0.170	0.499
Condenser number of tube passes	6	8	6
Condenser tube arrangement	Triangular	Triangular	Square
Evaporator outer tube diameter (m)	0.015	0.015	0.015
Evaporator shell diameter(m)	0.250	0.250	0.250
Evaporator baffle spacing (m)	0.015	0.150	0.150
Evaporator number of tube passes	4	4	4
Evaporator tube arrangement	Square	Triangular	Square
Refrigerant mass flow rate(kg/s)	0.772	0.296	0.250
Mass fraction (R227EA/R600)	–	–	0.247
Second law efficiency	0.3347	0.3872	0.4578
Specific investment cost (\$/W)	10,131.258	5764.107	9947.876

Table 8 Optimal values of cycle parameters and conflicting design objectives obtained with the MOPSO algorithm

Designvariables	R227EA	R600	R227EA/R600
Condenser temperature (K)	303.279	303.596	303.542
Evaporation pressure or temperature (MPaorK)	2.345 (MPa)	1.756 (MPa)	373.137 (K)
Superheat temperature (K)	27.056	0.2569	0.0015
Evaporator pinch temperature (K)	17.462	14.108	8.000
Condenser pinch temperature (K)	15.429	14.587	5.000
Condenser outer tube diameter (m)	0.015	0.015	0.015
Condenser shell diameter (m)	0.250	0.252	0.286
Condenser baffle sspacing (m)	0.285	0.176	0.499
Condenser number of tube passes	6	8	8
Condenser tube arrangement	Triangular	Triangular	Square
Evaporator outer tube diameter (m)	0.015	0.015	0.015
Evaporator shell diameter (m)	0.250	0.250	0.250
Evaporator baffle spacing (m)	0.015	0.150	0.150
Evaporator number of tube passes	6	6	6
Evaporator tube arrangement	Triangular	Square	Triangular
Refrigerant mass flow rate (kg/s)	0.771	0.283	0.229
Mass fraction (R227EA/R600)	–	–	0.235
Second law efficiency	0.3446	0.3972	0.4672
Specific investment cost (\$/W)	8149.547	5551.028	9646.097

TOPSIS results of OPSSAJ for the ORC cycle employing the refrigerant mixture are 0.4534 for exergetic efficiency and 7031.801 (\$/W) which are respectively 32.7% higher and 4.1% lower than those of the solutions obtained for the case of single-objective optimization. It is also realized that the OPSSAJ Pareto curve consists of more desirable solutions compared to that of NSGA-II and MOPSO

algorithms. Table 6 reports the TOPSIS results of OPSSAJ along with their corresponding design variables for the ORC system operating with different refrigerants. Tables 7 and 8 respectively report the TOPSIS results of NSGA-II and MOPSO algorithms for the ORC system. The results show that TOPSIS solution of OPSSAJ is more desirable than that of the NSGA-II and MOPSO algorithms.

6 Conclusion

This study proposed a hybrid Oppositional Salp Swarm—Jaya optimization (OPSSAJ) algorithm for solving multi objective design optimization of an Organic Rankine Cycle operating with a binary mixture composed of R227EA/R600. The proposed hybridization procedure focuses on enhancing the diversity of the population and maintaining a plausible balance between the diversification and intensification which are two important phases of any optimization algorithm. SSA is a powerful optimizer having plenty of applications in the literature. However, major drawback of SSA is too much exploitation caused by the chain movement of the salps on the fertile areas. This deficiency is conquered by using opposite point of the current solution which can diversify the search space and enables to reach unknown paths of the solution domain. Jaya algorithm is an effective metaheuristic optimizer having a superior capability in local minima avoidance thanks to its dexterous manipulation schemes. It is aimed to compensate for the algorithm-specific disadvantages of SSA by hybridizing it with Jaya and Oppositional-based algorithms, both of which having an exceptional exploration capability. The proposed hybrid OPSSAJ is applied to a suite of twenty-four test functions comprised of unimodal and multimodal problems in order to benchmark its optimization efficiency on multidimensional test problems. Statistical analysis have been performed and numerical results obtained from OPSSAJ have been compared with those acquired by some of recently developed metaheuristic optimizers. The proposed algorithm outperforms the contender algorithms with regards to the solution accuracy and robustness in most of the cases. Integrating the basic mutation scheme of JAYA with QOBL method significantly boosts the optimization performance of the hybrid method such that the proposed algorithm obtains the best optimum results in 20 out of 24 test functions and shows the best predictive accuracy when ranking-point based assessment is performed. As a real-world case application, OPSSAJ is evaluated on solving a multi objective design of an Organic Rankine Cycle. Optimum cycle design parameters have been acquired by the proposed algorithm considering single and multi objective optimization cases. Multi-objective optimization of ORC system results shows that using a binary mixture instead of a single refrigerant greatly enhances the second law efficiency of the cycle. However, using binary mixture increases the SIC of the system compared to the system running with R600 refrigerant, based on the numerical outcomes of the TOPSIS decision-maker. During sensitivity analysis, it is also interesting to see that the second law efficiency of the thermal cycle is increased then enters a declining zone as the mass fraction of R227EA in the

