Optimal Location and Sizing of Energy Storage Systems in DC-Electrified Railway Lines Using a Coral Reefs Optimization Algorithm with Substrate Layers

David Roch-Dupré [1], Carlos Camacho-Gómez [2], Asunción P. Cucala [3], Silvia Jiménez-Fernández [4], Álvaro López-López [3], Antonio Portilla-Figueras [4], Ramón R. Pecharromán [3], Antonio Fernández-Cardador [3] and Sancho Salcedo-Sanz [4,*]

1   Faculty of Economics and Business Administration, Institute for Research in Technology, Comillas Pontifical University, 28049 Madrid, Spain; droch@iit.comillas.edu
2   Department of Computer Systems, Universidad Politécnica de Madrid, 28031 Madrid, Spain; carlos.camacho@upm.es
3   Institute for Research in Technology, Comillas Pontifical University, 28049 Madrid, Spain; Paloma.cucala@iit.comillas.edu (A.P.C.); alvaro.lopez@iit.comillas.edu (Á.L.-L.); ramon@comillas.edu (R.R.P.); antonio.fernandez@iit.comillas.edu (A.F.-C.)
4   Department of Signal Processing and Communications, Universidad de Alcalá, 28805 Alcalá de Henares, Spain; silvia.jimenez@uah.es (S.J.-F.); antonio.portilla@uah.es (A.P.-F.)
*   Correspondence: sancho.salcedo@uah.es; Tel.: +34-91-885-67-31

Abstract: This paper deals with the problem of finding the optimal location and sizing of Energy Storage Systems in DC-electrified railway lines. These devices increment the use of the regenerated energy produced by the trains in the braking phases, as they store the energy to later provide to the catenary the excess of regenerated energy, that otherwise would be lost in the rheostats. However, these infrastructures require a high initial investment that, in some cases, may question their profitability. We propose a multi-method ensemble meta-heuristic to obtain the optimal solution to the problem, with a high level of accuracy. Specifically, the Coral Reefs Optimization with Substrate Layers (CRO-SL) is proposed, an evolutionary-type approach able to run different search procedures within the same population. We will evaluate the performance of the CRO-SL in the problem, and we will show that it performs better than the best known existing meta-heuristics for this problem.

Keywords: railway lines; Energy Storage Systems; multi-method ensembles; CRO-SL algorithm; meta-heuristics

1. Introduction

In the fight against climate change, there is a need for urban DC-electrified railway systems (such as tramway or metro systems) to become more energy efficient. Indeed, the authors of [1] state that energy efficiency goals are one of the main drivers for the future evolution of transportation systems. Among the possible ways to achieve it, the installation of electrical infrastructure improvements, such as Reversible Substations (RSs) [2–6] or wayside Energy Storage Systems (ESSs) [2,4,7–9], are within the measures with the highest energy saving potentials. These measures increment the use of the regenerated energy produced by the trains in the braking phases, as they give back to the utility grid (in the case of the RSs) or store to later provide to the catenary (in the case of the ESSs) the excess of regenerated energy, that otherwise would be lost in the rheostats. However, these infrastructures require a high initial investment that, in some cases, may question their profitability. Due to this, railway operators must use accurate methodologies to study in detail the impact that the installation of these improvements has on the system's energy efficiency, as well as their associated profitability before taking the final decision on whether or not undertaking the installation. These methodologies must be based on the application

of optimization algorithms, which determine the optimum installation in combination with accurate railway simulation models (in charge of assessing the energy saving obtained with the installation of the improvements). The high flexibility and great capacity to successfully deal with the computationally intensive and highly nonlinear and non-convex problems posed by the railway simulations, makes nature-inspired optimization algorithms a very good choice to tackle the optimization of the installation of infrastructure improvements. Nevertheless, there are few examples in the literature that combine the use of this type of algorithms with the application of railway simulation models. Among them, we can find authors that focus on the optimization of the RSs installation [10–12] or ESSs [13–15].

However, all these previous examples use low accurate railway simulation models, especially from the point of view of the traffic modeling. As stated in [16], the traffic model has a great impact when analyzing the receptivity (ability to re-use the regenerated energy) of a line and, in general terms, the efficiency. Thus, an oversimplified traffic model consisting of a single traffic timetable with constant dwell times can lead to significant errors in the computation of the energy saving figures and, consequently, to an incorrect decision-making. In order to avoid it, this paper continues the line of research proposed in [16,17] and makes use of very realistic traffic models that allow to generate a large number of representative and realistic scenarios, so that the information provided by the optimization algorithms is accurate enough to make good decisions about the optimum installation of RSs or ESSs.

In addition, even in sporadic cases where accurate and realistic models have been used in combination with nature-inspired optimization algorithms, such as in [18], the algorithms employed may have problems of rapid convergence to local optima (where they get trapped, being unable to achieve the global optimum). This can be accentuated in cases of very large and demanding search spaces, as it happens when the granularity of the decisions' variables increases. The higher the granularity of the power and capacity sizes to be installed (this is, the higher amount of possible power and capacity sizes available), the more demanding the search space is. This can be a problem for railway infrastructure administrators, because if the optimization algorithm used cannot deal with the desired level of granularity on the power or capacity sizes, the decisions about the infrastructure improvements can be oversimplified, leading to non-optimal solutions. For this reason, this paper goes a step further from all the previous research in this field and proposes the application of a multi-method ensemble based on an evolutionary-type approach, the Coral-reef Optimization with Substrate Layer (CRO-SL) algorithm. This algorithm is able to outperform the algorithms from previous research when the granularity of the decision variables is increased and, therefore, the search space becomes much more demanding.

In the last decades, a large number of meta-heuristics have been developed and applied in many real-world problems [19–21]. According to the No-Free-Lunch theorem, there is not a single algorithm that outperforms the rest in all possible optimization problems. Thus, the process of designing, testing, and comparing different methods can be time-consuming or even lead to poor performances. To avoid this point, the authors of [22] present many advises when tackling real problems with meta-heuristics. For that matter, in practice the number of problems that one algorithm can successfully solve is finite; thus, one of the recent research lines consists in hybridizing different existing methods into ensembles [23] in order to reach a better performance than the original algorithms on their own, and to provide a versatile method that can be able to perform well in different problems. There are two main ways to build an ensemble evolutionary method (EEM): mixing parts, also called agents, from different algorithms (low-level multi-method), or mixing entire algorithms into one (high-level multi-method). In general, the agent represents a search method acting over one or more populations, where the individuals compete or cooperate in the optimization task, but an agent may also represent different encodings, parameter configurations, or constraints. Many examples of EEM can be found in literature such as in [24], where two variants of the well-known L-SHADE algorithm were alternated over the population, as each one on their own produces stagnation. This ensemble approach

was tested in CEC 2014 and CEC 2017 benchmark functions. Furthermore, the work in [25] introduces a memetic algorithm where a Tabu Search process was used as local search and the global search was performed by the Teaching–Learning-Based Optimization algorithm, showing competitive results in large graph coloring problems. In [26], the authors propose a cooperative framework, where Differential Evolution (DE) and Quantum Evolutionary search methods were combined in order to enhance the convergence speed as well as the diversity of the population, for large-scale global optimization problems. In [27], a hybrid multi-objective methodology is proposed, where the leader direction of the Whale Algorithm is set by a competition among two DE crossovers. Finally, more EEMs can be found in the review given in [28]. In this case, the CRO-SL is a novel multi-method ensemble evolutionary-based approach, in which different search methods are applied to the individuals of a single population. This algorithm was first proposed in [29], and further described in [30]. It is based on a basic version of the CRO meta-heuristic proposed in [31] and successfully applied to different optimization problems [32,33]. The CRO-SL has been further applied to a large number alternative optimization problems, in different engineering fields [34–37]. The main advantage of the CRO-SL is the cooperation of many global and local search processes over the same population, which allows a balance between exploration and exploitation, and avoids stagnation in local optima. Furthermore, this algorithm also encourages diversity in the population, by keeping hollows in the reef (empty positions) for new offspring.

