Received: March 3, 2022. Revised: April 27, 2022.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

The Exploration of Wild Horse Optimization in Reliability Redundancy Allocation Problems

Najla Akram AL-Saati[1]*

[1]*Software Department, College of Computer Science and Mathematics, University of Mosul, Iraq.*
*Corresponding author's Email: dr.najla–alsaati@uomosul.edu.iq

Abstract: In this work, the wild horse optimization (WHO) algorithm, known for its ease of use, efficiency, and fast convergence, is explored in solving the reliability redundancy allocation problem (RRAP) for series-parallel systems. This problem has as of late caught the attention of researchers in this area, especially in today's rapidly growing field of artificial intelligence. The NP-hard RRAP problem deals with maximizing of reliability under certain constraints. This work uses WHO algorithm to maximize the overall system reliability by determining how many redundant components are to be used along with their reliabilities in each subsystem, such reliability is constrained by cost, volume, and weight. Testing is carried out to show the effectiveness of this algorithm using four known numerical examples, results are to be compared with simplified swarm algorithm (SSO), attraction-repulsion imperialist competitive algorithm (AR-ICA), hybrid salp swarm algorithm and teaching-learning based optimization (HSS-TLBO), particle swarm optimization (PSO), and gradient based optimization (GBO). Computational results show that WHO was able to find better feasible near-optimal solutions effectively and efficiently in terms of population size and number of iterations.

Keywords: Swarm intelligence, Wild horse optimization, Reliability, Series-parallel systems, Optimization.

1. Introduction

The optimization of system reliability is an important type of optimization problems, it covers a vital topic in real-life applications such as systems of telecommunication, transformation, space exploration, textile manufacturing, and airport security systems [1]. Many researchers considered the improvement of the reliability for systems or products to compete in business enterprises. In series-parallel systems, individual components largely affect the probability of a failure-free operation. Such systems are supposed to operate and be available for the extreme possible period to make the most of the overall profit. Still, failures are practically unavoidable phenomena in machine-driven systems, that is why, the analysis of reliability has grown to be a very important topic [2].

This problem is identified as NP-hard problems. Commonly, there two main approaches to gain better reliability for a system; either by increasing the reliability of system components, or by using redundancy in subsystems. In the first approach, system reliability can be reasonably enhanced, yet this enhancement may be unachievable despite using the most reliable parts. Employing the second methodology, on the other hand, involves selecting the ideal component combination and redundancy, that is, while enhancing the reliability, features such as cost, weight, and volume will increase too. No matter which approach is chosen, there exists some feasible techniques to improve the reliability such as unifying both approaches or reassigning substitutable elements. Such problems that maximize system reliability via redundancy and component reliability selections are called reliability redundancy allocation problems (RRAP) [3].

RRAP problems implicate the definition of reliability objectives for components/subsystems to fulfil the constraints for consuming resources such as cost. Thus, RRAPs are emerging as progressively significant mechanisms in the early stages of planning, designing, and systems control.

Depending on the approach used to improve system reliability, problems of designing reliability are divided into integer and mixed integer problems. Using integer reliability problems (RAPs), the allocation of component redundancy is to be determined while the component reliabilities are at hand. Whereas in mixed-integer reliability problems (RRAPs), the quantity of redundant components and their corresponding reliabilities are to be determined simultaneously.

It has been stated that the problems of nonlinear mixed-integer programming are further complicated than the basic redundancy allocation problems. It is also found that exact optimal solutions to RRAP cannot be not easily obtained due to high complexity of computation [2]. It is very complicated to reduce the system cost and improve the reliability at the same time, this is a result of the non-stochastic uncertain and conflicting factors of the problem [4].

That is why researchers in this field became devoted to use intelligent algorithms that can discover near optimal solutions in a fast and efficient manner, rather than the traditional exact solution methods that become unfeasible with large search spaces. Swarm intelligent algorithms are being widely employed, these population-based search algorithms have proved its ability to find good solutions in a relatively short time or less iterations. One of the lately introduced swarms is the wild horse optimization (WHO) symbolizing the behaviour of wild horses with their known decent mating feature prohibiting family members from mating with each other, thus providing variety and diversity in producing new solutions.