refrigerant mixture increases. This behavior can be attributed to the intrinsic thermal characteristic of R227EA refrigerant. Furthermore, a sensitivity analysis have been performed based on the non-dominated solution of the Pareto curve chosen by TOPSIS to observe the influences of the varying cycle parameters on the problem objectives.

Compliance with ethical standards

Conflicts of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Ethics approval This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Faramarzi A, Afshar MH (2012) Application of cellular automata to size and topology optimization of truss structures. *Sci Iran* 19:373–380
2. Haghparast P, Sorin MV, Richard MA, Nesreddine H (2019) Analysis and design optimization of an ejector integrated into an Organic Rankine Cycle. *Appl Therm Eng* 159:113979
3. Chen C, Huang R, Luo X, Chen J, Yang Z, Chen Y (2019) Conceptual design and thermodynamic optimization of a novel composition tunable zeotropic Organic Rankine Cycle. *Energy Procedia* 158:2019–2024
4. Meroni A, Andreasen JG, Persico G, Haglind F (2018) Optimization of Organic Rankine Cycle power systems considering multistage axial turbine design. *Appl Energy* 209:339–354
5. Scardigno S, Fanelli E, Viggiano A, Braccio G, Magi V (2015) A genetic optimization of a hybrid organic Rankine plant for solar and low-grade energy sources. *Energy* 91:807–815
6. Pantaleo AM, Camporeale SM, Sorrentino A, Miliozzi A, Shah N, Markides CN (2020) Hybrid solar-biomass combined Brayton/organic Rankine-cycle plants integrated with thermal storage: Techno-economic feasibility in selected Mediterranean areas. *Renew Energy* 3:2913–2931
7. Li J, Pei G, Li Y, Ji J (2010) Novel design and simulation of a hybrid solar electricity system with Organic Rankine Cycle and PV cells. *Int J Low-Carbon Tec* 5:223–230
8. Feng H, Chen W, Chen L, Tang W (2020) Power and efficiency optimizations of an irreversible regenerative Organic Rankine Cycle. *Energy Convers Manag* 220:113079
9. Baldasso E, Mondejar ME, Andersen JG, Ronnenfelt KAT, Nielsen BO, Haglind F (2020) Design of Organic Rankine Cycle power