The contributions of this paper are twofold:

- The main contribution of this work is to apply the CRO-SL in a very realistic railway simulation model to determine the optimum installation of ESSs in a railway line, increasing the number of decision variables (therefore, a more demanding search space for the algorithm) when compared to previous works [18].
- From an algorithmic point of view, a novel cooperative multi-agent ensemble method, the CRO-SL, has been successfully applied to a hard combinatorial optimization problem in the field of ESS design for transportation systems.

The results obtained are then compared to those provided by the algorithm with best results in a similar problem [18], to show the best performance of the CRO-SL and its great potential when dealing with very demanding search spaces.

The remainder of this paper has been structured as follows. The next section presents the problem definition, with details on the problem's encoding, fitness function proposed, and its calculation using simulation software. Section 3 presents the CRO-SL, its main characteristics, and the substrates (search procedures) considered in this paper. Section 4 presents the experimental part of the paper, where we have illustrated the good performance of the CRO-SL in two different case of study with an extremely high search space. Section 5 closes the paper by giving some final conclusions and remarks on the research carried out in this work.

2. Problem Definition

2.1. Formulation

The optimization proposed in this paper consists of determining the best configuration of Energy Storage Systems (ESSs) to be installed in a railway line, in terms of the following.

- Number of ESSs to be installed.
- Location of the ESSs to be installed.
- Power (kW) of each ESS to be installed.
- Capacity (kWh) of each ESS to be installed.

Note that ESSs have been selected instead of RSs because they are more demanding from the point of view of the decision making (the capacity is not a variable of decision in the case of the RSs). However, the application of the proposed optimization to the RSs is straightforward.

For the particular case of this optimization problem, each individual (potential solution to the problem), **x**, from the population is represented by a 2D matrix. The rows of the matrix correspond to the features of the installation. As the infrastructure improvements selected for the optimization are the ESSs, there are two rows: the first one associated with the power and the second one with the capacity. The columns of the matrix correspond to the potential locations where the ESS can be installed. Therefore, the dimension of the matrix that represents each individual i is $[2 \times N]$, and N stands for the number of potential locations where the ESS can be installed. In a visual way, the structure of **x** is given by Equation (1).

$$\mathbf{x_i} = \left[\begin{array}{cccc} \mathcal{P}_1 & \mathcal{P}_2 & \dots & \mathcal{P}_N \\ \mathcal{C}_1 & \mathcal{C}_2 & \dots & \mathcal{C}_N \end{array} \right] \tag{1}$$

where

- $\mathcal{P}_n$ stands for the power feature of $\mathbf{x_i}$ at location n. The values for this variable are discrete, ranging from 0 kW (no ESS installed at position n) to $\mathcal{P}^{max}$ kW (maximum power that can be installed) in steps of $\mathcal{S}_\mathcal{P}$ kW. Therefore, each power feature must take one value from the set of $\dfrac{\mathcal{P}^{max}}{\mathcal{S}_\mathcal{P}}$ discrete values available (being $\mathcal{P}^{max}$) a multiple of $\mathcal{S}_\mathcal{P}$).
- $\mathcal{C}_n$ stands for the capacity feature of $\mathbf{x_i}$ at location n. The values for this variable are discrete, ranging from 0 kWh (no ESS installed at position n) to $\mathcal{C}^{max}$ kWh (maximum capacity that can be installed) in steps of $\mathcal{S}_\mathcal{C}$ kWh. Therefore, each capacity feature must take one value from the set of $\dfrac{\mathcal{C}^{max}}{\mathcal{S}_\mathcal{C}}$ discrete values available (being $\mathcal{C}^{max}$) a multiple of $\mathcal{S}_\mathcal{C}$).

Although the optimization algorithms will treat each feature (power and capacity) independently, there must be a coherency between the power and capacity values installed at the same location, which means that there must be an element-to-element correspondence between the values from the different rows that share the same column. Specifically, in case of selecting a non-zero value for the power at a certain location, the value of the capacity must also be non-zero, and vice versa. The same happens in case of selecting a zero value (no installation) for power or capacity at a certain location. This restriction will be named "constraint of coherency among features", is formulated in Equation (2) and must be applied in all optimization algorithms used.

$$\mathcal{P}_n = 0 \longleftrightarrow \mathcal{C}_n = 0 \tag{2}$$

2.2. Fitness Function

The optimal solution that the optimization algorithms must find is the one that yields to the highest installation's Net Present Value (NPV). Therefore, the fitness function for a given individual **x** is the NPV associated with its ESSs configuration, given by (3).

$$NPV(\mathbf{x}) = \sum_{t=1}^{T} \frac{\left[E_\mathcal{R}^A - E_\mathcal{E}^A(\mathbf{x}) \right] \cdot e_{cost}}{\left[(1 + \mathcal{W}_{acc})^t \right]} - C_0(\mathbf{x}) \tag{3}$$

$$\textbf{s.t.} \quad C_0(\mathbf{x}) \leq M_b$$

where:

- $E_\mathcal{R}^A$ is the annual energy consumption obtained when the infrastructure is not improved. This value is obtained from simulation (the details about the communication between optimization algorithms and railway simulations are given in Section 2.3).
- $E_\mathcal{E}^A(\mathbf{x})$ is the annual energy consumption when installing the ESSs configuration associated with **x**. This value is also obtained from simulation.
- e_{cost} is the energy price.
- $C_0(\mathbf{x})$ is the installation cost of the ESSs configuration determined by **x**.
- $\mathcal{W}_{acc}$ is the Weighted Average Cost of Capital.

- T is the period to evaluate the investment (in years).
- M_b is the maximum budget.

The reason why the NPV has been selected for the fitness function is that it is capable of finding a balance between the energy saving (the main target from the environmental point of view) and the installations costs (very important in the final decision about installing or not the infrastructure improvements). On the one hand, the positive cash flows of the NPV are directly related with the energy saving. Particularly, the energy saving associated with individual **x**) is obtained from comparing the energy consumption with and without the ESSs installation ($E_{\mathcal{R}}^A - E_{\mathcal{E}}^A(\mathbf{x})$), and it is transformed into economic cash flows by multiplying it by e_{cost}. On the other hand, the installation costs represent the negative cash flows of the NPV.

The investment will be economically profitable if the NPV is positive in the period T.

2.3. Middleware between Optimization Algorithms and Railway Simulations

As previously stated, the optimization algorithms need information from the railway simulations in order to compute the fitness function. The description of the railway simulator used in this research is out of the scope of this paper, but it must be noted that it thoroughly detailed in [16,17,38], both from the traffic and electrical modeling points of view. The traffic modeling and simulation include use of different headways for the different time bands (the headway being the time interval between consecutive trains), use of different Automatic Train Operation (ATO) speed profiles, variability in the dwell times of the short-turn and terminal stations, use of realistic dwell times in passenger stations obtained from stochastic models and implementation of a centralized traffic regulation model that automatically manages traffic operation on metro lines. For more details about the traffic models implemented in the simulator, see [16,17].

The electrical modeling and simulation are based on the generation and resolution of the electrical scenarios associated with the traffic scenarios given by the traffic model. A time step of 1 second is used. Iterative techniques have been applied to solve the load flow problem associated with each electrical scenario. The iterative technique selected has been the unified-mixed AC-DC Newton–Raphson method. For more details about the electrical modeling and simulation, see in [38].

