Expert Systems With Applications xxx (xxxx) xxx

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Applying the coral reefs optimization algorithm for solving unequal area facility layout problems

Q1 L. Garcia-Hernandez[a,*], L. Salas-Morera[a], J.A. Garcia-Hernandez[a], S. Salcedo-Sanz[b], J. Valente de Oliveira[c]

[a] Area of Project Engineering, University of Cordoba, Spain
[b] Department of Signal Processing and Communications, Universidad de Alcalá, 28805 Alcalá de Henares, Madrid, Spain
[c] University of Algarve, Portugal

A R T I C L E I N F O

Article history:
Received 17 January 2019
Revised 9 July 2019
Accepted 15 July 2019
Available online xxx

Keywords:
Unequal area facility layout problem
Coral reefs optimization
Facility layout
Meta-heuristic
Bio-inspired algorithms

A B S T R A C T

Coral Reefs Optimization (CRO) is a recently proposed evolutionary-type algorithm which has shown promising results to tackle many complex optimization problems. This paper discusses the performance of this meta-heuristic in Unequal Area Facility Layout Problems (UA-FLPs). The UA-FLP is an important problem in industrial production, which considers a rectangular region and a set of rectangular facilities. These facilities must be allocated in the plant in the most adequate way satisfying certain constraints. The *Flexible Bay Structure* has been selected in order to represent solutions for the UA-FLP in the proposed CRO algorithm. In this paper, we detail the implementation of the algorithm and provide the results of different tests in several UA-FLP instances with different size and setting. The obtained results confirm the excellent performance of the proposed algorithm in solving UA-FLPs, improving alternative algorithms devoted to this problem in the literature.

1. Introduction

Facility Layout Design (FLD) decides the allocation of departments (or *facilities*) in a manufacturing layout, trying to reach well laid out facilities taking into account some objectives or criteria, under certain constraints. Considering Tompkins, White, Bozer, and Tanchoco (2010), a good distribution of the departments implies improvements in the efficiency and can decrease the total expenses in a company between 20% and 50%. For this reason, FLD is a very important issue to consider in order to reduce expenses and other work resources in a manufacturing (Kouvelis, Kurawarwala, & Gutierrez, 1992). There are many different Facility Layout Problems (FLPs) in FLD applications, which are determined by several features and design factors. In this respect, it is possible to find some classifications and taxonomies for FLPs in the works by Drira, Pierreval, and Hajri-Gabouj (2007), Hosseini-Nasab, Fereidouni, Fatemi Ghomi, and Fakhrzad (2018) and Anjos and Vieira (2017), among others. A particularly interesting FLP, due to its direct application to real cases, is known as Unequal Area Facility Layout Problem (UA-FLP). The UA-FLP was first described by Armour and Buffa (1963), and it takes into account an industrial plant and a set of unequal departments, both of them with rectangular shape. Then, the facilities must to be allocated adequately in the layout. As main constraints, in this version of FLP the overlap between facilities is not allowed and, in addition, they must be allocated within the boundary of the space plant layout. Normally, the main objective of UA-FLP is to minimize the cost of material flow between the departments that make up the industrial plant. (Gonçalves & Resende, 2015).

Different approaches have been recently applied aiming at solving the UA-FLP. In Komarudin and Wong (2010) it is established that it is possible to classify the approaches that solve this problem into deterministic procedures and heuristics/metaheuristics methods. Taking into consideration the deterministic methods, Meller, Narayanan, and Vance (1998) suggested a branch and bound approach that included a structure with an acyclic subgraph for solving this problem. In this sense, Montreuil (1991) and Konak, Kulturel-Konak, Norman, and Smith (2006) applied to UA-FLPs a proposal based on mixed integer programming. Afterward, Meller et al. (1998) modified Montreuil's proposal in order to solve large UA-FLPs. They reached an optimal solution for a UA-FLP with eight facilities. Later, Sherali, Fraticelli, and Meller (2003) suggested a upgraded model that solved more efficiently UA-FLPs by means of decreasing the amount of error. Moreover, Castillo, Westerlund, Emet, and Westerlund (2005) reached

* Corresponding author.
E-mail addresses: ir1gahel@uco.es (L. Garcia-Hernandez), lsalas@uco.es (L. Salas-Morera), sancho.salcedo@uah.es (S. Salcedo-Sanz).

https://doi.org/10.1016/j.eswa.2019.07.036
0957-4174/

optimal solution solving an UA-FLP of nine facilities using the same approach than Sherali et al. (2003) with some improvements. Recently, Saraswat, Venkatadri, and Castillo (2015) and Purnomo and Wiwoho (2016) used the proposal taken from Sherali et al. (2003) in order to consider more than one objective. Chae and Regan (2016) reached optimal designs for problems up to 12 facilities. They also considered both fixed and flexible dimensions for facilities.

In general, meta-heuristics methods perform better than deterministic algorithms for UA-FLPs, mainly in large and very large instances. That is why heuristic and meta-heuristic approaches have been more frequently used for solving UA-FLPs. For example, Tam (1992) developed a Simulated Annealing approach called LOGIC in order to find best solutions for this problem. More recently, Scholz, Petrick, and Domschke (2009) and Kulturel-Konak (2012) proposed Tabu search proposals for the UA-FLP.

Many researches have employed Genetic Algorithms (GAs) for solving UA-FLPs. This way, Tate and Smith (1995) suggested a GA that included a penalty function in order to focus the process of finding solutions only to the feasible ones. Azadivar and Wang (2000) addressed the UA-FLP by means of a GA that used a Slicing Tree Structure as layout representation. Considering aisles in the UA-FLP, Wu and Appleton (2002) and Gomez, Fernandez, De la Fuente Garcia, and Garcia (2003) proposed GA approaches for solving this problem. Enea, Galante, and Panascia (2005) used a GA to UA-FLP considering a fuzzy environment and also aspect ratio constraints. Moreover, Aiello, Enea, and Galante (2006) implemented a combination of a GA and Electre algorithm to address the UA-FLP. Liu and Meller (2007) applied an approach that combined Mixed-Integer Programming and GA to solve this problem. They deleted unfeasible features in order to easily solve the problem. Continuing with genetic approaches applied to this problem, García-Hernández, Pierreval, Salas-Morera, and Arauzo-Azofra (2013b) suggested an approach that combined Interactivity and a GA for capturing those features that the Decision Maker (DM) preferred in a particular solution. Their Interactive Genetic Algorithm was improved by García-Hernández, Palomo-Romero, Salas-Morera, Arauzo-Azofra, and Pierreval (2015) for achieving more diversity in the final layout solutions. In this respect, García-Hernández, Arauzo-Azofra, Salas-Morera, Pierreval, and Corchado (2015) reached an improvement by means of considering both Decision Maker preferences and quantitative factors in the final solution. They achieved it through an interactive multi-objective GA. More recently, Palomo-Romero, Salas-Morera, and GarcíHernández (2017) suggested a proposal that improved the quantitative performance of many of tested UA-FLPs using a GA based on an Island Model to explore different individuals from the varying search context.

Alternative meta-heuristics have also been used to address UA-FLPs. For example, ant colony optimization (Komarudin & Wong, 2010) (Wong & Komarudin, 2010) (Kulturel-Konak & Konak, 2011) (Liu & Liu, 2019), artificial immune system (Ulutas & Kulturel-Konak, 2012), biased random-key GA (Gonçalves & Resende, 2015), collision detection and response approach (Sikaroudi & Shahanaghi, 2016), GA combined with a decomposition strategy (Paes, Pessoa, & Vidal, 2017), among others. Finally, Kang and Chae (2017) solved UA-FLP by means of a modification of the Harmony Search method proposed by Shayan and Chittilappilly (2004). Additionally, they presented a new slicing tree representation for layout configuration.

In order to represent the plant layout design, some different approaches have been developed. The Block Layout Design Problem (BLDP) representation allows locating every facility in the plant freely in any position with the restriction of not overlapping with other facilities. In such representation, Mixed Integer Linear and Nonlinear Programming methods are used (Castillo et al., 2005; Gonçalves & Resende, 2015; Meller & Gau, 1996). In the search for a representation more useful to apply evolutionary algorithms, two more facilities layout representations have been proposed: Slicing Tree Structure (STS) and Flexible Bay Structure (FBS). In STS, the space is recursively divided into vertical and horizontal sections (Kang & Chae, 2017; Komarudin & Wong, 2010; Scholz et al., 2009; Shayan & Chittilappilly, 2004) while in FBS, the space is only divided into horizontal or vertical bands (Kulturel-Konak & Konak, 2011; Meller, 1997). In this way, STS and FBS structures are not comparable nor in the way they use to locate the facilities in the plant, neither in the results obtained by each one of them.

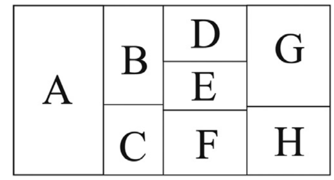

Fig. 1. Layout representation based on FBS.

A representation based on the Flexible Bay Structure (FBS) has been selected in this paper in order to represent a facility layout as an individual in an evolutionary-type algorithm. With respect to the advantages of using FBS as layout representation, it is can be stated that considering FBS as layout representation permits the UA-FLP become simpler and easier to be addressed, because of the UA-FLP complexity is decreased into determining the facilities location order and the total number of facilities that each bay will contain (Wong & Komarudin, 2010). Additionally, this kind of representation which was suggested by Tong (1991) has been widely used among the different structures available from the related references (Liu & Liu, 2019; Palomo-Romero et al., 2017; Wong & Komarudin, 2010). This mechanism of illustrating plant layout consists of an area with rectangular shape that is vertically or horizontally split into sub-areas (called bays). Then, each one is split again to assign the departments that compose the manufacturing plant. According to Tate and Smith (1995), the generated subareas possess the property of having flexible width in order to have enough space for containing different number of facilities. Finally, according to Aiello, Scalia, and Enea (2012), using FBS offers an additional benefit due to it gives the possibility of incorporating aisles in an easy way. Fig. 1 shows a facility layout representation based on FBS. This FBS example has been taken from Palomo-Romero et al. (2017).