- systems for maritime applications accounting for engine back-pressure effects. *Appl Therm Eng* 178:155527
10. Liu Z, Yang X, Jia W, Li H, Hooman K, Yang X (2020) Thermodynamic study on a combined heat and compressed air energy storage system with a dual-pressure Organic Rankine Cycle. *Energy Convers Manag* 221:113141
 11. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM (2017) Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems. *Adv Eng Softw* 114:163–191
 12. Rao RV (2016) Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *Int J Ind Eng Comput* 7:19–34
 13. Tizhoosh HR (2005) Opposition-based learning: a new scheme for machine intelligence. In: International conference on computational intelligence for modelling, control and automation, and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06), 28–30 November 2005, Vienna, Austria, IEEE
 14. El-Fergany AA (2018) Extracting optimal parameters of PEM fuel cells using Salp Swarm Optimizer. *Renew Energy* 119:641–648
 15. Faris H, Mafarja MM, Heidari AA, Aljarah I, Al-Zoubi AM, Mirjalili S, Fujita H (2018) An efficient binary Salp Swarm algorithm with crossover scheme for feature selection problems. *Knowl Based Syst* 154:43–67
 16. Gholami K, Parvaneh MH (2019) A mutated Salp Swarm algorithm for optimum allocation of active and reactive power sources in radial distribution systems. *Appl Soft Comput* 85:105833
 17. Xu S, Wang Y, Wang Z (2019) Parameter estimation of proton exchange membrane fuel cells using eagle strategy based on Jaya algorithm and Nelder-Mead simplex method. *Energy* 173:457–467
 18. Rao RV, Saroj A (2017) Constrained economic optimization of shell and tube heat exchangers using elitist-Jaya algorithm. *Energy* 128:785–800
 19. Rao RV, Rai DP (2017) Optimisation of welding processes using quasi-oppositional-based Jaya algorithm. *J Exp Theor Artif Intell* 29:1099–1117
 20. Kang L, Chen RS, Cao W, Chen YC (2020) Non-inertial opposition-based particle swarm optimization and its theoretical analysis for deep learning applications. *Appl Soft Comput* 88:106038
 21. Sarkhel R, Das N, Saha AK, Nasipuri M (2018) An improved Harmony Search algorithm embedded with a novel piecewise opposition based learning algorithm. *Eng Appl Artif Intell* 67:317–330
 22. Abd-Elaziz M, Oliva D (2018) Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm. *Eng Convers Manage* 171:1843–1859
 23. Mahapatra S, Raj S, Krisna SM (2020) Optimal TCSC location for reactive power optimization using oppositional Salp Swarm algorithm. In: Sharma R, Mishra M, Nayak J, Naik B, Pelusi D (eds) *Innovation in electrical power engineering, communication, and computing technology*, vol 630. Springer, Signapore
 24. Kang F, Li J, Dai J (2019) Prediction of long-term temperature effect in structural health monitoring of concrete dams using support vector machines with Jaya optimizer and Salp Swarm algorithms. *Adv Eng Softw* 131:60–76
 25. Malisia AR (2008) Improving the exploration ability of ant-based algorithms. *Oppositional concepts in computational intelligence*. Springer, Heidelberg, pp 121–142
 26. Mandal B, Roy PK (2013) Optimal reactive power dispatch using quasi-oppositional teaching-learning based optimization. *Int J Elec Power* 53:123–134
 27. Hazra S, Roy PK (2019) Quasi-oppositional chemical reaction optimization for combined economic emission dispatch in power system considering wind power uncertainties. *Renew Energy Focus* 31:45–62
 28. Xi H, Li MJ, He YL, Zhang YW (2017) Economical evaluation and optimization of Organic Rankine Cycle with mixture working fluids using R245fa as flame retardant. *Appl Therm Eng* 113:1056–1070
 29. Sadeghi M, Nemati A, Ghavimi A, Yari M (2016) Thermodynamic analysis and multi-objective optimization of various ORC (Organic Rankine Cycle) configurations using zeotropic mixtures. *Energy* 109:791–802
 30. Feng Y, Hung TC, Zhang Y, Li B, Yang J, Shi Y (2015) Performance comparison of low-grade ORCs (Organic Rankine Cycles) using R245fa, pentane and their mixtures based on the thermo-economic multi-objective optimization and decision makings. *Energy* 93:2018–2029
 31. Hwang CL, Yoon K (1981) *Methods for multiple attribute decision making*. Springer, Heidelberg, pp 58–181
 32. Turgut MS, Turgut OE (2020) Global best-guided oppositional algorithm for solving multidimensional optimization problems. *Eng Comput* 36:43–73
 33. Seif Z, Ahmadi MB (2015) An opposition-based algorithm for function optimization. *Eng Appl Artif Intell* 37:293–306
 34. Rahnamayan S, Tizhoosh HR, Salama MMA (2007) Quasi-oppositional Differential Evolution. In: 2007 Congress on evolutionary computation, p 2229–2236. IEEE
 35. Ergezer M, Simon D, Du D (2009) Oppositional biogeography-based optimization. In: 2009 International conference on systems, man and cybernetics, p 1009–1014. IEEE
 36. Xu Q, Wang L, He B, Wang N (2011) Modified opposition-based differential evolution for function optimization. *J Comput Inf Syst* 7:1582–1591
 37. Mahdavi S, Rahnamayan S, Deb K (2018) Opposition based learning: a literature review. *Swarm Evol Comput* 39:1–23
 38. Kaveh A, Mahdavi VR (2014) Colliding bodies optimization: a novel meta-heuristic method. *Comput Struct* 139:18–27
 39. Yang XS, Deb S (2009) Cuckoo Search via Levy flights. In: *World congress on nature & biologically inspired computing (NaBIC 2009)*, Publications, p 210–214. IEEE
 40. Pan W-T (2012) A new Fruit Fly Optimization Algorithm: taking the financial distress model as an example. *Knowl Based Syst* 26:69–74
 41. Kiran MS (2015) TSA: Tree-seed algorithm for continuous optimization. *Expert Syst Appl* 42:6686–6698
 42. Dhiman G, Kumar V (2017) Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. *Adv Eng Softw* 114:48–70
 43. Dhiman G, Kumar V (2018) Emperor penguin optimizer: A bio-inspired algorithm for engineering problems. *Knowl Based Syst* 159:20–50
 44. Kaveh A, Majid IG (2017) Vibrating particles system algorithm for truss optimization with multiple natural frequency constraints. *Acta Mech* 228:307–322
 45. Arora S, Singh S (2019) Butterfly optimization algorithm: a novel approach for global optimization. *Soft Comput* 27:715–734
 46. Askarzadeh A (2016) A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Comput Struct* 169:1–12
 47. Li YR, Du MT, Wu CM, Wu SY, Liu C (2014) Potential of Organic Rankine Cycle using zeotropic mixtures working fluids for waste heat recovery. *Energy* 77:509–519
 48. Gnielinski V (1976) New equations for heat and mass transfer in turbulent pipe and channel flow. *Int Chem Eng* 16:359–367
 49. Thome JR (1996) Boiling of new refrigerants: a state-of-the-art review. *Int J Refrig* 19:435–457
 50. Gungor W, Winterton R (1986) A general correlation for flow boiling in tubes and annuli. *Int J Heat Mass Tran* 29:351–358
 51. Florides GA, Kalogirou SA, Tassou SA, Wrobel LC (2003) Design and construction of LiBr–water absorption machine. *Energy Convers Manag* 44:2483–2508