The middleware between the optimization algorithms and the railway simulations is depicted in Figure 1 and enumerated below. This process is repeated in a kind of feedback loop, until achieving the optimum solution.

1. The algorithm gives the characteristics of the ESSs configurations to be tested (in terms of number, location, power, and capacity) to the railway simulation module. This information becomes part of the electrical infrastructure data, included within the data required to perform the simulations.
2. Several traffic scenarios (selected to represent the traffic operation in a realistic way) are simulated with the ESSs configuration provided by the algorithm. The output from the simulations is the energy saving associated to each configuration. This information is, at the same time, the input data for the optimization algorithm.
3. The value for the fitness function of each configuration is computed from its associated energy saving and installation costs (see Equation (3)). This fitness value provides the information required by the optimization algorithms to update the configurations to be tested by the simulator in the next iteration.
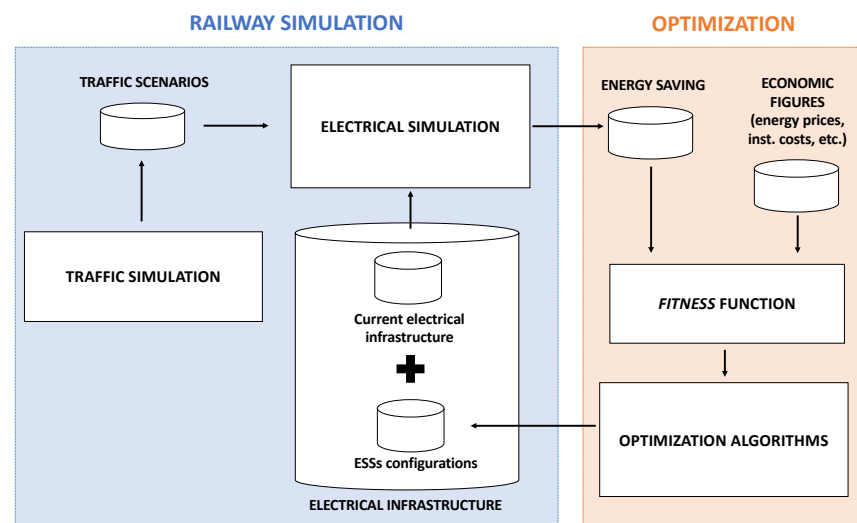
Figure 1. Middleware communicating the railway simulator and the optimization algorithm.

3. Methods

In this Section, the optimization algorithms considered in this work are described. First, the Coral Reef Optimization (CRO) algorithm, the basic form of the algorithm, is explained. Then, the multi-method ensemble algorithm defined on top of the CRO, Coral Reef Optimization with Substrate Layers (CRO-SL) algorithm, as well as search methods implemented in the substrate layers are detailed. Finally, the Genetic Algorithm (GA) used for comparison purposes is also described in this section.

3.1. The Coral Reef Optimization Algorithm for Optimization

The Coral Reef Optimization Algorithm (CRO) was recently proposed by Salcedo et al. in [30,31]. This approach is a kind of evolutionary-type algorithm which imitates the evolution of coral reefs and the different processes occurring in these ecosystems. Mathematically, the algorithm considers a square model for the reef (Λ) of size $N \times M$. 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 optimization problem. Once we have modeled the reef and the corals, the algorithm's processes are defined using the following steps: After the reef initialization described above, the phase of reef formation is artificially simulated. This phase consists of κ iterations: at each of such iterations the corals' reproduction in the reef is emulated by applying different operators and processes as described in Algorithm 1—a modeling of corals' sexual reproduction (broadcast spawning and brooding) and asexual reproduction.

After the reproduction stage, the set of formed larvae (newly produced solutions to the problem) attempts to find a place on the reef to develop and further reproduce. This deployment may occur in a free space inside the reef (hole), or in an occupied location (by fighting against the coral currently settled in that place). If larvae are not successful in locating a place to settle after a number of attempts, they are considered as predated by animals in the reef, and so that solution is discarded from the reef. The coral builds a new reef layer in every iteration. Note that the CRO approach is an evolutionary-type approach with some characteristics from Simulated Annealing, as the number of holes in the reef allows that poorly fitted solutions are considered, yet allowing the search in other parts of the search space and not focusing on local optima. This process of considering poorly fitted solutions is more intense in the first stages of the algorithm. Figure 2 shows a flowchart of the CRO algorithm's dynamics.

Algorithm 1 Pseudocode for the CRO algorithm.

Require: Valid values for the parameters controlling the CRO algorithm
Ensure: A single feasible individual with optimal value of its *fitness*
 1: Initialize the algorithm
 2: **for** each iteration of the simulation **do**
 3:     Update values of influential variables: Depredation probability, etc.
 4:     Sexual reproduction processes (broadcast spawning and brooding)
 5:     Asexual reproduction process
 6:     Settlement of new corals
 7:     Depredation process
 8:     Evaluate the new population in the coral reef
 9: **end for**
10: Return the best individual (final solution) from the reef

---
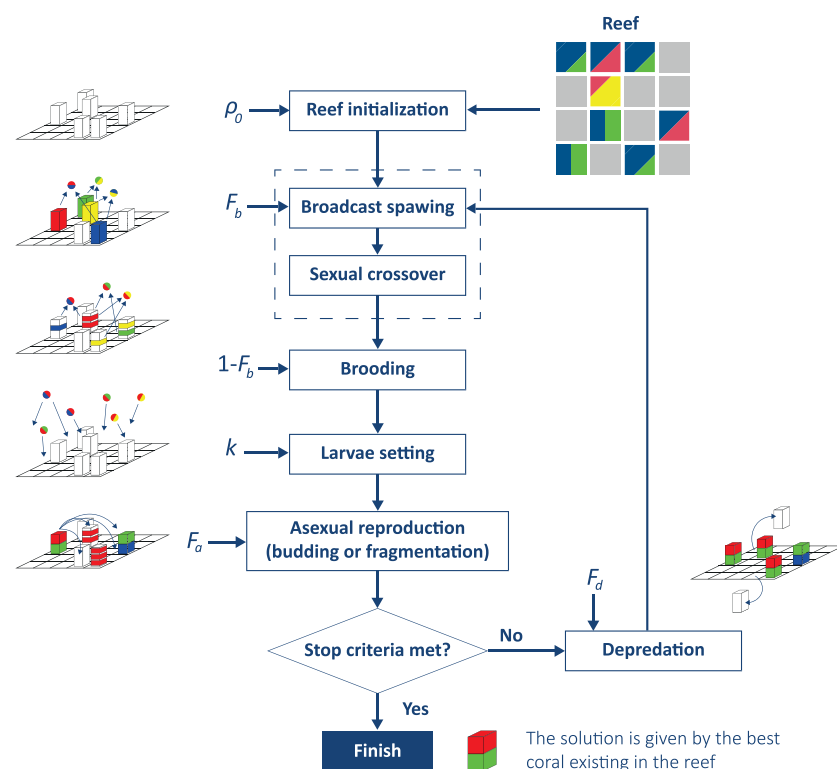
Figure 2. Flowchart of the CRO dynamics.

3.2. The CRO-SL: A Multi-Method Ensemble Evolutionary Algorithm

The concept of *ensemble* in optimization refers to a combination of algorithms or parts of them to better deal with optimization problems [23]. The aim of this idea is that the new strategy can lead to better performance than any single strategy on its own in the same problem. In [23], the authors present a classification of ensembles attending to the type of the constituent agents and its implementation: when the agents are different types of search strategies, operators or processes, etc., it is classified as a *low-level ensemble*; on the other hand, *high-level ensembles* refer to the methods which combine complete algorithms to select the best approach for a given problem.