In this work we test the performance of a different current evolutionary-based algorithm, the Coral Reefs Optimization (CRO) (Salcedo-Sanz, Del Ser, Landa-Torres, Gil-López, & Portilla-Figueras, 2013) (Salcedo-Sanz, Del Ser, Landa-Torres, Gil-López, & Portilla-Figueras, 2014a) in order to address the UA-FLP. The CRO is an evolutionary-type algorithm which evolution is guided by imitating processes occurring in real coral reefs, such as reproduction, the fight for space or the predation. The CRO is an algorithm which results in a kind of hybrid Evolutionary Algorithm and Simulated Annealing (Salcedo-Sanz et al., 2014a), and it has been shown to improve both techniques in diverse instances in areas such as Telecommunications (Salcedo-Sanz, Sanchez-Garcia, J.A., Jimenez-Fernandez, & Ahmadzadeh, 2014d) (Salcedo-Sanz, García-Díaz, Portilla-Figueras, Ser, & Gil-López, 2014b), Energy (Salcedo-Sanz, Camacho-Gómez, Mallol-Poyato, Jiménez-Fernández, & DelSer, 2016) (Salcedo-Sanz, Pastor-Sánchez, Prieto, Blanco-Aguilera, & García-Herrera, 2014c), Structural Engineering (Salcedo-Sanz, Camacho-Gómez, Magdaleno, Pereira, & Lorenzana, 2017) (Camacho-Gómez, Wang, Pereira, Díaz, & Salcedo-Sanz, 2018) or Bio-medical applications (Bermejo, Chica, Damas, Salcedo-Sanz, & Cordón, 2018) (Yan, Ma, Luo, & Patel, 2019). Recently,

Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

Facility sequence								Bay divisions						
A	B	C	D	E	F	G	H	1	0	1	0	0	1	0

Fig. 2. Facility layout chromosome.

the CRO has also been used to different problems such as clustering (Medeiros, Xavier, & Canuto, 2015), neural network training (Yang, Zhang, & Zhang, 2016), time series analysis (Durán-Rosal, Gutiérrez, Salcedo-Sanz, & Hervás-Martínez, 2018) or resource allocation problems (Ficco, Esposito, Palmieri, & Castiglione, 2018), among others. In these works, the CRO has been successfully applied by reaching an excellent performance in the tested problem (Salcedo-Sanz, 2017). This work deals to investigate the performance of Coral Reefs Optimization addressing the UA-FLP. From the best of our knowledge, it is the first time that CRO is applied to solve the UA-FLP. We will show that the CRO algorithm is able to outperform other evolutionary based approaches in a number of large UA-FLP instances.

The remainder of this work has been organized as follows: Section 2 details the novel suggested approach for solving the UA-FLP. Section 3 describes the experimental part of the work, with the results achieved in many different UA-FLPs. A comparison with published results reached by other approaches is carried out at this stage. Finally, Section 4 closes this research with a summary of the main concluding remarks and some future research lines that can be drawn based on this work.

2. Proposed approach

For addressing the UA-FLP we propose a new CRO approach which considers material flow as optimization criterion. Below, we will describe the algorithm's structure and implementation.

2.1. Individual codification

In order to encode an individual of the CRO reef, the chromosome structure suggested by Gomez et al. (2003) has been used. It is illustrated in Fig. 2. This encoding structure is formed by two different segments. The first one illustrates the sequence of departments in the facility layout, which is taken reading from top to bottom in each bay and reading the bay from left to right in the facility layout. An integer permutation from 1 to n (being n the total number of departments that exist in the layout) is employed in the first segment. The information about where are the cuts that delimit the bays of the layout is offered by the second segment. This one is composed by $(n-1)$ elements which have binary values. So that, if it is the value' 1' in a certain segment position means that the department in the same segment position of the first segment, is the last element of the bay. Else, it will appeared the value' 0' in the segment. Fig. 2 gives the individual chromosome associated to the facility representation offered in Fig. 1.

2.2. Objective function

Armour and Buffa (1963) stated the UA-FLP for the first time. The problem is defined by means of a rectangular layout of dimensions $(W \times H)$ which are fixed. Additionally, there is a group of facilities or departments with a determined area (A_i). The sum of the department areas must be less or equal than the total area of the rectangular layout (see Eq. (1)).

$$\sum_{i}^{n} A_i \leq W \times H \tag{1}$$

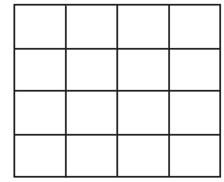

Fig. 3. Example of a coral reef with size 4 x 4.

The objective of the problem is to place all the departments in the layout, optimizing a given criterion and taking into consideration that overlapping between departments is not allowed. In Aiello et al. (2012) it is stated that the UA-FLP involves as main objective the minimization of the material flow between departments. The fitness score used in evolutionary algorithms to evaluate UA-FLP test problems is therefore based on material flow. Additionally, in order to guide the search process to feasible individuals, a penalty function proposed by Tate and Smith (1995) have been used. This way, for every solution in the algorithm, a penalty mark is defined, which is proportional to the number of facilities that make up the layout and that not satisfy the aspect ratio constraint (either the maximum aspect ratio or minimum side length). These facilities are considered as *unfeasible*. The fitness function that minimizes the material flow is the following:

$$g(\mathbf{x}) = \sum_{i}^{n}\sum_{j}^{n} f_{ij} d_{ij} + (D_{inf})^k (V_{feas} - V_{all}) \tag{2}$$

where n is the number of departments in the layout, f_{ij} is the material flow between the departments i and j, d_{ij} is the Manhattan distance between i and j, $Dinf$ is the number of facilities which are unfeasible, $Vfeas$ is the best feasible fitness value that has been yet achieved, $Vall$ is the best overall fitness value that has been yet achieved, an k is a penalty parameter that fits the value of the penalty function (it has been set as 3, following the suggestion in Tate & Smith, 1995).

2.3. The Coral Reef Optimization Algorithm

The Coral Reef Optimization Algorithm (CRO) was recently proposed by Salcedo-Sanz et al. (2014a). This approach is a kind of evolutionary-type algorithm which imitates the evolution of coral reefs and the different processes occurring in these ecosystems. We will consider Λ as a model of the reef with size of $N \times M$ square grid (see Fig. 3). Each square located in $\Lambda(i, j)$ is a place that can host a coral $\Xi(i, j)$ where i and j are the coordinates of the square in the reef. Each coral is a representation of a solution to our problem, in our particular case, a plant layout solution for the UA-FLP. Once we have modeled the reef and the corals itself, the algorithm process is define using the steps that are detailed as follows.

2.3.1. Initialization of the algorithm

One of the most important parameters of the CRO algorithm is the number of initial corals in the reef. A rate specifying the proportion between empty and in-use squares in the reef is defined, ρ_0, in such a way that $0 < \rho_0 < 1$. Taking into consideration this parameter, the initial number of corals is calculated as:

$$InitialCorals = N \times M \times \rho_0$$

The initial corals are randomly generated and placed (also in a random way) in empty squares of the reef. Fig. 4 illustrates a coral reef initialized with random corals in a proportion of' 0.5' between empty and in use squares, i.e. $\rho_0 = 0.5$. This step is summarized in Algorithm 1. Once the reef are initialized, the simulation of the corals' evolution starts with an iterative execution of the corals' reproduction, which is realized by means of diverse operators until

Algorithm 1 Reef initialization.

Input Reef size (width and height) and occupation rate
Output Initial reef population

procedure INITIALIZE REEF(n, m, ρ_0) ▷ Coral Reef initialization
  $ref\_size \leftarrow n \times m$
  $k \leftarrow reef\_size \times \rho_0$ ▷ Number of initial corals
  for k times **do**
    generate **random** coral
    place coral in random empty reef position
  end for
  return initial reef
end procedure

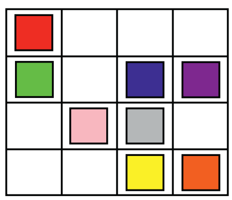

Fig. 4. Example of a coral reef with random individuals inserted and $\rho_0 = 0.5$.

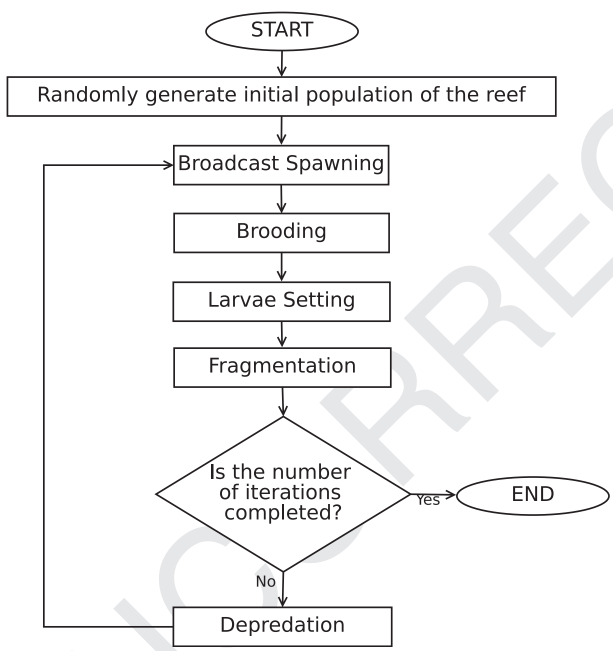

Fig. 5. Proposed CRO algorithm flowchart diagram.

the stop criterion is reached (in our particular case, when the required number of iterations have been satisfied). This iterative process (detailed in Salcedo-Sanz et al., 2014a) will be described in the following section.