The increasing demand for high reliable systems necessitates the study of reliability optimization. Many researchers have attempted to solve this problem using various methods and techniques. Starting at 2008, Moghaddam et. al., used genetic algorithms to tackle the RRAP and proved its efficiency [5]. A PSO grounded on Gaussian distribution and chaotic sequence was employed in 2009 by Dos and Coelho in solving the RRAPs [6]. In 2011, Yeh and Hsieh suggested using artificial Bee colony algorithm, as this algorithm possess benefits of memory, multi-character local search and a method of improving solutions by being able to discover new optimal solution [7]. But their results were found to be infeasible in 2015 by Huang et. al.,[8]. In 2013, Garg and Sharma proposed a penalty-based PSO aiming to search for the optimal solution with nonlinear constraints for a pharmaceutical plant [9]. In the same year, Afonso et. al, used an idea of attraction and repulsion with the imperialist competitive algorithm (ICA) for solving four different designs of the RRAPs, their approach (AR-ICA) showed an improved search for the problem [10]. A hybrid procedure of advanced genetic algorithm and PSO to find best solutions was adopted by Sahoo et. al., in 2014 [11]. The simplified swarm optimization (SSO) was suggested by Huang and Yeh for solving the RRAP and improve computational proficiency in 2015 [8]. In 2017, Valaei and Behnamian considered the redundancy allocation problem and the sequencing of standby elements for the 1-out-of-N:G heterogeneous cold standby systems, concurrently [12]. By 2020, Taghiyeh et. al., proposed two diverse approaches, using Zadeh's extension principle and modification of fuzzy parametric programming, to consider uncertainty in the overspeed protection system [13]. Lately in 2021, Jaleel and Abd presented a modified PSO approach with analysis to evaluate reliability and estimate optimal locations and capacities of the DGs units using multi-objective functions for reducing power-loss and improving voltage profile [14]. Also in 2021, Ashraf et. Al., used A parameter free penalty gradient based optimizer (GBO) to provide feasible solutions for the RRAPs of the pharmaceutical plant [15]. Most recently in 2022, Kundu et. al., used a hybrid salp swarm algorithm (HSS-TLBO) based on teaching-learning based optimization (TLBO) for RRAPs, aiming to merge the SSA ability of global search with the fast converging of (TLBO) to maximize reliability [16].

In this work, the wild horse optimization is considered for solving the RRAP problems aiming to maximize the overall system reliability under certain constraints. The primary focus is the adjustment of the number of component and their reliabilities to meet the constraints of resource utilization denoted by system cost, weight, and volume. To test the efficiency of the algorithm, four numerical examples are used.

2. The reliability redundancy allocation problem (RRAP)

In general, a system can be organized into any of the following categories: Series, parallel, series–parallel, and complex systems. Components or subsystems connected in series means that when one of them fail, then the whole system fails. While in parallel configurations, all the components have to fail in order for the whole system to fail. As for the complex/bridge system, it has an interconnection component that transmits the excess capacity to other components or subsystems. Every component is made up of (n) parallel components.

The reliability optimization problem has three distinctive forms: redundancy allocation, reliability allocation, and reliability redundancy allocation problems (RRAP). The approach for solving each of them differs depending on the problem assumptions and type. Redundancy allocation problems denote systems of discrete component types having fixed reliability, cost and weight. The problem is selecting optimal collection of components types that meet the constraints to attain the objective function, these problems are used with series-parallel systems. The problem of reliability allocation, on the other hand, has a fixed system structure and continuous decision variables for component reliability. The cost, weight, and volume are functions of component reliability, using more components to increase system reliability can increase cost, and possibly weight and volume [17].

As for the reliability redundancy allocation problem, it is considered to have the utmost general problem representation. The system and subsystems can be in any arrangement and all subsystems have (n_i) components and (r_i) reliabilities. The task is to allocate the optimal redundancy and reliability to the components of these subsystems with the purpose of optimizing the overall system reliability.

In dealing with RRAPs, the objective function is the maximization of the total system reliability (R_s), this is done by resolving the number of components and their reliability for individual subsystems simultaneously under several constraints [18].

In series systems, the overall system reliability can be computed as in Eq. (1), whereas in parallel systems, system reliability is computed as in Eq. (2).

$$R_s = \prod_{i=1}^{m} R_i \quad (1)$$

$$R_s = 1 - \prod_{i=1}^{m}(1 - R_i) \quad (2)$$

Where (R_i) is the reliability for (i^{th}) subsystem defined in Eq. (3) by using (n_i) as the quantity of redundant components for that subsystem and (r_i) is the individual component reliability.

$$R_i = 1 - (1 - r_i)^{n_i} \quad (3)$$

RRAP seeks to enhance the whole system reliability in a definite controlled setting, depending on the constraints of cost, weight, and volume of the system. The cost function (f_{cost}) relies on the count of the subsystems (m) and their reliabilities along with the components count in each subsystem (n_i) as indicated in Eq. (4). The value (1000) is supposed to be the mean time between failures, (α_i) and (β_i) are function parameters for cost.

The function controlling the weight of the system (f_{weight}) is subject to the count of subsystems and the count of components in each subsystem, (w_i) is the parameter of weight function. Its definition is given in Eq. (5).

As for the volume function (f_{volume}), it is similarly established on the count of subsystems and the count of components in each one using (v_i) as the volume function parameter as shown in Eq. (6).

$$f_{cost} = \sum_{i=1}^{m} f_{cost_i} = \sum_{i=1}^{m} \alpha_i \left(\frac{-1000}{ln\, r_i}\right)^{\beta_i} \left(n_i + e^{\frac{n_i}{4}}\right) \quad (4)$$

$$f_{weight} = \sum_{i=1}^{m} f_{weight_i} = \sum_{i=1}^{m} w_i . n_i . e^{\frac{n_i}{4}} \quad (5)$$

$$f_{volume} = \sum_{i=1}^{m} f_{volume_i} = \sum_{i=1}^{m} v_i . n_i^2 \quad (6)$$

In general, an RRAP can be stated as in Eq. (7), where the reliability is the objective function.