The CRO-SL approach is a recently proposed low-level multi-method ensemble, based on the CRO algorithm defined above. Specifically, in the CRO-SL, a set of "substrates" are defined in the CRO reef [39]. At each substrate, a different search operator is applied to the corals for the formation of the offspring. Likewise, the CRO-SL introduces novel forms to maintain the diversity of the reef and promotes the cooperation of different search operators within a single population.

Figure 3 presents a flowchart that describes the main characteristics of the CRO-SL ensemble. Note that the algorithm's structure is similar to that of the basic CRO (see Figure 2), but the reefs structure is different. In this case, the reef is divided into different "substrates", each one representing a search procedure. The corals in each different substrate will be evolved by applying the corresponding search procedure assigned to that substrate, i.e., Gaussian Mutation, Differential Evolution, Two-points crossover, etc. This approach can be seen as a multi-method ensemble, where the different search procedures in each substrate cooperate in order to obtain the best solution to the optimization problem at hand. See in [37] for further details on the proposed algorithm.
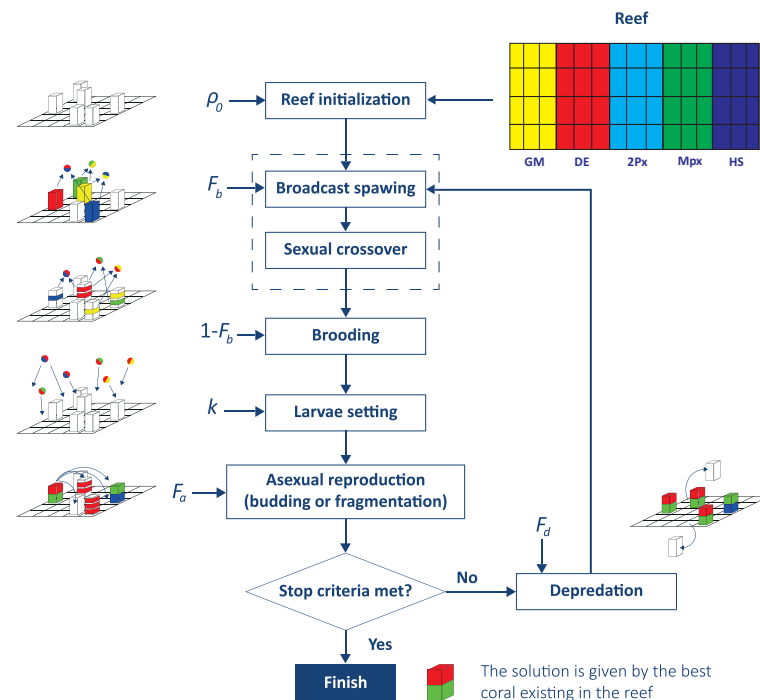

Figure 3. Flowchart of the CRO-SL dynamics.

The CRO-SL was first introduced in [29], and further developed in [39]. As previously summarized, the CRO-SL algorithm is a general ensemble approach which promotes cooperative evolution, where each substrate layer represents different processes, in this case, different search space techniques. The CRO-SL has shown previous considerable results in a large number of hard optimization problems in engineering, such as Micro-grids design [34,35], antennas design and vibration control in civil structures [40,41], medical image processing [42], or data clustering [36], among others.

Substrate Layers Defined in the CRO-SL

Very different search methods can be defined in the CRO-SL as agents (substrates) for the ensemble approach. They are usually defined at the practitioner's discretion. In previous approaches, it has been shown that combinations of well-known traditional operators with novel search techniques provide the best set of techniques (substrates) in the CRO-SL. In this paper, the operators considered to tackle this problem are the same as used in previous applications of the CRO-SL, such as Harmony Search (HS), Differential Evolution (DE), classical two-points crossover (2Px), classical multi-points crossover (MPx), and Gaussian-based mutation (GM).

1. HS: A metaheuristic method based on stochastic optimization [43]. It imitates the process found in music improvisation which searches for better harmony. There are two parameters that determine the way in which new larvae are generated: (i) Harmony Memory Considering Rate (HMCR), which ranges from 0 to 1. If a uniformly spawned value is above the value of HMCR, then the encoded parameter

value is uniformly drawn from the values in the coral, (ii) Pitch Adjusting Rate (PAR) which ranges from 0 to 1, which sets the probability of choosing a neighbor value of the current larva.

2. DE: An Evolutionary Algorithm (EA), which has good abilities for global search [44]. The new larvae can be generated either by the mutation or crossover process. For a randomly selected encoded parameter, if a uniformly generated value is above the Crossover Probability (CR) value, which ranges from 0 to 1, the new value is obtained by $x'_i = x_i^1 + F(x_i^2 - x_i^3)$, in which F is the evolution factor. Otherwise, the value is crossed with the randomly selected encoded parameter.

3. 2Px: The crossover operator is the most used exploration mechanism in genetic and evolutionary algorithms [45], as its combination with an efficient mutation process allows achieving a suitable balance between exploration and exploitation. 2Px selects two random parents and exchanges the genetic material in-between two random points on them. Despite each substrate is linked to a searching process, when another parent must be picked, the selection is not limited to their substrate, but it can be chosen from any part of the population instead. The reason is to contribute to genetic information exchange among substrates, so they can easily cooperate.

4. MPx: This search method is a generalization of the 2Px. In this case, k points are selected in the parents. In this work, due to the dimensionality of the problem, the value of k has been chosen to be 3. Thus, a binary vector decides whether the parts of each parent are exchanged or not for the new offspring generation.

5. GM: The Gaussian Mutation is a noisy search method based on adding random values from the Gaussian distribution to the encoded parameter values, thus generating an offspring. The standard deviation σ value in this work is linearly decreasing during the run, from $0.2 \cdot (A - B)$ to $0.02 \cdot (A - B)$, where $[B, A]$ is the domain search. The Gaussian probability density function is

$$f_{G(0,\sigma^2)}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}.$$

With the aim of exploring the search space at the beginning of the optimization process and exploiting it at the end, the parameter σ is adapted during the simulation.

Note that we have considered a version of the CRO-SL where the parameters of the algorithm such as the reef size, probabilities of reproduction, or substrates size are fixed at the beginning of the algorithm and they are not changed during the CRO-SL run. A dynamic version of this algorithm is possible, where its parameters are dynamically updated during the search process.

3.3. Genetic Algorithm (GA)

As explained in the introduction section, different nature-inspired optimization algorithms were used in [18], in a similar—but more simplified—problem to determine the optimal installation of ESSs in railway lines: the Genetic Algorithm (GA), the Particle Swarm Optimization Algorithm (PSO), and the Fireworks Algorithm (FA). The best performance among the three optimization algorithms was by the GA. Particularly, the rate of convergence of the GA was 100% (achieving the optimum solution in the 20 instances run), while the rate of convergence of the PSO was only 65% (achieving the optimum solution in 13 of the 20 instances run). Additionally, the GA demonstrated to be faster than the FA, requiring less iterations to achieve the optimum. Therefore, in this paper we have used the GA as the algorithm to be compared against the CRO-SL.

The main characteristics of the GA proposed by [18] are below. For the sake of clarity, the general nomenclature used in Section 2.1 will be adapted to the usual nomenclature of the GA: the individuals will be referred to as chromosomes, the power features will be referred to as power genes, and the capacity features will be referred to as capacity genes.

- **Selection**: the *tournament selection* has been selected. A pair of chromosomes is randomly selected from the previous iteration's population (or from the initial population

in case of the first iteration). The fitness of both chromosomes is compared and the chromosome with the highest fitness is chosen for the next generation. The same chromosome can be selected more than once and the procedure is repeated until obtaining a new population with the same size as the previous one.