2.3.2. Iterative coral evolution

The reproduction phase is defined by different operators for modeling *Sexual Reproduction* (that can be external and internal) and *Asexual Reproduction*. All these kind of reproduction phases will generate new corals from the existing ones in the reef which will be denoted as *larvae*. Between sexual and asexual reproduction phases, is the *Larvae Setting* step, where some of the new *larvae* elements will take place into the coral reef. Finally, a *depredation phase* will eliminate the weakest corals in the reef. Fig. 5 summarizes the entire process of the CRO algorithm. Additionally, Algorithm 2 shows the flowchart diagram of the CRO algorithm with the different CRO phases which are detailed below.

Algorithm 2 CRO algorithm.

Input Algorithm's control parameters
Output Feasible solution with best *fitness*

procedure CRO($n, m, \rho_0, F_b, F_a, F_d, P_d$) ▷ Coral Reef Optimization algorithm
  initialize reef with size $n \times m$ and occupation rate ρ_0
  repeat
    reproduce corals fraction F_b by **broadcast spawning**
    reproduce corals fraction $1 - F_b$ by **brooding**
    larvae evaluation
    larvae setting
    reproduce best corals fraction F_a by **asexual reproduction**
    predation of F_d worst reef corals with P_d probability
  until *stop condition*
  return best *feasible* solution
end procedure

1. Broadcast spawning (External sexual reproduction)
This phase is made up by two steps. Firstly, a number of the corals that exist in the reef, denoted by ρ_k, is selected randomly to be *broadcast spawners*. This fraction of *broadcast spawners* is calculated with respect to the overall amount of existing corals in the reef and it is denoted as F_b. The remaining corals which have not been chosen for being *broadcast spawners* $(1 - F_b)$ will be selected for being reproduced in the *brooding* phase. Secondly, from the *broadcast spawners* (ρ_k), the algorithm will select couples of corals in order to be reproduced. This selection of corals is random and with replacement, once a couple is selected, it can not be selected again for being reproduced in the same step. Each of the selected couples will form two children by sexual crossover. Specifically in our approach, the Partially-Mapped Crossover operator (PMX) proposed by Goldberg and robert (1985), is used for the facility sequence segment, and the One Point Crossover (Holland, 1992) is applied over the split segment. Then, a child will be randomly selected as coral larva which is then released out to the water. This crossover process is illustrated in Fig. 6 where it is shown how the layout representations change during CRO algorithm. The larvae result is stored until the *Larvae Setting* phase. Fig. 7 and Algorithm 3 detail the *broadcast spawning* phase.

Algorithm 3 Broadcast spawning.

Input Coral reef, External sexual reproduction rate
Output Generated larvae set

procedure BROADCAST SPAWNING($reef, F_b$)
  $\rho_k \leftarrow coral\_num \times F_b$ ▷ Number of corals to reproduce by broadcast spawning
  select ρ_k corals from $reef$
  pair selected corals
  for each coral pair **do**
    apply **crossover**
    add generated solution to larvae set
  end for
  return generated **larvae** set
end procedure

2. Brooding (Internal sexual reproduction)
The remaining corals of the previous phase $(1 - F_b)$ are selected to be reproduced by *brooding*, which consist of the formation

Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

Fig. 6. Graphical diagram that illustrates Crossover process in Broadcast Spawning step.

of a coral larva by means of a random mutation in each $1 - F_b$ coral element. The obtained larvae is then released out to the water in a similar way than it is performed in the previous phase. Fig. 9 shows *brooding* reproduction over the two corals which have not been selected to be reproduced in the previous phase (Fig. 7). This mutation process is illustrated using Fig. 8 where it is shown how the layout representations change again during CRO algorithm. Moreover, Algorithm 4 expresses how this phase is performed. The resulting larvae is stored until the *Larvae Setting* phase.

3. Larvae setting
   At this moment, all the *larvae* created by *Broadcast Spawning* or *Brooding* are stored. Then, the next step consists of trying to set

Algorithm 4 Brooding.

Input Coral reef
Output Generated larvae set

```
procedure BROODING(reef)
    select all corals not reproduced by broadcast spawning from reef
    for each selected coral do
        apply mutation
        add generated solution to larvae set
    end for
    return generated larvae
end procedure
```


Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

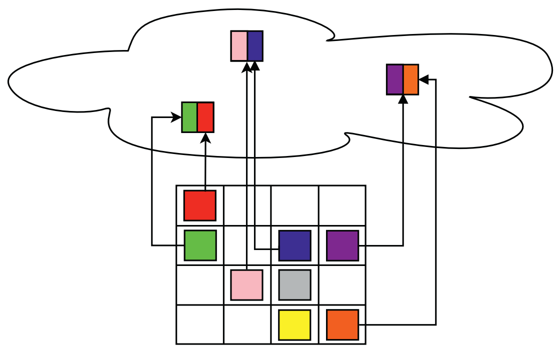

Fig. 7. Graphical diagram that illustrates Broadcast Spawning step.

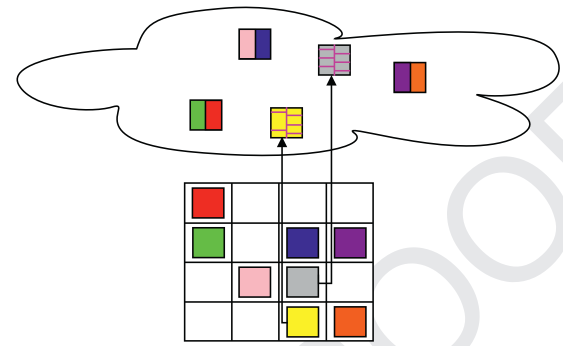

Fig. 9. Graphical diagram that illustrates Brooding phase.

and grow those larvae into the reef. For that matter, the *fitness function* for both larvae and corals that exist in the reef is computed (in our particular case, the fitness function is the existing material flow between the departments that compose the plant layout). Then, a larva is selected to be placed in a random location of the reef. If this position is free, the larva will be allocated there. If it is not, the fitness of the coral and the larva will be compared. This way, if the larva fitness is better (it has less value of material handling cost) than the coral, the coral will be replaced by the larva. If the larva does not replace the coral (it has higher value of material handling cost), it will try κ times (this number is '3' as suggested by Salcedo-Sanz et al., 2013) to be placed in another position of the reef. If the larva can not be placed in κ attempts, it will be deprecated. This mechanism is explained by means of Fig. 10 and Algorithm 5.

4. Budding or fragmentation (Asexual reproduction)
In this phase, all the existing corals in the reef are ranked as a function of their level of *fitness*. Then, a fraction of them denoted by F_a, is duplicated itself and tries to be allocated in a different square in the reef. This is performed by means of the same process that has been explained in the *Larvae Setting* phase. This asexual reproduction is illustrated by means of Fig. 11 and Algorithm 6.

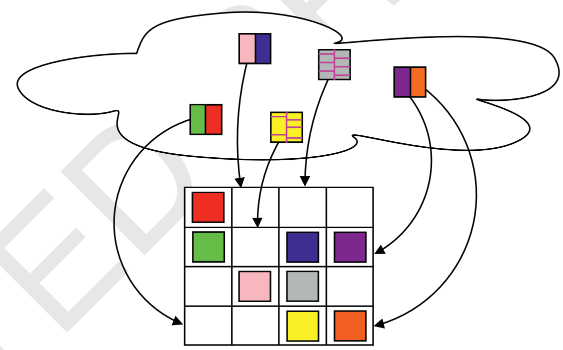

Fig. 10. Graphical diagram that illustrates larvae setting phase.

5. Depredation
At the end of each algorithm iteration, a fraction of the worse fitness corals denoted by F_d that exist in the reef will be deprecated with a very low probability denoted by P_d. This liberates space in the reef for next coral generation. Depredation step is shown using Fig. 12 and Algorithm 7.

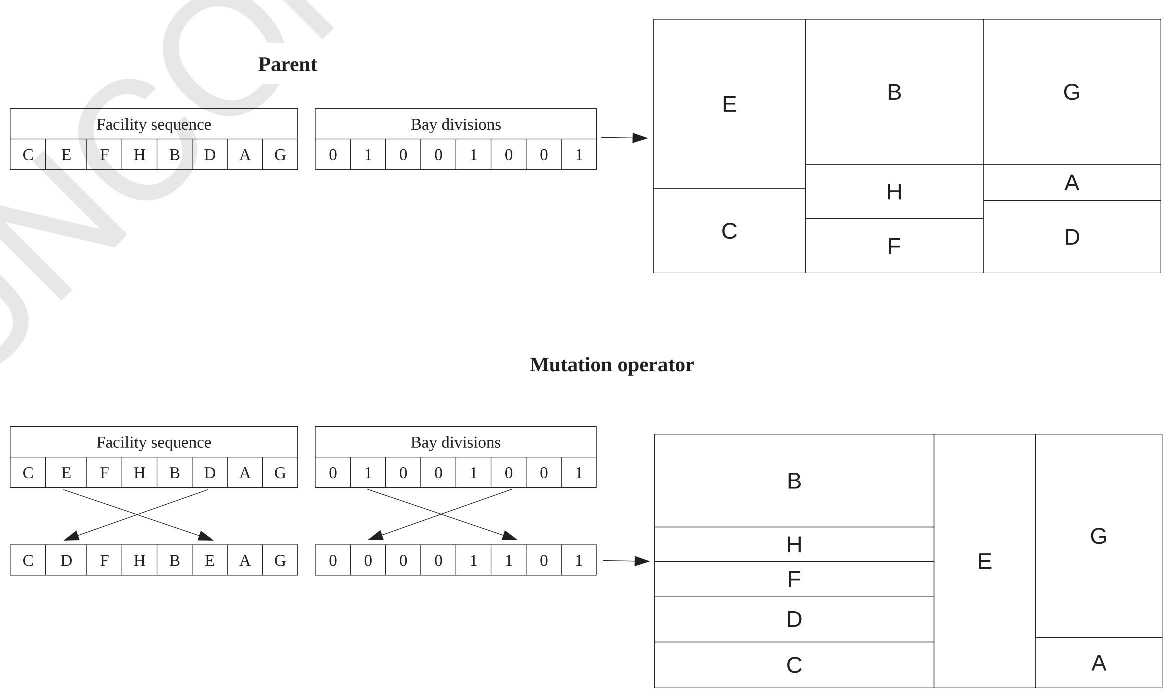

Fig. 8. Graphical diagram that illustrates Mutation process in Brooding step.

Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

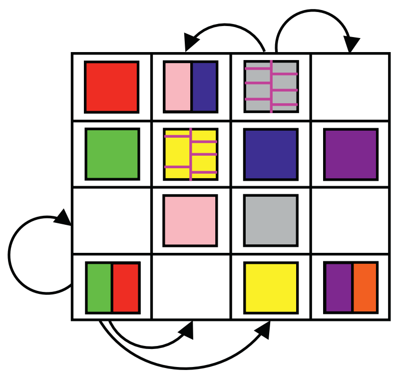

Fig. 11. Graphical diagram that illustrates budding phase.

Algorithm 5 Larvae setting.

Input Coral reef, larvae set
Output Updated reef

```
procedure LARVAE_SETTING(reef, larvae)
    for each larvae do
        placed ← False
        k ← 3                ▷ Number of attempts to settle in the reef
        while not placed and k > 0 do
            pos ← random reef position
            if pos is empty or larva fitness is better than resident's then
                larva settles in pos
                placed ← True
            else
                k ← k − 1
            end if
        end while
    end for
    return reef
end procedure
```

Algorithm 6 Asexual reproduction.

Input Coral reef, Asexual reproduction rate
Output Updated reef

```
procedure ASEXUAL_REPRODUCTION(reef, F_a)
    n_a ← coral_num × F_a          ▷ Number of corals to duplicate
    select the best n_a corals from reef
    for each selected coral do
        settle coral in reef       ▷ Same procedure as larvae_setting
    end for
    return reef
end procedure
```

Algorithm 7 Depredation.

Input Coral reef, depredation fraction, depredation probability
Output Updated reef

```
procedure DEPREDATION(reef, F_d, P_d)
    n_d ← coral_num × f_d          ▷ Number of corals that may be predated
    select the worst n_d corals from reef
    for each selected coral do
        if random(0.0, 1.0) <= P_d then
            remove coral from reef
        end if
    end for
    return reef
end procedure
```

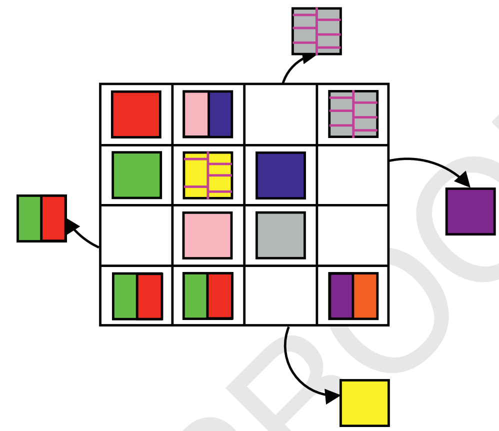

Fig. 12. Graphical diagram that illustrates depredation step.

3. Experimental set and results obtained

The performance of the proposed CRO approach is tested in comparison with state-of-the-art algorithms for the UA-FLP in this section. For this, we have used many UA-FLP instances taken from other works of related references. The set of well-known UA-FLPs are: Slaughterhouse detailed in Salas-Morera, Cubero-Atienza, and Ayuso-Munoz (1996); CartonPacks and Chopped-Plastic from García-Hernández, Arauzo-Azofra, Salas-Morera, Pierreval, and Corchado (2013a); O7, O8 and O9, described by Meller et al. (1998); VC10 (both side and aspect ratio constraints) illustrated in van Camp, Carter, and Vannelli (1992); MB12 explained by Bozer and Meller (1997); Ba12 detailed in Bazaraa (1975); Ba14 presented in Komarudin and Wong (2010) of the problem described in Bazaraa (1975); Ma15 (with two different shape constraints) from Bozer, Meller, and Erlebacher (1994); AB20 detailed by Armour and Buffa (1963); SC30, a modification taken from Komarudin and Wong (2010) of the problem described in Liu and Meller (2007); SC35 from Liu and Meller (2007); and DU62 described by Dunker, Radons, and Westkämper (2003).

The characteristics of the selected UA-FLPs for being tested are described in Table 1. This information is the UA-FLP name, number of facilities, facility width, facility height, shape constraint (being α the maximum aspect ratio constraint, and *lmin* the minimum side length constraint), and finally the references for the problem data sources. Note that the used measure distance is the Manhattan as default parameter. However, the Euclidean distance have been applied to the instances of Slaughterhouse, Carton Packs and Chopped Plastic. Note that Ba14 problem has two different values for the minimum side length constraint which is' 1' for the departments that are from 1 to 12, and, it is' 0' for the departments 13 and 14.

The proposed CRO performance deeply depends on a set of parameters. We have tuned them in an empirical way. Thus, we have performed different checks in order to reach the best set of values for the algorithm in the UA-FLP. Table 2 illustrated the best values obtained for the CRO parameters. Taking into consideration the values express in Table 2, a full-factorial experiment has been performed testing sets of UA-FLPs with each possible combination of parameters. Specifically, the representative sets of UA-FLPs which have been selected for tuning our CRO algorithm have been O9 from Meller et al. (1998), Ma15a taken from Bozer et al. (1994) and SC30 taken from Liu and Meller (2007). These problems have been chosen as representative ones in order to consider the different department sizes (small, medium and large) of the UA-FLPs. Then, a comparison between the reached solutions has been done in order to select which parameter option fits better. The best CRO configu-

Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

Table 1
Features of the tested well-known problems.

Problem name	*Fac.*	*W × H*	*Aspect ratio*	*Reference*
Slaughterhouse	12	51.14 × 30.00	α=4	Salas-Morera et al. (1996)
CartonPacks	11	20.00 × 14.50	α=4	García-Hernández et al. (2013a)
ChoppedPlastic	10	10.00 × 30.00	α=4	García-Hernández et al. (2013a)
O7	7	8.54 × 13.00	α=4	Meller et al. (1998)
O8	8	11.3 × 13.00	α=4	Meller et al. (1998)
O9	9	12.00 × 13.00	α=4	Meller et al. (1998)
vC10Ra	10	25.00 × 51.00	α=5	van Camp et al. (1992)
Vc10Rs	10	25.00 × 51.00	*Min.side*=5	van Camp et al. (1992)
Ba12	12	6.00 × 10.00	*Min.side*=1	Bazaraa (1975)
MB12	12	6.00 × 8.00	α=4	Bozer and Meller (1997)
Ba14	14	7.00 × 9.00	*Min.side*={1,0}	Komarudin and Wong (2010)
Ma15	15	15.00 × 15.00	α=5	Bozer et al. (1994)
Ma15s	15	15.00 × 15.00	*Min.side*=1	Bozer et al. (1994)
AB20	20	2.00 × 3.00	α=5	Armour and Buffa (1963)
SC30	30	12.00 × 15.00	α=5	Liu and Meller (2007)
SC35	35	16.00 × 15.00	α=4	Liu and Meller (2007)
Du62	62	Arbitrary × Arbitrary	α=4	Dunker et al. (2003)

Table 2
CRO parameters selection.

UA-FLP	Chosen values			Tested values		
	O9	*Ma15s*	*SC30*	*Combination of:*		
N × M	25 × 25	25 × 25	25 × 25	10 × 10	15 × 15	25 × 25
ρ_0	c0.4	0.4	0.4	0.4	0.5	0.6
F_b	c0.9	0.9	0.9	0.8	0.85	0.9
F_a	c0.1	0.1	0.2	0.1	0.15	0.2
F_d	0.1	0.1	0.1	0.01	0.05	0.1
P_a	c0.1	0.1	0.1	0.01	0.05	0.1

Table 3
Statistical results reached by the CRO algorithm.

Problem name	*OFV Best*	*OFV Mean*	*CPU(s)*
Slaughterhouse	3487.12	3487.12	78.00
CartonPacks	80.91	80.91	74.00
ChoppedPlastic	265.77	265.77	65.00
O7	134.16	134.16	4.00
O8	245.48	245.48	24.00
O9	239.44	239.44	49.00
vC10Ra	20142.13	20576.93	61.00
Vc10Rs	22897.65	22898.65	63.00
Ba12	8021.0	8103.96	87.00
MB12	125.00	125.00	81.00
Ba14	4665.93	4731.23	92.00
Ma15	26800.63	26972.95	104.00
Ma15s	22871.97	23034.88	106.00
AB20	5243.95	5250.02	202.00
SC30	3519.44	3566.27	622.00
SC35	4263.3	4409.34	552.00
Du62	713876.55	3719342.03	871.00

ration for each representative UA-FLP instance has been shown in the column ‘Chosen value’.

The experimentation has been replicated five times for each UA-FLP like in Komarudin and Wong (2010) with a stopping criteria of 10,000 iterations as maximum and 500 iterations without improvement. The CRO algorithm was coded with Python 2.7.3. All experiments were performed using an Intel Core i5 6200U (2.30 GHz × 4), 8GB RAM and a Linux operating system.