$$Max\ R_s = f(r_i, n_i) \quad (7)$$

Subject to
$f_{cost}(r_i, n_i, \alpha_i, \beta_i) \leq C$
$f_{weight}(n_i, w_i) \leq W$
$f_{volume}(n_i, v_i) \leq V$

Where *C*, *W*, and *V* signify upper boundaries for system cost, weight and volume respectively. [18]

3. Wild horse optimizer (WHO)

Wild horses are stable family groups consisting of a single stallion horse along with one or many mares and offspring. A stallion is usually a male horse used for breeding, and a mare is an adult female horse. Single groups have an adult stallion and youthful horses, and a stallion is put next to mares to give them a chance to mate. In family groups, foals typically start to graze in their first week, and as they grow older, they graze more and need less rest. They depart their groups before puberty, male horses usually join single groups to grow up and mate, while females enrol in other family groups. This parting is done to avoid mating of father with its daughter or siblings [19].

There is a noticed intergroup dominance in the groups of wild horses in dry seasons, as groups of upper ranks have access to a water resource whenever they request, while groups of lesser ranks wait for their turn. The leader of a group is responsible of the moving speed and direction. The family leader is typically the most dominant, and the remaining come in sequence of lessening dominance, the stallion

Received: March 3, 2022. Revised: April 27, 2022.

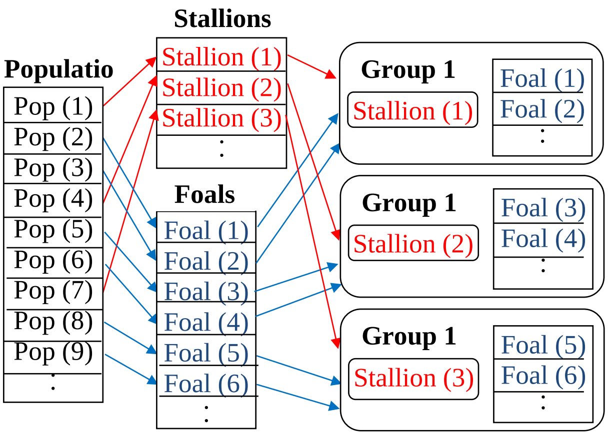

Figure. 1 Population construction process

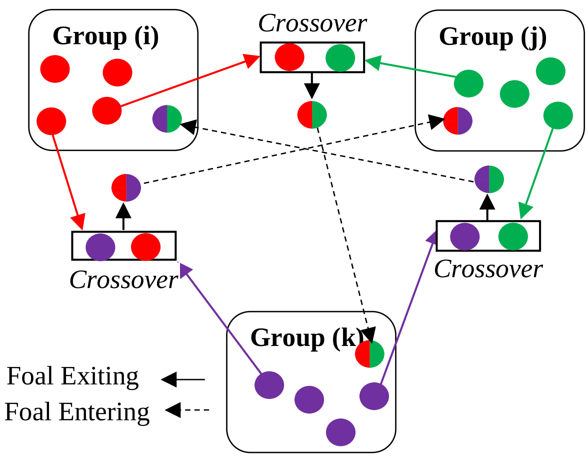

Figure. 2 Parting from groups for mating

come after the group in a short distance.

3.1 The wild horse optimization algorithm

The algorithm starts with a randomly initialized population of (N) individual divided into (S) groups, where (S) is calculated as ($N \times PS$), with (PS) being the ratio of stallions in the over-all population. The remaining foal individuals (N–S) are equally distributed amongst these groups. Fig. 1 is a sample of constructing such a population. Group leaders are chosen according to the fitness of individuals of the group, as the fittest individual becomes the leader [19].

After dividing the population comes the graze and mate operations. As already mentioned, foals graze around their group for the majority of their time. To implement this behaviour, a stallion is considered to be the midpoint of the grazing area, while other group individuals explore round it.

The grazing behaviour is simulated using Eq. (8), to resemble the movement of group members and the searching process surrounding the leader with a varied radius (VR) given in Eq. (9).

$$Xnew_{i,S}^{j} = VR \times \left(St^{j} - X_{i,S}^{j}\right) + St^{j} \tag{8}$$

$$VR = 2Z\cos(2\pi RZ) \tag{9}$$

Where ($X_{i,S}^{j}$) is the individual current location, ($Xnew_{i,S}^{j}$) is the new location of the group individual as it grazes. The location of the stallion is denoted by (St^{j}), and (R) symbolizes a random number that uniformly ranges between [−2, 2], it changes the angles of rotation for grazing (360) degrees. The (Cos) function is used to vary the radius of the movement and (Z) is calculated using Eq. (10).

$$P = \vec{R_1} < TDR; IDX = (P = 0);\ Z = R_2\ \Theta\ Indx + \vec{R_3}\ \Theta(\sim Indx) \tag{10}$$

Here (P) is a vector of 0s and 1s of size equal to the problem dimension, ($\vec{R}_1$), ($\vec{R}_3$) denote random vectors that are distributed uniformly between [0, 1]. The (R_2) signifies a random number uniformly distributed in the range [0, 1], the values of ($Indx$) are indexes of ($\vec{R}_1$) satisfying the condition ($P=0$). While (TDR) refers to a parameter that starts at (1) and decreases to end at (0) through the execution (Eq. (11)), given that (*iter*) denotes current iteration, and (*max-itr*) is the maximum iterations [20].