52. Shah MM (1979) A general correlation for heat transfer during film condensation inside tubes. *Int J Heat Mass Tran* 22:547–556
53. Mignard D (2014) Correlating the chemical engineering plant cost index with macro-economic indicators. *Chem Eng Res Des* 92:285–294
54. Turton R, Bailie RC, Whiting WB, Shaeiwitz JA, Bhattacharyya D (2012) *Analysis, synthesis and design of chemical processes*, 4th edn. Prentice-Hall, London
55. Malisia AR (2008) Improving the exploration ability of ant-based algorithms. *Oppositional concepts in computational intelligence*. Springer, Heidelberg, pp 121–142
56. Jenkins S (2020) 2020 CEPCI Updates October (Preliminary) and September (Final). *Chemical engineering essential for the CPI professional*. <https://www.chemengonline.com/2020-cepci-updates-october-prelim-and-sept-final>. Accessed 10 Nov 2020
57. Schuster A, Karellas S, Kakaras E, Spliethoff H (2009) Energetic and economic investigation of Organic Rankine Cycle applications. *Appl Therm Eng* 29:1809–1817
58. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
59. Coello Coello CA, Lechuga MS (2002) MOPSO: a proposal for multiple objective particle swarm optimization. In: *Proceedings of the 2002 Congress on Evolutionary Computation CEC'02*.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.