- **Crossover**: the one-point crossover mechanism has been selected. In this variant, chromosomes are randomly selected in pairs, as well as a single crossover point for each pair. The part of the chromosome after the crossover point is exchanged between the pair of chromosomes (this exchange includes both power and capacity genes).
- **Mutation**: each chromosome can modify a maximum number of 2 power and capacity genes (to comply with the constraint of coherency among features, the power and capacity genes mutated must be in the same position, or what is the same, must share the column in the matrix of (1)). The probability for a chromosome to mutate 1 power and capacity gen (randomly selected) is 20% and to mutate 2 power and capacity genes (randomly selected too) is 10%. These mutations consist of adding or subtracting $\mathcal{S}_{\mathcal{P}}$ or $\mathcal{S}_{\mathcal{C}}$ (each possibility with a 50% probability) to, respectively, the power and capacity genes randomly selected for the mutation.

4. Experiments and Results

4.1. Case Study Characteristics

A real Spanish metro line has been used for the case study, but with modifications to the topology to increase the difficulty of the decisions that the optimization algorithms must take to improve the infrastructure. The main characteristics of this topology are depicted by Figure 4. As can be seen, it is a Y-shaped line with short-turn and terminal stations in the branches and a terminal station in the common section. Regarding the passenger stations, there are 12 in the common section, 12 in the longest branch (branch 1), and 9 in the shortest branch (branch 2). Regarding the train services, there are four different types: all of them change from down to up direction in the terminal station of the common section and each one changes from up to down direction in one of the short-turn or terminal stations located in the branches (each type of service in a different one).
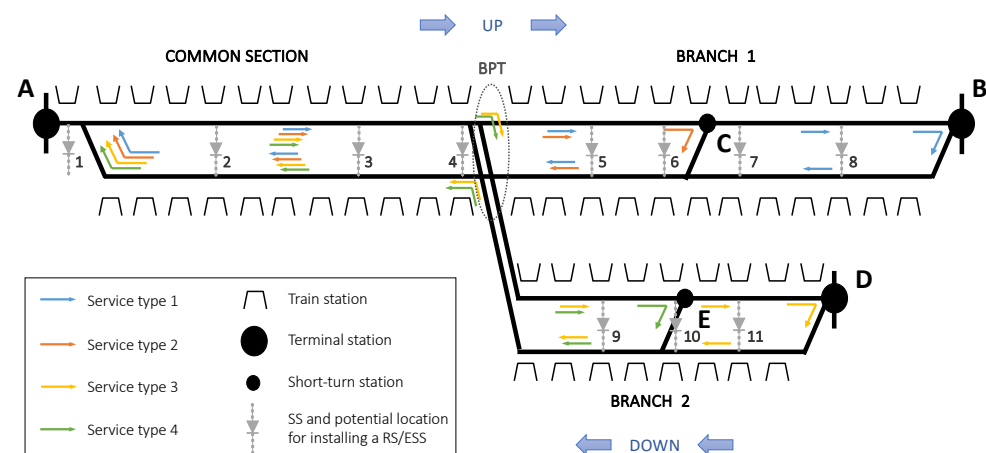

Figure 4. Case study line's topology.

The main electrical characteristics of the case study are given below.

- **Infrastructure**: 11 traction substations (SSs): 4 SSs of 6.6 MVA in the common section, 4 SSs of 4.4 MVA in the longest branch, and 3 SSs of 4.4 MVA in the shortest one. The nominal voltage is 1600 V and the no-load voltage is 1650 V. The total impedance of the active plus return circuit is 26 mΩ/km.
- **Rolling stock**: Automatic Train Operation-guided trains with a maximum motoring power of 5 MW, a maximum regenerating power of 4 MW and an auxiliary consumption of 200 kW. They incorporate a regenerative braking system (pneumatic braking is only used when the electrical braking is not able to provide the braking force

commanded) and the voltage threshold for the activation of the rheostatic braking is 1800 V.

- **ESSs storage technology**: electrochemical double-layer capacitors (EDLC) have been chosen as they have an excellent balance between power and energy density. The ESS management control is determined by the control curve described in [18]. This control curve tries to maximize the availability of the ESS to store regenerated energy by setting the activation values of the charging and discharging phases very close to the no-load (V_{NL}). In fact, these values are, respectively, $1.01 \times V_{NL}$ and $0.99 \times V_{NL}$.

Regarding the traffic operation, different moments of operation have been considered:

- Very peak-hour with large perturbations: it is represented by 200 different traffic scenarios with 3.5 min headway. All the traffic scenarios have been generated with the realistic traffic model for large perturbations of [17].
- Very peak-hour/peak-hour/off peak-hour/sparse traffic operation with small perturbations. Each moment of operation has an associated headway of, respectively, 3.5/5/7/15 min and is represented by 200 different traffic scenarios, generated all of them with the realistic traffic model for small perturbations of [16].

Note that each traffic scenario previously mentioned represents the operation during a long enough period. According to the work in [16], to represent a typical operation cycle, the time length for each scenario must be approximately the time required by a train to go from one terminal station to the other. As trains can give different types of service for the case study line, the simulation length has been defined as the time required by the longest service to make its itinerary in one direction (up or down direction), including the time spent before departing in the other direction. This time varies from 3700 s to 6400 s depending on the headway value.

The distribution of the annual hours of operation with each headway and perturbation type is depicted in Table 1 (operation is from 6:00 a.m. to 2:00 + 1 a.m., while maintenance is from 2:00 a.m. to 6:00 a.m.).

Table 1. Annual operation hours of the line under study.

	Annual Operation Hours	**Percentage of Total**
3.5 min with large perturbations	942.5	(12.95% of total)
3.5 min with small perturbations	942.5	(12.95% of total)
5 min with small perturbations	1885.0	(25.90% of total)
7 min with small perturbations	2782.0	(38.20% of total)
15 min with small perturbations	728.0	(10.00% of total)

Additionally, the train load varies depending on the value of the headway: with 3.5 and 5 min headway (very peak hour and peak hour) the train load is 90% of the maximum load, with 7 min headway (off peak-hour) the train load is 50% of the maximum load and with 15 min headway (sparse traffic conditions) the train load is 25% of the maximum load.

4.2. Search Space

Decision variables for both optimization algorithms are as follows:

- Locations: Every SS location in the line's case study is considered as candidate where an ESS could be installed, therefore $N = 11$.
- Power to be installed at each location: this decision variable can take any value from the set $[0 : \mathcal{S}_\mathcal{P} : \mathcal{P}^{max}]$, where the power step is $\mathcal{S}_\mathcal{P} = 500$ kW and the maximum power that can be installed is $\mathcal{P}^{max} = 3000$ kW.
- Capacity to be installed in each location: this decision variable can take any value from the set $[0 : \mathcal{S}_\mathcal{C} : \mathcal{C}^{max}]$, where the capacity step is $\mathcal{S}_\mathcal{C} = 2.5$ kWh and the maximum capacity that can be installed is $\mathcal{C}^{max} = 15$ kWh.

Note that the search space used in this optimization process is far bigger than the search space considered in [18], where the optimum ESSs installation in the same line was determined using three different nature-inspired optimization algorithms: Genetic Algorithm (GA), Particle Swarm Optimization algorithm (PSO), and Fireworks Algorithm (FA). In that paper, the best performance, in terms of effectiveness, speed of convergence, and robustness, was given by the GA, and the only decision variables were the locations of the ESSs and the power to be installed. In order to determine the good performance of the CRO, this algorithm was first applied to the original search space used in [18], achieving always the same solution as the one given by the GA (which always performed better than the PSO an FA algorithms).