$$TDR = 1 - iter \times \left(\frac{1}{max-itr}\right) \tag{11}$$

Horses possess a unique behaviour opposed to other sorts of animals; it is, as stated earlier, the practice of parting foals from the group in order to mate when they reach puberty. To implement this action, a foal (assumed to be a male) leaves group (i), and a foal (assumed to be a female) leaves group (j), they can mate (crossover) as they are not related to each other. But their offspring must leave them and joins a distinct group, assumed (k). This series of parting, mating and reproduction is repetitive for all groups as illustrated in Fig. 2. This process is formulated in Eq. (12).

$$X_{S,K}^{p} = Crossover\left(X_{S,i}^{q}, X_{S,j}^{Z}\right) \quad for\ i \neq j \neq k, p = q = end. Crossover{:}\ mean \tag{12}$$

Where ($X_{S,K}^{p}$) signifies the location for horse (p) in group (k) that leaves the group granting its location to an offspring whose parents had parted group (i) and (j). The ($X_{S,i}^{q}$) is the position of foal (q) parting group (i) to mate with horse (z) that has parted from group (j).

As for leading the group, the leader has to guide

Received: March 3, 2022. Revised: April 27, 2022.

the individuals of his group to an appropriate area near a "water hole", other groups walk likewise to the same water hole. Here, leaders have to compete for dominating that water hole, other groups can't use that water unless the domination group leave it. Eq. (13) symbolizes this attitude mathematically [20].

$$Stnew_{S_i} = \begin{Bmatrix} VR \times (WH - St_{S_i}) + WH \ \ if \ Rn > 0.5 \\ VR \times (WH - St_{S_i}) - WH \ \ if \ Rn \leq 0.5 \end{Bmatrix} \quad (13)$$

Where ($Stnew_{S_i}$) is the new location of the leader of group (i), (WH) is the location of the water hole, and (St_{S_i}) is the previous known location for the leader of group (i). Here (Rn) is a uniform random number ranging between [-2, 2]. Updating the location of the new stallion location is done relatively to the best location.

In the initialization, leaders are chosen in random, later in the algorithm, they are chosen according to fitness. when one of the individuals is fitter than the group leader, their locations are exchanged Eq. (14).

$$St_{S_i} = \begin{Bmatrix} X_{S,i} \ \ if \ fitness(X_{S,i}) < fitness(St_{S_i}) \\ St_{S_i} \ \ if \ fitness(X_{S,i}) > fitness(St_{S_i}) \end{Bmatrix} \quad (14)$$

Where (St_{S_i}) is the location for leader of group (i), and ($X_{S,i}$) denotes individuals in the group.

Wild horse optimization has some inherent characteristics when dealing with optimization problems, such as in parting and mating with individuals from other groups, there exists a higher probability for overcoming local optima. The calculation of random horse movements allows for improved diversity in populations. It is also noticed that through optimization, leaders move to the locations of best horses which are the important zones of the searching space. Value of the best leader is stored for each iteration and is then measured up to the best leader obtained lately [19].

Fig. 3 gives the flowchart for the wild horse algorithm optimization.