3.1. Results

Table 3 presents the statistical results obtained by the suggested CRO algorithm. For each UA-FLP, the best objective function value (best OFV), the mean objective function value (mean OFV) and CPU time (in seconds) for reaching the best objective function value, are detailed. From the table, it can be extract that the CRO algorithm is robust because of the percentage of gap between the best and mean objective function value is relatively low. This gap usually increases as the number of facilities increases in the UA-FLP. Regarding CPU time, See and Wong (2008) stated that in facility layout design the CPU time is not an extremely important issue. In this context, our proposal is able to reach satisfactory solutions in an reasonable CPU time if it is compared to alternative approaches (as for instance Komarudin & Wong, 2010; Palomo-Romero et al., 2017, among others).

A comparison of the results reached by our CRO algorithm and the results taken from related references that uses both FBS and STS, have been performed in order to analyze the performance of the proposed CRO approach. This information is shown by means on Tables 4 and 6. The first one (Table 4) offers for each data set problem the following information: The best known solution result, its associated layout representation, and also, the reference of the paper that obtained it. Additionally, taking into account that we have used FBS as layout representation in our approach, Table 4 also presents for each problem, the best known solution results and their associated reference considering particularly FBS as layout representation. In this table, we have set in bold font those results reached by our proposed approach which are the best known results. This way, regarding Tables 4 and 5, it can be seen that our proposal reaches or improves the best solution fitness in 7 cases out of 17 tested problems when considering both STS and FBS as layout representation. This fact (our proposal reaches or improves the best solution) happens in 14 cases out of 17 tested problems when we consider exclusively FBS representation. In the remaining cases, our approach is able to reach solutions very close to the best known ones.

According with Kang and Chae (2017) the STS can reach layout solutions that cannot be represented by means of FBS. That is the reason why in most cases, the solutions obtained using STS achieve better results than those that are reached using FBS. For this reason, we consider interesting to analyze the result compar-

Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

Table 4
Summary of test problems and their best-known and best-known FBS solutions.

Problem	*Best known*	*Layout represent.*	*Reference*	*Best known FBS*	*Reference*
Slaughterhouse	**3487.12**	FBS	This approach	**3487.12**	This approach
CartonPacks	**89.02**	FBS	This approach	**89.02**	This approach
ChoppedPlastic	**265.77**	FBS	This approach	**265.77**	This approach
O7	131.56	STS	Gonçalves and Resende (2015)	**134.16**	This approach
O8	243.12	STS	Wong and Komarudin (2010)	**245.48**	This approach
O9	236.14	STS	Kang and Chae (2017)	**239.44**	This approach
Vc10Ra	18522.79	STS	Kang and Chae (2017)	**20142.13**	This approach
Vc10Rs	19951.17	STS	Gonçalves and Resende (2015)	**22897.65**	This approach
Ba12	**8021.0**	FBS	This approach	**8021.0**	This approach
MB12	**125.00**	FBS	This approach	**125.00**	This approach
Ba14	4628.79	STS	Gonçalves and Resende (2015)	**4665.93**	This approach
Ma15a	**26800.63**	FBS	This approach	**26800.63**	This approach
Ma15s	**22871.97**	FBS	This approach	**22871.97**	This approach
AB20	4959.11	STS	Kang and Chae (2017)	**5243.95**	This approach
SC30	3352.70	STS	Kang and Chae (2017)	3443.34	Kulturel-Konak and Konak (2011)
SC35	3316.77	STS	Gonçalves and Resende (2015)	3613.11	Kulturel-Konak and Konak (2011)
Du62	3635307.0	STS	Kang and Chae (2017)	3641497.00	Kulturel-Konak and Konak (2011)

Table 5
Test result comparisons between the best solutions reached by our CRO algorithm and alternative published FBS approaches.

Problem	*CRO*	*Palomo(2017)*	*Kulturel-Konak (2011)*	*Kulturel-Konak (2012)*	*Wong (2010)*	*Enea (2005)*
Slaughterhouse	**3487.12**	–	–	–	–	3854.00
CartonPacks	**89.02**	–	–	–	–	94.10
ChoppedPlastic	**265.77**	–	–	–	–	377.18
O7	**134.16**	134.19	–	–	–	–
O8	**245.48**	245.51	–	–	–	–
O9	**239.44**	241.06	–	–	241.06	–
Vc10Ra	**20142.13**	**20142.13**	**20142.13**	21463.07	21463.1	–
Vc10Rs	**22897.65**	22899.65	22899.65	22899.65	22899.65	–
Ba12	**8021.0**	8435.83	8129.00	**8021.0**	8786.00	–
MB12	**125.00**	**125.00**	–	–	–	–
Ba14	**4665.93**	4665.93	4780.91	4739.74	5004.55	–
Ma15a	**26800.63**	–	27545.27	–	27545.30	–
Ma15s	**22871.97**	–	23197.80	–	23197.80	–
AB20	**5243.95**	5256.10	5336.36	5297.6	5677.83	–
SC30	3519.44	3613.11	**3443.34**	3563.95		–
SC35	4263.3	3885.29	**3700.75**	–	–	–
Du62	3713876.55	–	**3641497.00**	–	–	–

Table 6
Summary of the results reached by the proposed CRO.

Problem name	*Best sol.*	*FBS Diff(%)*	*STS Diff(%)*	*Solution by CRO*
Slaughterhouse	**3487.12**	**10.52**	**10.52**	1 | 8-2 | 4–5 | 12-7-6 | 11-3-10-9
CartonPacks	**89.02**	**5.70**	**5.70**	2-6-11 | 9-10-1-8 | 5-4-7-3
ChoppedPlastic	**265.77**	**41.61**	**41.61**	10-2-3-4-5-6-7 | 1-9-8
O7	**134.16**	**0.02**	- 1.93	3-5-7-8 | 1-4-6-2
O8	**245.48**	**0.02**	- 0.96	5-8-6-3 | 2-1-4-7
O9	**239.44**	**0.69**	- 1.37	5-9-6-2-3 | 8-1-4-7
Vc10Ra	**20142.13**	**0.00**	- 8.03	5-8-10-9-2-6-1 |-7-3
Vc10Rs	**22871.97**	**1.43**	- 12.86	7 | 5-10-9 | 3 | 11 | 12 | 8 | 6 | 4-2 | 1
Ba12	**8021.0**	**0.00**	**0.00**	4–10 | 9-5-7 | 3 | 2–12 | 1 | 11-8-6
MB12	**125.00**	**0.00**	**0.00**	12 | 10-7-3-4-2-8-6-5-1-9 | 11
Ba14	**4665.93**	**0.00**	- 0.79	7-11-5 | 10 | 1 | 3 | 9 | 4-2 | 13-1-4-12-8-6
Ma15a	**26800.63**	**2.78**	**2.78**	6–11 | 2-1-8-7-13 | 4-15-3 | 5-14-12-10-9
Ma15s	**22871.97**	**1.43**	**1.43**	9-10-12-15-6-8-11-7 | 14-4-3-13 | 5-2 | 1
AB20	**5243.95**	**0.23**	- 11.20	1-16-11 | 17-13 | 12-9-15 | 3–14 | 19-10 | 6-4-2-7-20 | 18-5
SC30	3519.44	- 2.16	- 4.73	19-34-30-10 | 2-6-22-26 | 17-25-29-35-28-21 | 3-4-1-20
SC35	3885.29	- 4.74	- 14.63	19-34-30-10 | 2-6-22-26 | 17-25-29-35-28-21 | 3-4-1-20
Du62	3713876.55	- 1.9	- 2.11	19-34-30-10 | 2-6-22-26 | 17-25-29-35-28-21 | 3-4-1-20 | 23-33-18-24-32 | 13-15-7-11-8 | 12-34-9 | 14-31-5-27-16

ison of our proposal against other works that use FBS in its approach. In particular, these FBS proposals are taken from Palomo-Romero et al. (2017), Kulturel-Konak and Konak (2011), Kulturel-Konak (2012), Wong and Komarudin (2010) and Enea et al. (2005). Table 5 displays the results achieved by our proposal and the previous ones. For each UA-FLP, we have highlighted in bold the best solution. First, Table 5 shows that the proposed CRO algorithm is able to reach better results than the other compared FBS approaches in most cases. As it was mentioned previously, the CRO algorithm improves the results of 14 out of 17 tested problems. Specifically, note that the suggested CRO approach obtains better solutions than the approach by Enea et al. (2005) in all problems compared: Slaughterhouse, Carton Packs and Chopped Plastic. The CRO also obtained better results than the algorithm by Wong and Komarudin (2010), in all cases of the seven problems in which we compared with this approach. Also, compared with the algorithm

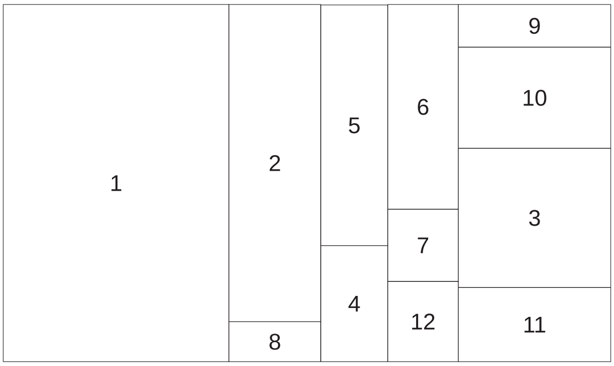

Fig. 13. Best design reached by the proposed CRO approach in the Slaughterhouse UA-FLP.