In the present work, five instances have been run for each optimization algorithm (GA and CRO-SL) in the new—and far more demanding—search space. With the aim to speed up the simulation time and reduce the computational burden associated with the high number of electrical simulations that must be performed during the optimization process (each individual has an associated ESSs configuration to be simulated in each iteration), parallel computing has been used. Therefore, the configurations to be simulated at each iteration have been distributed among a number of workers equivalent to the number of logical processors of the server, whose properties are given below.

- CPU: Intel(R) Xeon(R) Silver 4116, 24 Cores-2100 MHz (48 logical processors).
- RAM: 128 GB.
- Disk: DELL PERC H330 1.65 TB.

The simulation time required to complete a single iteration of any optimization algorithm is, on average, ~60 min, so the time required to run a whole instance is around 8 days. As five instances have been run for both algorithms (CRO-SL and GA), the total simulation time was around 80 days.

Although computation time is high, this is not critical regarding the applicability of the proposal, because optimizing the railway infrastructure is an off-line process (not in real-time) that takes place in the planning phase. Besides, note that the big responsible for such long computation time is not the algorithm, but the electrical simulations associated with any new configuration of ESSs that must be tested. These electrical simulations are extremely demanding in terms of computation time because they must solve a lot of nonlinear load flows.

4.3. Parameterizations of Algorithms and Objective Function

The CRO internal configuration and parameters used in the simulations run are the following:

- Reef size: 200 individuals (corals) which 20% are initially empty places.
- HS substrate: $HMCR = 0.8$, $PAR = 0.2$, $\delta = 0.8$.
- DE substrate: F linearly decreasing during the run from 1 to 0.5.
- GM substrate: σ linearly decreasing during the run from 1 to 0.5.
- 2Px and MPx substrates: $k = 2, 3$ respectively.
- Number of maximum attempts in larvae setting: 3.
- Depredation process: Probability of depredation per iteration is set to 0.1. If depredation succeed, 10% of the weakest corals are removed from the reef.

In turn, the GA's internal configuration and parameters used are the following. Note that the parameters selected are those which yield the best performance in previous research [18] (except the population size, which is bigger because the search space is more demanding in this case study):

- Population size: 200 individuals (chromosomes).
- Crossover: One-point crossover.
- Mutation: During the mutation process, only two positions (genes) of each individual can be modified. The probability for a chromosome to randomly mutate 1 position is 20% and to randomly mutate 2 positions is 10%. If mutation takes place at a power

gene, the position is modified in $\pm\mathcal{S}_{\mathcal{P}}$. If it takes place in a capacity gene, the position is modified in $\pm\mathcal{S}_{\mathcal{C}}$.

Finally, the values considered for the fitness function are as follows:

- $e_{cost} = 0.0642 \in /$kWh. This value has taken into account the energy tolls established by the Spanish Government and the average energy price for Spain in 2018. See in [46] for more details about the calculation methodology.
- The installation cost ($C_0(x)$) is calculated with values inspired by the work in [4] (the storage technology chosen for the ESSs are EDLCs).
- $\mathcal{W}_{acc} = 2.5\%$ of interest rate.
- $T = 15$ years. This is a fair estimation of the ESSs' life according to the International Renewable Energy Agency (IRENA) [47].
- $M_b = 280000 \in$. This value makes it possible to try a considerable amount of different ESSs combinations that are within the budget limits and, at the same time, it is restrictive enough to unable highly costly configurations. Railway operators must provide this value in a real project.

4.4. Results

4.4.1. Performance Comparison between the CRO-SL and GA

Figure 5 shows the fitness evolution for both algorithms with the number of iterations. The first subplot represents each of the five instances run for each optimization algorithm, while the second subplot represents the average.
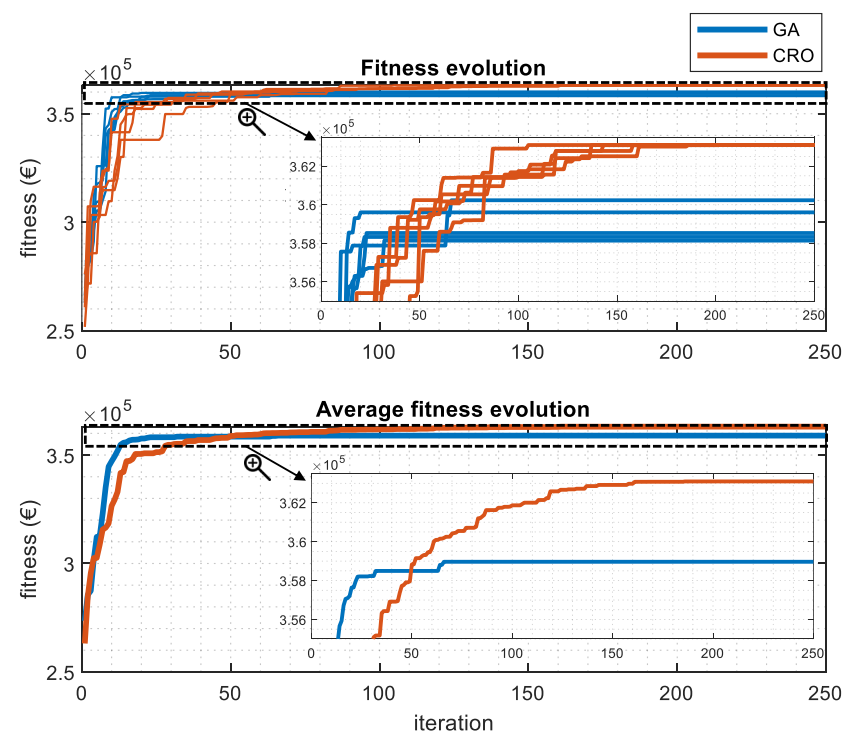

Figure 5. Comparison of the fitness achieved using the GA and the CRO-SL.

It can be seen that the GA, although faster than the CRO-SL obtaining a good solution at the beginning, gets trapped in several different local optima. By contrast, the CRO-SL is slower achieving an initial evolution, but it is always able to escape from the local optima (where the instances of the GA get trapped) at a reasonably constant speed (measured in number of iterations) and evolves its fitness until achieving a shared maximum value, which is very likely to be the global optimum. Therefore, this shows that the CRO-SL outperforms the GA, especially in terms of effectiveness.

4.4.2. Analysis of the Solutions

Before analyzing the solutions, the main conclusions obtained from pre-optimization tests to characterize the case-study line will be given. This will help us to understand the search space better. In these tests, it was found that the receptivity of the line was reasonably high. Indeed, it was found that, from the power-to-be-installed perspective, the receptivity is almost absolute (which means that practically the whole amount of regenerated energy is recovered) when the total power installed is greater than or equal to 3 MW. Besides, from the capacity-to-be-installed perspective, small sizes were found to be more than enough. This is mainly because the control curve applied for the ESSs tries to discharge them as soon as possible, so that the energy is not accumulated for a long time in the ESSs and the probability to accept the excess of regenerated energy in any time without the need of big capacities is high.

The analysis of the solutions given by the two algorithms shows several differences between them. As the best solution is always provided by the CRO-SL, an initial detailed analysis is given for this solution. Then, it will be compared with the sub-optimal solutions provided by the GA.