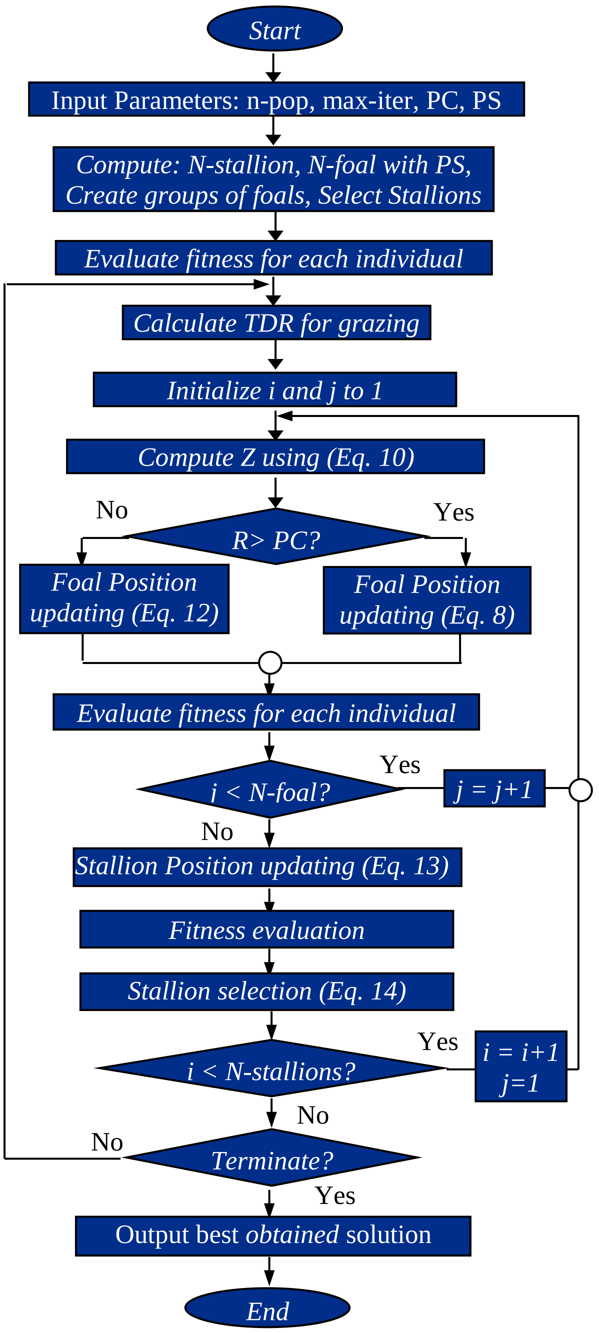

Figure. 3 Flowchart of the wild horse optimization

4. Test examples and parameter setting

Assessing the efficiency and capability of the employed algorithm requires the exploration of a number of numerical problems chosen from the literature as benchmark problems. Results are to be compared with previously found ones. Table 1 gives the parameter settings for WHO used in solving the RRAP problems.

Testing involves exploring four problems: Series-parallel system (P1), complex/bridge system (P2), the overspeed protection system (P3) and the pharmaceutical plant. These problems are discussed in next paragraphs.

Problem (P1) symbolizes the series-parallel

Table 1. Parameter setting for WHO used in testing

Parameter	Setting
PS	0.2
PC	0.13
Crossover	Mean
Population Size	25 - 100
No. of Iterations	75 - 250

Received: March 3, 2022. Revised: April 27, 2022.

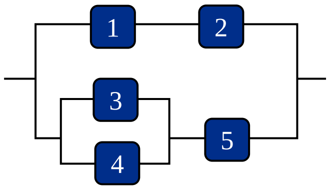

Figure. 4 The series-parallel system (P1)

Table 2. Data used in problem P1

Subsys i	$\alpha_i \times 10^5$	β_i	$w_i v_i^2$	w_i	V	C	W
1	2.500	1.5	2	3.5	180	175	100
2	1.450	1.5	4	4.0			
3	0.541	1.5	5	4.0			
4	0.541	1.5	8	3.5			
5	2.100	1.5	4	4.5 3.5*			

* As used by [16]

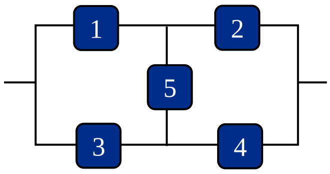

Figure. 5 The complex/bridge system (P2)

system depicted in Fig. 4; the problem is mathematically formulated as in Eq. (15). Table 2 shows the test input data [2], the (w_5) value differs according to [16] as specified in Table 2, this is to be considered in the comparisons.

$$Max\ f(r,n) = 1 - (1 - R_1R_2)(1 - (1 - R_3)(1 - R_4)R_5) \quad (15)$$

Subject to

$$f_{cost}(r,n) = \sum_{i=1}^{m} \alpha_i \left(\frac{-1000}{\ln r_i}\right)^{\beta_i} \left(n_i + e^{\frac{n_i}{4}}\right) \le C$$

$$f_{weight}(r,n) = \sum_{i=1}^{m} w_i n_i \exp(n_i/4) \le W$$

$$f_{volume}(r,n) = \sum_{i=1}^{m} w_i v_i^2 n_i^2 \le V$$

$$0 \le r_i \le 1,\ n_i \in Z^+$$

For problem P2, the complex/bridge system is illustrated in Fig. 5, the formulation of the problem is shown in Eq. (16). The input data used for the test are given in Table 3 [2].

$$Max\ f(r,n) = R_1R_2 + R_3R_4 + R_1R_4R_5 + R_2R_3R_5 - R_1R_2R_3R_4 - R_1R_2R_3R_5 - R_1R_2R_4R_5 - R_1R_3R_4R_5 - R_2R_3R_4R_5 + 2R_1R_2R_3R_4R_5 \quad (16)$$

Subject to

$f_{cost}(r,n), f_{weight}(r,n)$, and $f_{volume}(r,n)$

$0 \le r_i \le 1,\ n_i \in positive\ integer, 1 \le i \le m$

Table 3. Input data used in P2

Subsys i	$\alpha_i \times 10^5$	β_i	$w_i v_i^2$	w_i	V	C	W
1	2.33	1.5	1	7	110	175	200
2	1.450	1.5	2	8			
3	0.541	1.5	3	8			
4	8.050	1.5	4	6			
5	1.950	1.5	2	9			

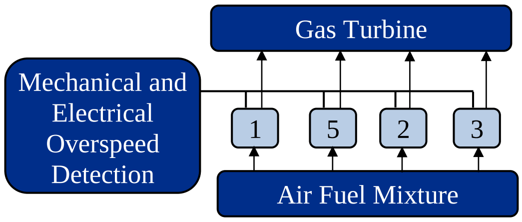

Figure. 6 The overspeed protection system (P3)

Table 4. Data used in problem P3

Subsys i	$\alpha_i \times 10^5$	β_i	v_i	w_i	V	C	W	T
1	1	1.5	1	6	250	400	500	1000
2	2.3	1.5	2	6				
3	0.3	1.5	3	8				
4	2.3	1.5	2	7				