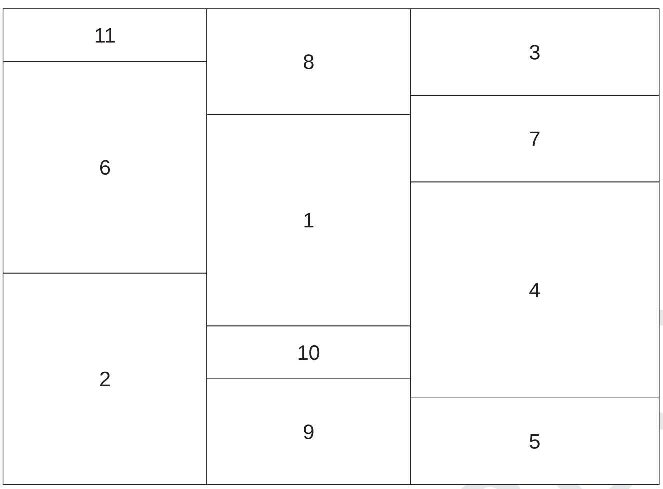

Fig. 14. Best design reached by the proposed CRO approach in the CartonPacks UA-FLP.

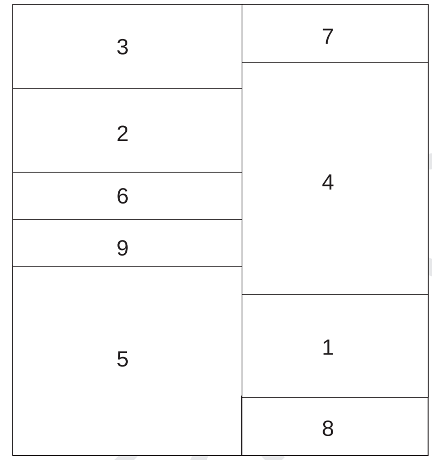

Fig. 16. Best design reached by the proposed CRO approach in the O9 UA-FLP.

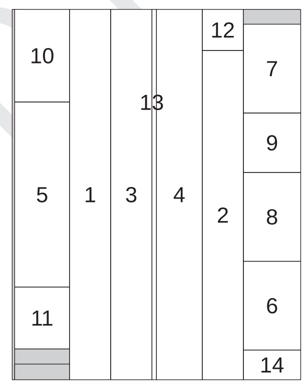

Fig. 17. Best design reached by the proposed CRO approach in the Ba14 UA-FLP.

presented by Kulturel-Konak (2012), the CRO achieved better solutions in all of six problems in which we tested both algorithms. Additionally, our approach was able to obtain better or same solutions than the proposal by Kulturel-Konak and Konak (2011) in 7 out of 10 problems analyzed. Finally, our approach was capable to reach equal or better solutions than the proposal by Palomo-Romero et al. (2017) in 10 out of 11 problems tested.

Considering FBS, the proposed CRO algorithm has been capable to equal or win previous algorithms results in most cases. The CRO has equalized the best result for three problems and has improved the best solution for other eleven UA-FLPs (considering a total of 17 test UA-FLPs). Note that we have demonstrated effectiveness of the suggested CRO algorithm when addressing small problems (it reaches better on all problems which have between 7 and 15 facilities, as Slaughterhouse, Carton Packs, Chopped Plastic, O7, O8, O9, Vc10Ra, Vc10Rs, Ba12, MB12, Ba14, Ma15), solving medium problems (our CRO algorithm achieves better result on problem AB20 and very close result on SC30 which respectively have 20 and 30 facilities), and also, addressing large problems (our CRO algorithm is able to find solutions close to the best ones on problems SC35 and DU62 which respectively have 35 and 62 facilities). In contrast, exclusively in the three problems (SC30, SC35 and DU62) where our approach is not capable to achieve the best solution, the suggested CRO algorithm is able to reach solutions very close to the best known result taken from the references.

Moreover, Table 6 further compares the results reached by the CRO approach and the best known result obtained by other authors in related literature. This way, Table 6 shows the solution with best fitness produced by the suggested CRO approach, the difference (in percentage) between the solution with best fitness reached by the

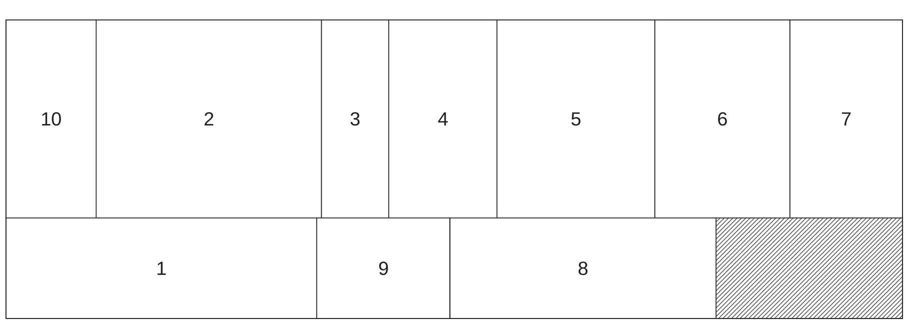

Fig. 15. Best design reached by the proposed CRO approach in the ChoppedPlastic UA-FLP.

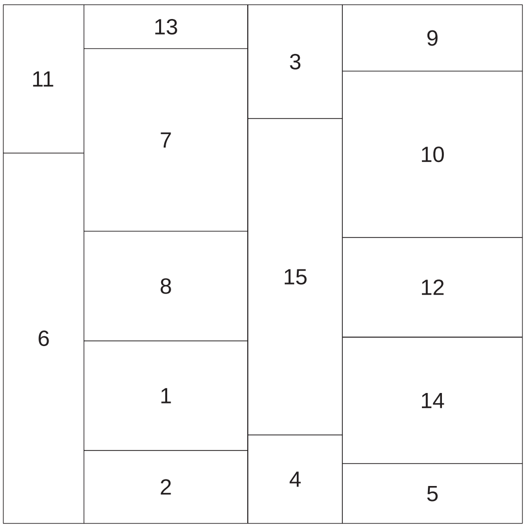

Fig. 18. The best design reach by the proposed CRO approach for ma15a UA-FLP.

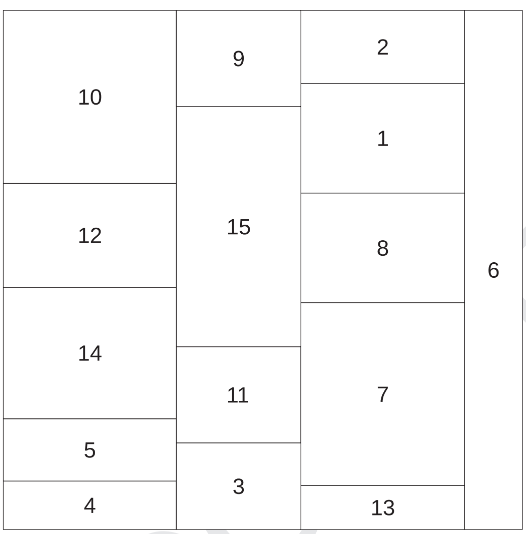

Fig. 19. The best design reach by the proposed CRO approach for ma15s UA-FLP.

CRO approach and the best known FBS result reached by previous works, and finally, this table presents the best facilities designs obtained by our CRO algorithm. In order to complete this table, examples of the facility layout solutions of the problems: Slaughterhouse, CartonPacks, ChoppedPlastic, O9, Ba14, ma15a and ma15s, that were generated by the proposed CRO algorithm and improved substantially the solutions than were reached by previous works, are respectively displayed in Figs. 13–19. These Figures offer the facility layout distribution of instances without empty space (as O9, ma15a and ma15s) and also, with empty space consideration (Ba14). As it was said previously, we have used the same definition of Ba14 that Komarudin and Wong (2010), in their work, it is specified that Ba14 is a problem with 14 facilities and 4 portions of remaining space which each one has an area equal to 0,5.

It is well known that a correct plant layout design can increase efficiency and reduce industrial production costs in a very remarkable way. In this sense, the obtained results contribute to a significant improvement of industrial plants performance.

4. Conclusions

In this work, an evaluation of the performance of applying Coral Reefs Optimization to UA-FLPs considering FBS as representation structure, has been performed. From the best of our knowledge, is it the first time that CRO has been employed to solve UA-FLP. The proposed CRO approach has been applied to 17 UA-FLP instances taken from the related references, and its performance has been analyzed by comparison with different state-of-the-art approaches extracted from recent literature. From the empirical study carried out, we have found that the proposed CRO approach is able to reach or improve the best known results in 14 out of the 17 tested UA-FLPs when considering exclusively FBS representation. Moreover, our suggested proposal reaches or improves the best solution in 7 cases of the 17 tested problems when considering as layout representation both STS and FBS. In the remaining cases, our approach is able to reach solutions with results very close to the best known ones. This fact shows an excellent performance of the CRO algorithm when solving UA-FLPs.

A promising future line of work could be to add some qualitative preferences to the CRO algorithm. Furthermore, this research could be extended in order to take into account the possibility of adding additional considerations as, for example, the inclusion of aisles. Finally, another possible research direction could be to combine alternative methods of layout representation together with CRO for addressing UA-FLPs, and test advanced versions of the CRO approach (Salcedo-Sanz, 2017) in this problem.

Conflict of interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Credit authorship contribution statement

L. Garcia-Hernandez: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing. **L. Salas-Morera:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing. **J.A. Garcia-Hernandez:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Visualization, Writing - review & editing. **S. Salcedo-Sanz:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing. **J. Valente de Oliveira:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing.

Acknowledgments

This work has been partially supported by the project TIN2017-85887-C2-2-P of the Spanish Ministry of Economy, Industry and Competitiveness, Government of Spain. The authors acknowledge the funding by the program to promote research of the University of Córdoba (Spain), under the Research Project XXII PP. Mod. 4.1. The authors would like to thank the program to promote PhD mobility between international research institutions of the Spanish Education Ministry for funding part of this research under the program "José Castillejo2018".