The optimum configuration obtained by the CRO-SL (the same in the five instances run) is given in Figure 6 and also displayed in Table 2.
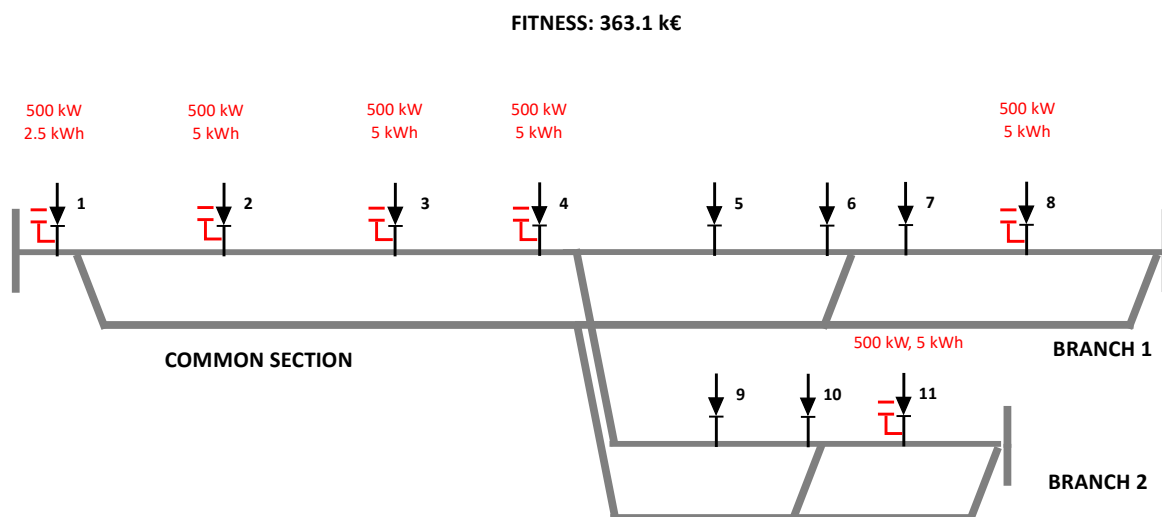

Figure 6. Optimum installation obtained by the CRO-SL algorithm for all the runs.

Table 2. Best solution found by the CRO-SL. The solution is represented by a pair of columns, where the first one refers to the power size ($\mathcal{P}$ [kW]) and the second to the capacity size ($\mathcal{C}$ [kWh]).

	CRO-SL Solution	
Location	$\mathcal{P}$	$\mathcal{C}$
1	500	2.5
2	500	5
3	500	5
4	500	5
5	–	–
6	–	–
7	–	–
8	500	5
9	–	–
10	–	–
11	500	5
Fitness	363.1 [k€]	

As can be seen, the total power installed is 3 MW, distributed into ESSs of 500 MW located at positions 1, 2, 3, 4, 8, and 11. The fact of not installing more power (until the maximum theoretical value from the pre-optimization tests of 3 MW) is due to economic reasons: the marginal benefit, in terms of energy saving, from installing the additional power does not compensate the cost associated to it. Small power sizes (500 kW) have been selected in order to install a big number of ESSs—six, distributed along the whole line—so that they can store the energy regenerated at any part of the line. Particularly, most of the ESSs are installed at the common section (shared by all the types of service and, therefore, with the highest number of trains). Besides, the ESSs installed at the common section, especially those closer to the branches, are also able to store the regeneration that takes place at the beginning of the branches (the whole line is a single electrical sector and, consequently, electrically connected). Finally, the regeneration that takes place at the end of the branches is stored by the two ESSs installed at the end (positions 8 and 11). Regarding the capacity, all the ESSs have a capacity of 5 kWh, except the one installed at position no.1, which has a capacity of 2.5 kWh. These values are also very reasonable because, as found in the pre-optimization tests, the capacities do not have to be very big, thanks to the ESS control curve used (which maximizes the ESS capability to store the regenerated energy that otherwise would be lost in the rheostats). Additionally, the big number of ESSs installed increases the probability to always have available (at least) one of the ESSs.

Regarding the solutions of the GA, Table 3 shows the characteristics of the solutions achieved in each of the 5 instances run (each solution is represented in two consecutive columns, where the first column corresponds to the power sizes and the second column to the capacity sizes).

As can be seen, each solution is different to the other, which means that the GA gets trapped in different local optima. This is largely due to (1) the big search space (with respect to the original search space presented in [18]) and (2) the difficulty of the fitness landscape.

Regarding the difficulty of the landscape, first it must be noted that the search space seems to have many different local optima, as different configurations can lead to similar energy savings. Additionally, it has been observed that the different variables of decision have different levels of influence on the fitness function, the power sizes and locations being the main variables of decision, relegating the capacity to a secondary level (which means that the fitness is less sensitive to most of its variations). This is largely due to their impact on the installation costs (crucial for the fitness function): in all the solutions obtained, the cost associated with the total power installed is more than 15 times higher than the cost associated with the total capacity installed.

Table 3. Fitness values obtained for best solutions provided by the GA. Each solution is represented by a pair of columns, where the first refers to the power sizes ($\mathcal{P}$ [kW]) and the second to the capacity sizes ($\mathcal{C}$ [kWh]).

Location	Solution #1 $\mathcal{P}$	$\mathcal{C}$	Solution #2 $\mathcal{P}$	$\mathcal{C}$	Solution #3 $\mathcal{P}$	$\mathcal{C}$	Solution #4 $\mathcal{P}$	$\mathcal{C}$	Solution #5 $\mathcal{P}$	$\mathcal{C}$
1	500	5	500	2.5	500	2.5	500	2.5	500	2.5
2	500	2.5	500	2.5	500	5	500	5	500	2.5
3	500	2.5	500	7.5	500	2.5	500	2.5	500	2.5
4	500	5	500	2.5	500	2.5	500	2.5	500	2.5
5	–	–	–	–	–	–	–	–	–	–
6	–	–	–	–	–	–	500	2.5	500	7.5
7	–	–	500	7.5	500	12.5	–	–	–	–
8	500	7.5	–	–	–	–	500	2.5	500	2.5
9	–	–	–	–	–	–	–	–	–	–
10	–	–	–	–	–	–	–	–	–	–
11	500	2.5	500	2.5	500	2.5	500	2.5	500	2.5
Fitness	360.2 [k€]		359.6 [k€]		358.5 [k€]		358.3 [k€]		358.1 [k€]	

This makes the GA to focus on the power size and location selection (which are the variables with the highest influence on the fitness), being unable to deal properly with the capacity selection. On the contrary, the CRO-SL can carry out a powerful global search of location and power sizes, but also an excellent treatment of the capacity values of the ESS. This way, note that the main difference between these algorithms seems to be the higher ability of the CRO-SL to carry out a good search for all the variables of decision, without relegating the capacity search of the solution. Indeed, among all the solutions provided by the GA, the one with the best fitness is very similar to the solution provided by the CRO-SL: it has the same locations and the same power sizes for each ESS, and the only difference is on the capacities of the ESSs, which are better selected by the CRO-SL approach.

Regarding the computational performance of the CRO-SL in this problem, Figure 7a shows the number of larvae (potential solutions to the problem) getting into the reef per substrate, over the iterations. Note that a new individual can enter in the population at a random place if its associated fitness value is better than the occupied one or if the place is empty. Due to the initial spots in the reef, in the first iterations of the algorithm's run there is a higher probability to enter, independently of the quality of solutions, so this number is usually larger at the beginning of the algorithm's run. In Figure 7b, the accumulative ratio of times that each substrate is producing the best larva is shown. It is remarkable that the best larva might not be better than the fittest coral of the reef. Regarding both figures, it can be pointed out that DE substrate seems to be the best for search space exploration and exploitation. The contribution of the other substrates seems to be smaller, but note how all of them include solutions in the reef, especially the GM substrate, at the end of the algorithm's evolution.
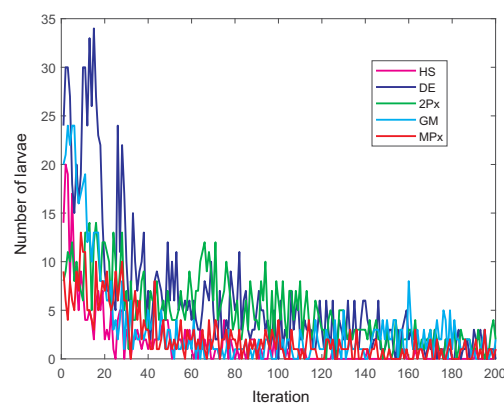
(**a**) # of larvae intro the reef per substrate.
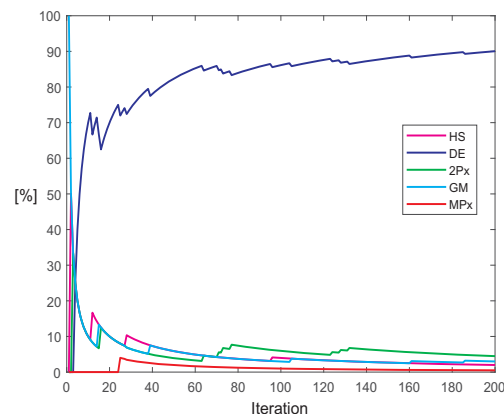
(**b**) % of best larvae per substrate.