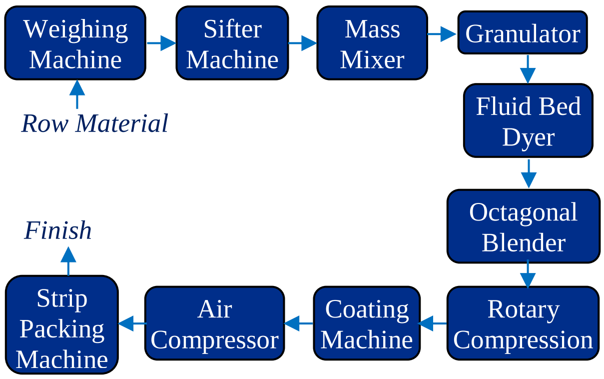

Figure. 7 The pharmaceutical plant

Table 5. Data used in problem P4

Subsys i	$\alpha_i \times 10^5$	β_i	v_i	w_i
1	0.611360	1.5	4.0	9.0
2	4.032464	1.5	5.0	7.0
3	3.578225	1.5	3.0	5.0
4	3.654303	1.5	2.0	9.0
5	1.163718	1.5	3.0	9.0
6	2.966955	1.5	4.0	10.0
7	2.045865	1.5	1.0	6.0
8	2.649522	1.5	1.0	5.0
9	1.982908	1.5	4.0	8.0
10	3.516724	1.5	4.0	6.0

V	C	W	T
289	553	483	1000

Whereas problem P3 symbolizes the "overspeed protection system" of a turbine having a time-related cost function as illustrated in Fig. 6. The control system in this problem is a 4-staged mixed series-parallel system. The problem assumes that

Received: March 3, 2022. Revised: April 27, 2022.

throughout the operation time, system components are not allowed to fail. [13] The problem is expressed in Eq. (17). The input data are in Table 4.

$$Max\ f(r,n) = \prod_{i=1}^{m}[1-(1-r_i)^{n_i}] \quad (17)$$

Subject to

$$f_{cost}(r,n) = \sum_{i=1}^{m} \alpha_i (\frac{-1000}{ln\, r_i})^{\beta_i} (n_i + e^{\frac{n_i}{4}}) \le C$$
$$f_{weight}(r,n) = \sum_{i=1}^{m} w_i n_i\ e^{\frac{n_i}{4}} \le W$$
$$f_{volume}(r,n) = \sum_{i=1}^{m} v_i n_i^2\ \le V$$
$$0.5 \le r_i \le 1-10^{-6},\ r_i \in\ R^+$$
$$1.0 \le n_i \le 10,\ n_i \in Z^+$$

Problem P4 considers a larger and more complicated problem of optimizing the reliability of a pharmaceutical plant depicted in Fig. 7. [4] The model is expressed in Eq. (18), Table 5 shows the data used in the test.

$$Max\ \ f(r,n) = \prod_{i=1}^{10}[1-(1-r_i)^{n_i}] \quad (18)$$

Subject to

$$f_{cost}(r,n) = \sum_{i=1}^{10} \alpha_i \left(\frac{-1000}{ln\, r_i}\right)^{\beta_i} \left(n_i + e^{\frac{n_i}{4}}\right) \le C$$
$$f_{weight}(r,n) = \sum_{i=1}^{10} w_i n_i\ exp(n_i/4)\ \le W$$
$$f_{volume}(r,n) = \sum_{i=1}^{10} v_i n_i^2\ \le V$$

$$0.5 \le r_i \le 1-10^{-6},\ r_i \in\ R^+$$
$$1.0 \le n_i \le 5,\ n_i \in Z^+$$
$$0.5 \le R_s \le 1-10^{-6}$$

5. Results and comparisons

The WHO algorithm is used to find suitable solutions to the previously discussed problems. in such optimization problems, the number of redundant components (*n*) and the corresponding reliability (*r*) of each component in all the subsystems with various constraints must be determined concurrently.

Results gained after applying WHO to P1, P2, and P3 are compared to those of the simplified swarm algorithm (SSO) [8], and the attraction-repulsion imperialist competitive algorithm (AR-ICA) [10], and hybrid salp swarm algorithm with teaching-learning based optimization (HSS-TLBO) [16] As for P4, the obtained results are compared to PSO [9] and GBO [13]. Results are given in Table 10, the best solution for each problem is stated and compared to the above-mentioned methods.

To assess the enhancement achieved, the "maximum possible improvement" (*MPI*) is calculated to show the amount of improvement made in the obtained solutions over previous ones. The MPI is given as in Eq. (19) [7].