Please cite this article as: L. Garcia-Hernandez, L. Salas-Morera and J.A. Garcia-Hernandez et al., Applying the coral reefs optimization algorithm for solving unequal area facility layout problems, Expert Systems With Applications, https://doi.org/10.1016/j.eswa.2019.07.036

Appendix A. Data set for Slaughterhouse UA-FLP.

This UA-FLP is a real case problem that was planned in the city of Córdoba (Spain). The facility plant dimensions are 30 m × 51.14 m. It was first described by Salas-Morera et al. (1996). Table 7 gives information about the department names, and also, their associated area and aspect ratio constraints. Fig. 20 details material handling flow between the facilities that made up the plant layout.

Appendix B. Data set for CartonPacks UA-FLP.

This UA-FLP is related to a carton recycling plant of 20 m × 14.5 m. It was described by García-Hernández et al. (2015). Briefly, Table 8 offers information about the department names, and also, their associated area and aspect ratio constraint. Fig. 21 details material handling flow between the facilities that made up the plant layout.

Table 7
Facility features for the Slaughterhouse problem.

Id	Facility	Area (m^2)	Aspect ratio
A	Stables	570	4
B	Slaughter	206	4
C	Entrails	150	4
D	Leather & skin	55	4
E	Aeration chamber	114	4
F	Refrigeration chamber	102	4
G	Entrails chamber	36	4
H	Boiler room	26	4
I	Compressor room	46	4
J	Shipping	109	4
K	Offices	80	4
L	Byproduct shipping	40	4

Table 8
Facility features for the CartonPacks problem.

Id	Facility	Area (m^2)	Aspect ratio
A	Raw Material	40	4
B	Finished products	40	4
C	Mechanic	20	4
D	Offices	50	4
E	Staff WC	20	4
F	Expedition	40	4
G	Hydraulic 1	20	4
H	Hydraulic 2	20	4
I	Crushing	20	4
J	Circ. saw	10	4
K	Heat exchange	10	4

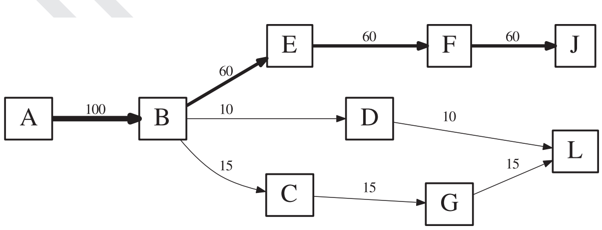

Fig. 20. Material flow requirements for the Slaughterhouse problem.

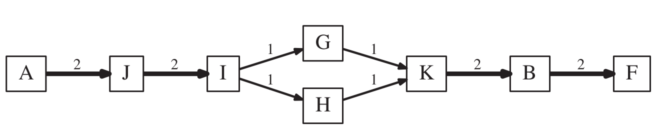

Fig. 21. Material flow requirements for the CartonPacks problem.

Table 9
Facility features for the ChoppedPlastic problem.

Id	Facility	**Area** (m^2)	Aspect ratio
A	Reception	35	4
B	Raw material	50	4
C	Washing	15	4
D	Drying & skin	24	4
E	Chopped	35	4
F	Finished product	30	4
G	Expedition	25	4
I	Office	30	4
J	Toilets	15	4
K	Repair shop	20	4

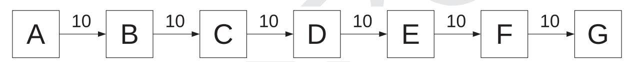

Fig. 22. Material flow requirements for the ChoppedPlastic problem.

Appendix C. Data set for ChoppedPlastic UA-FLP.

This UA-FLP is related to a chopped plastic plant of 30 m × 10 m. It was described by García-Hernández et al. (2013a). Briefly, Table 9 offers information about the department names, and also, their associated area and aspect ratio constraint. Fig. 22 details material handling flow between the facilities that made up the plant layout.

References

Aiello, G., Enea, M., & Galante, G. (2006). A multi-objective approach to facility layout problem by genetic search algorithm and electre method. *Robotics and Computer-Integrated Manufacturing, 22*, 447–455.

Aiello, G., Scalia, G. L., & Enea, M. (2012). A multi objective genetic algorithm for the facility layout problem based upon slicing structure encoding. *Expert Systems with Applications*, (0).

Anjos, M. F., & Vieira, M. V. (2017). Mathematical optimization approaches for facility layout problems: The state-of-the-art and future research directions. *European Journal of Operational Research, 261*(1), 1–16. doi:10.1016/j.ejor.2017.01.049.

Armour, G. C., & Buffa, E. S. (1963). A heuristic algorithm and simulation approach to relative location of facilities. *Management Science, 9*, 294–309.

Azadivar, F., & Wang, J. (2000). Facility layout optimization using simulation and genetic algorithms. *International Journal of Production Research, 38*(17), 4369–4383. doi:10.1080/00207540050205154.

Bazaraa, M. S. (1975). Computerized layout design: A branch and bound approach.. *A I I E Transactions, 7*(4), 432–438.

Bermejo, E., Chica, M., Damas, S., Salcedo-Sanz, S., & Cordón, O. (2018). Coral reef optimization with substrate layers for medical image registration. *Swarm and Evolutionary Computation, 42*, 138–159. doi:10.1016/j.swevo.2018.03.003.

Bozer, Y. A., & Meller, R. D. (1997). A reexamination of the distance-based facility layout problem. *IIE Transactions, 29*(7), 549–560. doi:10.1080/07408179708966365.

Bozer, Y. A., Meller, R. D., & Erlebacher, S. J. (1994). An improvement-type layout algorithm for single and multiple-floor facilities. *Management Science, 40*(7), 918–932.

Camacho-Gómez, C., Wang, X., Pereira, E., Díaz, I., & Salcedo-Sanz, S. (2018). Active vibration control design using the coral reefs optimization with substrate layer algorithm. *Engineering Structures, 157*, 14–26. doi:10.1016/j.engstruct.2017.12.002.

van Camp, D., Carter, M., & Vannelli, A. (1992). A nonlinear optimization approach for solving facility layout problems. *European Journal of Operational Research, 57*(2), 174–189.

Castillo, I., Westerlund, J., Emet, S., & Westerlund, T. (2005). Optimization of block layout design problems with unequal areas: A comparison of MILP and MINLP optimization methods. *Computers and Chemical Engineering, 30*(1), 54–69. doi:10.1016/j.compchemeng.2005.07.012.

Chae, J., & Regan, A. C. (2016). Layout design problems with heterogeneous area constraints. *Computers & Industrial Engineering, 102*, 198–207. doi:10.1016/j.cie.2016.10.016.

Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2007). Facility layout problems: A survey. *Annual Reviews in Control, 31*(2), 255–267.

Dunker, T., Radons, G., & Westkämper, E. (2003). A coevolutionary algorithm for a facility layout problem. *International Journal of Production Research, 41*(15), 3479–3500. doi:10.1080/0020754031000118125.

Durán-Rosal, A. M., Gutiérrez, P. A., Salcedo-Sanz, S., & Hervás-Martínez, C. (2018). A statistically-driven coral reef optimization algorithm for optimal size reduction of time series. *Applied Soft Computing, 63*, 139–153. doi:10.1016/j.asoc.2017.11.037.

Enea, M., Galante, G., & Panascia, E. (2005). The facility layout problem approached using a fuzzy model and a genetic search. *Journal of Intelligent Manufacturing, 16*, 303–316. doi: 10.1007/s10845-005-7025-4.

Ficco, M., Esposito, C., Palmieri, F., & Castiglione, A. (2018). A coral-reefs and game theory-based approach for optimizing elastic cloud resource allocation. *Future Generation Computer Systems, 78*, 343–352. doi:10.1016/j.future.2016.05.025.

García-Hernández, L., Arauzo-Azofra, A., Salas-Morera, L., Pierreval, H., & Corchado, E. (2013a). Recycling plants layout design by means of an interactive genetic algorithm. *Intelligent Automation & Soft Computing, 19*(3), 457–468. doi:10.1080/10798587.2013.803679.

García-Hernández, L., Arauzo-Azofra, A., Salas-Morera, L., Pierreval, H., & Corchado, E. (2015). Facility layout design using a multi-objective interactive genetic algorithm to support the DM. *Expert Systems, 32*(1), 94–107. doi:10.1111/exsy.12064.

García-Hernández, L., Pierreval, H., Salas-Morera, L., & Arauzo-Azofra, A. (2013b). Handling qualitative aspects in unequal area facility layout problem: An interactive genetic algorithm. *Applied Soft Computing, 13*(4), 1718–1727. doi:10.1016/j.asoc.2013.01.003.

García-Hernández, L., Palomo-Romero, J. M., Salas-Morera, L., Arauzo-Azofra, A., & Pierreval, H. (2015). A novel hybrid evolutionary approach for capturing decision maker knowledge into the unequal area facility layout problem. *Expert Systems with Applications, 42*(10), 4697–4708. doi:10.1016/j.eswa.2015.01.037.

Goldberg, D. E., & robert (1985). Alleles, loci, and the traveling salesman problem. In J. J. Grefenstette (Ed.), *Proceedings of the first international conference on genetic algorithms and their applications*. Lawrence Erlbaum Associates, Publishers. (pp. 154–159).

Gomez, A., Fernandez, Q. I., De la Fuente Garcia, D., & Garcia, P. J. (2003). Using genetic algorithms to resolve layout problems in facilities where there are aisles. *International Journal of Production Economics, 84*(3), 271–282.

Gonçalves, J. F., & Resende, M. G. (2015). A biased random-key genetic algorithm for the unequal area facility layout problem. *European Journal of Operational Research, 246*(1), 86–107. doi:10.1016/j.ejor.2015.04.029.