Table 6. Comparing WHO with AR-ICA and SSO for P1

	R_s	n_i	r_i	MPI
AR-ICA	0.99997661	2 2 2 2 4	0.82201264 0.84365640 0.89129092 0.89869886 0.86824939	0.166%
SSO	0.99997657			0.337%
WHO	**0.9999766489**	**2 2 2 2 4**	**0.8199933074** **0.8451438315** **0.8955732508** **0.8954101684** **0.8683173737**	

Table 7. Comparing WHO with HSS-TLBO for P1

	R_s	n_i	r_i	MPI
HSS-TLBO	0.99998633737	3 2 2 2 4	0.7753618512628 0.8714241422773 0.8903702230415 0.8914438741116 0.8630261550595	0.0031 %
WHO	**0.99998633779**	**3 2 2 2 4**	**0.775007936893** **0.871112710024** **0.891103780170** **0.891343630734** **0.863097073724**	

Table 8. Comparing WHO with AR-ICA, SSO, and HSS-TLBO for P2

	R_s	n_i	r_i	MPI
AR-ICA	0.99988963	3 3 2 4 1	0.82764257 0.85747845 0.91419677 0.64927379 0.70409200	0.007%
SSO	0.99988862			0.91%
HSS-TLBO	0.99988963738	3 3 2 4 1	0.8280051677 0.8578130972 0.9142533044 0.6482662731 0.7038807118	0.0001 %
WHO	**0.9998896375**	**3 3 2 4 1**	**0.8281266446** **0.8577893836** **0.9142001356** **0.6482089690** **0.7038280579**	

$$MPI(\%) = \frac{R_{s\_WHO} - R_{s\_other}}{1-R_{s\_other}} \quad (19)$$

Where the reliability is considered to be ≤1, $R_{s\text{-}WHO}$ is the system reliability obtained using WHO algorithm and $R_{s\text{-}other}$ is that found by other methods in the comparison.

Table 9. Comparing WHO with AR-ICA, SSO, and HSS-TLBO for P3

	R_s	n_i	r_i	MPI
AR-ICA	0.999954673	5 6 4 5	0.90148988 0.85003526 0.94812952 0.88823833	0.0022%
SSO	0.99995416		Not Given	1.12%
HSS-TLBO	0.999954674664	5 6 4 5	0.901623877 0.849936249 0.948146758 0.888204712	2.42689E-05%
WHO	**0.999954674675**	**5 6 4 5**	**0.901615299** **0.849920747** **0.948140499** **0.888223413**	

Tables 6, 7, 8, and 9 show the comparison among WHO and the above-mentioned methods for test problems P1, P2, and P3. In problem P1, HSS-TLBO [16] used a different value for (w_5) input as previously indicated in Table 2. That is why two comparisons are made, Table 6 shows the results using ($w_5 = 4.5$), whereas Table 7 displays the results of ($w_5 = 3.5$). Each table represents a comparison with previous algorithms according to the (w_5) value they used.

Results show that WHO achieved better results than the other methods using only (25) population individuals in less than (75) iterations. As for the (MPI), it indicated a small amount of improvement in the solutions.

Table 10 shows the results of comparing WHO to PSO and GBO for test problem P4. These results clearly signify the ability of WHO in attaining best obtainable results of high reliability in an efficient manner using (100) individual in (250) iterations. Here the MPI shows a noticeable improvement in the solution found.

Adding up all the results together, the WHO algorithm has proven, through testing and comparisons, to act efficiently in navigating through the search space of this complicated problem. It also showed a fast convergence in finding acceptable and better near-optimal solutions in terms of overall system reliability.

6. Conclusion

In this work, the wild horse optimization (WHO) was employed to tackle the NP-hard reliability redundancy allocation problem (RRAP), this employment was tested using four known numerical examples, namely: series-parallel, complex/bridge, overspeed protection, and the pharmaceutical plant

Table 10. Comparing WHO with PSO and GBO for P4

	R_s	n_i	r_i	MPI
PSO	0.956021	3 3 3 3 3 3 3 3 3 3	0.871922 0.827480 0.835569 0.800000 0.865663 0.831345 0.864687 0.800000 0.858897 0.832932	6.463 %
GBO	0.958861643	3 3 3 3 3 3 3 3 3 3	0.88225093405317 0.82111281375906 0.82584880693797 0.82524467049383 0.86440310806613 0.83308511765053 0.84593367878544 0.83667825093861 0.84796655134447 0.82602987045366	0.0037 %
WHO	**0.9588631552**	**3 3 3 3** **3 3 3 3** **3 3**	**0.882091158677** **0.821309427801** **0.825819434365** **0.825204093069** **0.864077669011** **0.832956278073** **0.846094331706** **0.83690268494** **0.847159384971** **0.826596578285**	

systems to investigate this complex optimization problem. The test aimed to find best solutions in terms of high overall system reliability. Results of WHO algorithm has proven its ability to discover new optimal solutions, and as demonstrated in the testing process and comparisons, solutions found outperformed those found by simplified swarm, attraction-repulsion imperialist competitive, hybrid salp swarm with teaching-learning based optimization, particle swarm optimization, and gradient based optimization in terms of optimality. It can be concluded that the overall performance of the used algorithm was superior even when the MPI was small, as it still indicated an enhancement in the solutions attained.

As for future work, other swarm intelligent algorithms can be considered to further enhance solutions or be used to solve other numerical examples. In addition, mutli objective optimization RRAPs can be analysed and studied.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The conceptualization, methodology, selection of numerical examples, implementation, validation, investigation and analysis of results with comparison, preparation and visualization, supervision, writing, review and editing, have all been done by the first author.

References

[1] M. Sadeghi, E. Roghanian, H. Shahriari, and H. Sadeghi, "Reliability Optimization for Non-repairable Series-parallel Systems with a Choice of Redundancy Strategies and Heterogeneous Components: Erlang Time-to-failure Distribution", In: *Proc. of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, Vol. 235, No. 3, pp. 509–528, 2020.