Holland, J. H. (1992). *Adaptation in natural and artificial systems*. Cambridge, MA, USA: MIT Press.

Hosseini-Nasab, H., Fereidouni, S., Fatemi Ghomi, S. M. T., & Fakhrzad, M. B. (2018). Classification of facility layout problems: a review study. *The International Journal of Advanced Manufacturing Technology, 94*(1), 957–977. doi:10.1007/s00170-017-0895-8.

Kang, S., & Chae, J. (2017). Harmony search for the layout design of an unequal area facility. *Expert Systems with Applications, 79*, 269–281. doi:10.1016/j.eswa.2017.02.047.

Komarudin, & Wong, K. Y. (2010). Applying ant system for solving unequal area facility layout problems. *European Journal of Operational Research, 202*(3), 730–746.

Konak, A., Kulturel-Konak, S., Norman, B., & Smith, A. (2006). A new mixed integer formulation for optimal facility layout. *Operation Research Letters, 34*(6), 660–672.

Kouvelis, P., Kurawarwala, A. A., & Gutierrez, G. J. (1992). Algorithms for robust single and multiple period layout planning for manufacturing systems. *European Journal of Operational Research, 63*(2), 287–303.

Kulturel-Konak, S. (2012). A linear programming embedded probabilistic tabu search for the unequal-area facility layout problem with flexible bays. *European Journal of Operational Research, 223*(3), 614–625. doi:10.1016/j.ejor.2012.07.019.

Kulturel-Konak, S., & Konak, A. (2011). A new relaxed flexible bay structure representation and particle swarm optimization for the unequal area facility layout problem. *Engineering Optimization, 43*(12), 1263–1287. doi:10.1080/0305215X.2010.548864.

Liu, J., & Liu, J. (2019). Applying multi-objective ant colony optimization algorithm for solving the unequal area facility layout problems. *Applied Soft Computing, 74*, 167–189. doi:10.1016/j.asoc.2018.10.012.

Liu, Q., & Meller, R. D. (2007). A sequence-pair representation and mip-model-based heuristic for the facility layout problem with rectangular departments. *IIE Transactions, 39*(4), 377–394. doi:10.1080/07408170600844108.

Medeiros, I. G., Xavier, J. C., & Canuto, A. M. P. (2015). Applying the coral reefs optimization algorithm to clustering problems. In *2015 international joint conference on neural networks (IJCNN)* (pp. 1–8). doi:10.1109/IJCNN.2015.7280845.

Meller, R., Narayanan, V., & Vance, P. (1998). Optimal facility layout design. *Operations Research Letters, 23*(3–5), 117–127.

Meller, R. D. (1997). The multi-bay manufacturing facility layout problem. *International Journal of Production Research, 35*(5), 1229–1237. doi:10.1080/002075497195290.

Meller, R. D., & Gau, K.-Y. (1996). The facility layout problem: Recent and emerging trends and perspectives. *Journal of Manufacturing Systems, 15*(5), 351–366. doi:10.1016/0278-6125(96)84198-7.

Montreuil, B. (1991). A modelling framework for integrating layout design and flow network design. *Material handling' 90*. Berlin, Heidelberg: Springer Berlin Heidelberg. (pp. 95–115).

Paes, F. G., Pessoa, A. A., & Vidal, T. (2017). A hybrid genetic algorithm with decomposition phases for the unequal area facility layout problem. *European Journal of Operational Research, 256*(3), 742–756. doi:10.1016/j.ejor.2016.07.022.

Palomo-Romero, J. M., Salas-Morera, L., & García-a-Hernández, L. (2017). An island model genetic algorithm for unequal area facility layout problems. *Expert Systems with Applications, 68*, 151–162. doi:10.1016/j.eswa.2016.10.004.

Purnomo, M. R. A., & Wiwoho, Y. S. (2016). Multi-objective mixed integer programming approach for facility layout design by considering closeness ratings, material handling, and re-layout cost. *IOP Conference Series: Materials Science and Engineering, 105*, 012045. doi:10.1088/1757-899x/105/1/012045.

Salas-Morera, L., Cubero-Atienza, A., & Ayuso-Munoz, R. (1996). Computer-aided plant layout | Distribucion en planta asistida por ordenador. *Informacion Tecnologica, 7*(4), 39–46.

Salcedo-Sanz, S. (2017). A review on the coral reefs optimization algorithm: new development lines and current applications. *Progress in Artificial Intelligence, 6*(1), 1–15. doi:10.1007/s13748-016-0104-2.

Salcedo-Sanz, S., Camacho-Gómez, C., Mallol-Poyato, R., Jiménez-Fernández, S., & DelSer, J. (2016). A novel coral reefs optimization algorithm with substrate layers for optimal battery scheduling optimization in micro-grids. *Soft Computing, 20*(11), 4287–4300.

Salcedo-Sanz, S., Camacho-Gómez, C., Magdaleno, A., Pereira, E., & Lorenzana, A. (2017). Structures vibration control via tuned mass dampers using a co-evolution coral reefs optimization algorithm. *Journal of Sound and Vibration, 393*, 62–75. doi:10.1016/j.jsv.2017.01.019.

Salcedo-Sanz, S., Del Ser, J., Landa-Torres, I., Gil-López, S., & Portilla-Figueras, A. (2013). The coral reefs optimization algorithm: an efficient meta-heuristic for solving hard optimization problems. In *Proceedings of the 15th international conference on applied stochastic models and data analysis (ASMDA2013), Mataró* (pp. 751–758).

Salcedo-Sanz, S., Del Ser, J., Landa-Torres, I., Gil-López, S., & Portilla-Figueras, J. A. (2014a). The coral reefs optimization algorithm: A Novel metaheuristic for efficiently solving optimization problems. *Scientific World Journal, 2014*. doi:10.1155/2014/739768.

Salcedo-Sanz, S., García-Díaz, P., Portilla-Figueras, J., Ser, J. D., & Gil-López, S. (2014b). A coral reefs optimization algorithm for optimal mobile network deployment with electromagnetic pollution control criterion. *Applied Soft Computing, 24*, 239–248. doi:10.1016/j.asoc.2014.07.007.

Salcedo-Sanz, S., Pastor-Sánchez, A., Prieto, L., Blanco-Aguilera, A., & García-Herrera, R. (2014c). Feature selection in wind speed prediction systems based on a hybrid coral reefs optimization - extreme learning machine approach. *Energy Conversion and Management, 87*, 10–18. doi:10.1016/j.enconman.2014.06.041.

Salcedo-Sanz, S., Sanchez-Garcia, J., J. A. , P.-F., Jimenez-Fernandez, S., & Ahmadzadeh, A. (2014d). A coral-reef optimization algorithm for the optimal service distribution problem in mobile radio access networks. *Transactions on Emerging Telecommunications Technologies, 25*(11), 1057–1069. doi:10.1002/ett.2759.

Saraswat, A., Venkatadri, U., & Castillo, I. (2015). A framework for multi-objective facility layout design. *Computers & Industrial Engineering, 90*, 167–176. doi:10.1016/j.cie.2015.09.006.

Scholz, D., Petrick, A., & Domschke, W. (2009). Stats: A slicing tree and tabu search based heuristic for the unequal area facility layout problem. *European Journal of Operational Research, 197*(1), 166–178.

See, P. C., & Wong, K. Y. (2008). Application of ant colony optimisation algorithms in solving facility layout problems formulated as quadratic assignment problems: a review. *International Journal of Industrial and Systems Engineering, 3*(6), 644–672.

Shayan, E., & Chittilappilly, A. (2004). Genetic algorithm for facilities layout problems based on slicing tree structure. *International Journal of Production Research, 42*(19), 4055–4067. doi:10.1080/00207540410001716471.

Sherali, H., Fraticelli, B., & Meller, R. (2003). Enhanced model formulations for optimal facility layout. *Operations Research, 51*, 629–644.

Sikaroudi, A. M. E., & Shahanaghi, K. (2016). Facility layout by collision detection and force exertion heuristics. *Journal of Manufacturing Systems, 41*, 21–30. doi:10.1016/j.jmsy.2016.07.001.

Tam, K. Y. (1992). A simulated annealing algorithm for allocating space to manufacturing cells. *International Journal of Production Research, 30*(1), 63–87. doi:10.1080/00207549208942878.

Tate, D. M., & Smith, A. E. (1995). Unequal area facility layout using genetic search. *IIE Transactions, 27*, 465–472.

Tompkins, J., White, J., Bozer, Y., & Tanchoco, J. (2010). *Facilities Planning* (4th). New York: Wiley.

Tong, X. (1991). *SECOT: A sequential construction technique for facility design*. University of Pittsburg: Doctoral Dissertation.

Ulutas, B. H., & Kulturel-Konak, S. (2012). An artificial immune system based algorithm to solve unequal area facility layout problem. *Expert Systems with Applications, 39*(5), 5384–5395.

Wong, K. Y., & Komarudin (2010). Solving facility layout problems using flexible bay structure representation and ant system algorithm. *Expert Systems with Applications, 37*(7), 5523–5527. doi:10.1016/j.eswa.2009.12.080.

Wu, Y., & Appleton, E. (2002). The optimisation of block layout and aisle structure by a genetic algorithm. *Computer Industrial Engineering, 41*(4), 371–387.

Yan, C., Ma, J., Luo, H., & Patel, A. (2019). Hybrid binary coral reefs optimization algorithm with simulated annealing for feature selection in high-dimensional biomedical datasets. *Chemometrics and Intelligent Laboratory Systems, 184*, 102–111. doi:10.1016/j.chemolab.2018.11.010.

Yang, Z., Zhang, T., & Zhang, D. (2016). A novel algorithm with differential evolution and coral reef optimization for extreme learning machine training. *Cognitive Neurodynamics, 10*(1), 73–83. doi:10.1007/s11571-015-9358-9.