[2] T. C. Chen, "IAs Based Approach for Reliability Redundancy Allocation Problems", *Applied Mathematics and Computation*, Vol. 182, No. 2, pp. 1556–1567, 2006.

[3] W. C. Yeh, Y. Z. Su, X. Z. Gao, C. F. Hu, J. Wang, and C. L. Huang, "Simplified Swarm Optimization for Bi-objection Active Reliability Redundancy Allocation Problems", *Applied Soft Computing*, Vol. 106, p. 107321, 2021.

[4] H. Garg, and S. P. Sharma, "Multi-objective Reliability-Redundancy Allocation Problem Using Particle Swarm Optimization", *Computers and Industrial Engineering*, Vol. 64, Issue. 1, pp. 247-255, 2013.

[5] R. T. Moghaddam, J. Safari, and F. Sassani, "Reliability Optimization of Series–parallel Systems with a Choice of Redundancy Strategies Using a Genetic Algorithm", *Reliability Engineering and System Safety*, Vol. 93, No. 4, pp. 550–556, 2008.

[6] L. S. Coelho, "An Efficient Particle Swarm Approach for Mixed-integer Programming in Reliability Redundancy Optimization Applications", *Reliability Engineering & System Safety*, Vol. 94, No. 4, pp. 830–837, 2009.

[7] W. C. Yeh and T. J. Hsieh, "Solving Reliability Redundancy Allocation Problems Using an Artificial Bee Colony Algorithm", *Computers and Operations Research*, Vol. 38, No. 11, pp. 1465–1473, 2011.

[8] C. Huang and W. Yeh, "Simplified Swarm Optimization Algorithm for Reliability Redundancy Allocation Problems", In: *International Telecommunication Networks and Applications Conference (ITNAC)*, 2015, pp. 1-4, DOI: 10.1109/ATNAC.2015.7366780.

[9] H. Garg and S. P. Sharma, "Reliability-redundancy allocation problem of pharmaceutical plant", *Journal of Engineering Science and Technology*, Vol. 8, No. 2, pp. 190–198, 2013.

[10] D. Afonso, V. C. Mariani, and L. S. Coelho, "Modified Imperialist Competitive Algorithm Based on Attraction and Repulsion Concepts for Reliability-Redundancy Optimization", *Expert Systems with Applications*, Vol. 40, No. 9, pp. 3794–3802, 2013.

[11] L. Sahoo, A. Banerjee, A. K. Bhunia, and S. Chattopadhyay, "An Efficient GA-PSO Approach for Solving Mixed-Integer Nonlinear Programming Problem in Reliability Optimization", *Swarm and Evolutionary Computation*, Vol. 19, pp. 43–51, 2014.

[12] M. R. Valaei and J. Behnamian, "Allocation and Sequencing in 1-out-of-n Heterogeneous Cold-standby Systems: Multi-Objective Harmony Search with Dynamic Parameters Tuning", *Reliability Engineering & System Safety*, Vol. 157, pp. 78–86, 2017.

[13] S. Taghiyeh, M. Mahmoudi, S. Fadaie, and H. Tohidi, "Fuzzy Reliability- Redundancy Allocation Problem of The Overspeed Protection System", *Engineering Reports*, 2: e12221, 2020. DOI:10.1002/eng2.12221.

[14] A. M. Jaleel and M. K. Abd, "Reliability Evaluation of Electric Distribution Network with Distributed Generation Integrated", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 5, pp. 306-319, 2021, doi: 10.22266/ijies2021.1031.28.

[15] Z. Ashraf, M. Shahid, and F. Ahmad, "Gradient Based Optimization Approach to solve Reliability Allocation System", In: *Proc. of the International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 2021, pp. 337-342, DOI: 10.1109/ICCCIS51004.2021.9397197.

[16] T. Kundu, Deepmala, and P. K. Jain, "A hybrid salp swarm algorithm based on TLBO for reliability redundancy allocation problems", *Applied Intelligence*, 2022, DOI: 10.1007/s10489-021-02862-w.

[17] D. W. Coit and E. Zio, "The Evolution of System Reliability Optimization", *Reliability Engineering & System Safety*, Vol. 192, p. 106259, 2018.

[18] H. Marouani and O. A. Mutiri, "Optimization of Reliability–Redundancy Allocation Problems: A Review of the Evolutionary Algorithms", *Computers, Materials & Continua*, Vol. 71, No.

1, pp. 537-571, 2022. DOI:10.32604/cmc.2022.020098.

[19] I. Naruei and F. Keynia, "Wild horse optimizer: A New Meta-Heuristic Algorithm for Solving Engineering Optimization Problems", *Engineering with Computers,* 2021. DOI:10.1007/s00366-021-01438-z.

[20] M. H. Ali, S. Kamel, M. H. Hassan, M. T. Véliz, and H. M. Zawbaa, "An Improved Wild Horse Optimization Algorithm for Reliability Based Optimal DG Planning of Radial Distribution Networks", *Energy Reports*, Vol. 8, pp. 582-604, 2022.