Figure 7. Number of larvae entering to the reef and percentage of best larva generated per substrate in the CRO-SL: (**a**) # of larvae into the reef and (**b**) % of best larvae per substrate.

To conclude the analysis of results, it has been shown that the GA is faster than the CRO-SL but it gets trapped in local optima. The CRO-SL finds the best solution with fitness function 363.1 k€. The solutions provided by the GA are not the optimal solution found by the CRO-SL, and the difference in the value of fitness goes from 0.81% to 1.4%. This would have an impact on the profitability of the investment. In addition, the decision on the location and the sizes of the ESSs would vary depending on the algorithm applied to solve this decision problem. Thus, the CRO-SL is more effective finding the optimal solution of the proposed problem.

5. Conclusions

In this paper, we have faced a problem of optimal location and sizing of Energy Storage Systems (ESS) in DC-electrified railway lines. We have proposed and evaluated the performance of a multi-method ensemble meta-heuristic in this problem: the Coral Reefs Optimization with Substrate Layers (CRO-SL). It is an evolutionary-type multi-method approach able to run different search procedures within the same population. We have adapted the CRO-SL to cope with the problem at hand, by incorporating specific operators, a mix of the most successful search procedures in evolutionary algorithms, such as 2-point or multi-point crossovers, Gaussian mutation, Differential Evolution, or Harmony Search. All these approaches are combined into a single population algorithm to carry out an accurate search for the optimal solution to the problem. In fact, the CRO-SL has shown excellent performance in a real case study to determine the location and sizing of ESS in a real Spanish metro line, consistently improving the results obtained by an existing Genetic Algorithm in this problem in all the instances run.

Author Contributions: Conceptualization, S.S.-S., A.P.C., D.R.-D, S.J.-F. and A.F.-C.; methodology, S.S.-S. and D.R.-D.; software, D.R.-D., C.C.-G.; validation, Á.L.-L., R.R.P., A.P.-F. and A.F.-C.; writing—original draft preparation, S.S.-S., S.J.-F. and A.P.C.; supervision, S.S.-S., S.J.-F., A.P.C. and R.R.P. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Martinis, V.D.; Corman, F. Data-driven perspectives for energy efficient operations in railway systems: Current practices and future opportunities. *Transp. Res. Part C* **2018**, *95*, 679–697. [CrossRef]
2. Arboleya, P.; Mohamed, B.; El-Sayed, I. Off-board and on-board energy storage versus reversible substations in DC railway traction systems. *IET Electr. Syst. Transp.* **2020**, *10*, 185–195. [CrossRef]
3. Zhang, G.; Tian, Z.; Tricoli, P.; Hillmansen, S.; Liu, Z. A new hybrid simulation integrating transient-state and steady-state models for the analysis of reversible DC traction power systems. *Int. J. Electr. Power Energy Syst.* **2019**, *109*, 9–19. [CrossRef]
4. González-Gil, A.; Palacin, R.; Batty, P. Sustainable urban rail systems: Strategies and technologies for optimal management of regenerative braking energy. *Energy Convers. Manag.* **2013**, *75*, 374–388. [CrossRef]
5. Gelman, V. Energy Storage That May Be Too Good to Be True: Comparison Between Wayside Storage and Reversible Thyristor Controlled Rectifiers for Heavy Rail. *IEEE Veh. Technol. Mag.* **2013**, *8*, 70–80. [CrossRef]
6. Ibaiondo, H.; Romo, A. *Kinetic Energy Recovery on Railway Systems with Feedback to the Grid*; IEEE Press: Piscataway, NJ, USA, 2010; pp. 94–97.
7. Khodaparastan, M.; Dutta, O.; Saleh, M.; Mohamed, A.A. Modeling and Simulation of DC Electric Rail Transit Systems with Wayside Energy Storage. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2218–2228. [CrossRef]
8. Gao, Z.; Fang, J.; Zhang, Y.; Jiang, L.; Sun, D.; Guo, W. Control of urban rail transit equipped with ground-based supercapacitor for energy saving and reduction of power peak demand. *Int. J. Electr. Power Energy Syst.* **2015**, *67*, 439–447. [CrossRef]
9. Iannuzzi, D.; Ciccarelli, F.; Lauria, D. Stationary ultracapacitors storage device for improving energy saving and voltage profile of light transportation networks. *Transp. Res. Part C* **2012**, *21*, 321–337. [CrossRef]
10. Jefimowski, W.; Szelag, A. The multi-criteria optimization method for implementation of a regenerative inverter in a 3kV DC traction system. *Electr. Power Syst. Res.* **2018**, *161*, 61–73. [CrossRef]
11. Pereira, F.H.; Pires, C.L.; Nabeta, S.I. Optimal placement of rectifier substations on DC traction systems. *IET Electr. Syst. Transp.* **2014**, *4*, 62–69. [CrossRef]

12. Chuang, H.J.; Chen, C.S.; Lin, C.H.; Chu, S.H. Optimization of inverter placement for mass rapid transit systems using genetic algorithm. In Proceedings of the 2005 IEEE/PES Transmission & Distribution Conference & Exposition, Asia and Pacific, Dalian, China, 18 August 2005; pp. 1–6.

13. Calderaro, V.; Galdi, V.; Graber, G.; Piccolo, A. *Optimal Siting and Sizing of Stationary Supercapacitors in a Metro Network Using PSO*; IEEE Press: Piscataway, NJ, USA, 2015; pp. 2680–2685.

14. Xia, H.; Chen, H.; Yang, Z.; Lin, F.; Wang, B. Optimal Energy Management, Location and Size for Stationary Energy Storage System in a Metro Line Based on Genetic Algorithm. *Energies* **2015**, *8*, 11618–11640. [CrossRef]

15. Wang, B.; Yang, Z.; Lin, F.; Zhao, W. An Improved Genetic Algorithm for Optimal Stationary Energy Storage System Locating and Sizing. *Energies* **2014**, *7*, 6434–6458. [CrossRef]

16. Roch-Dupré, D.; Cucala, A.; Pecharromán, R.; López-López, A.; Fernández-Cardador, A. Evaluation of the impact that the traffic model used in railway electrical simulation has on the assessment of the installation of a Reversible Substation. *Int. J. Electr. Power Energy Syst.* **2018**, *102*, 201–210. [CrossRef]

17. Roch-Dupré, D.; Cucala, A.; Pecharromán, R.; López-López, A.; Fernández-Cardador, A. Simulation-based assessment of the installation of a Reversible Substation in a railway line, including a realistic model of large traffic perturbations. *Int. J. Electr. Power Energy Syst.* **2020**, *115*, 105476. [CrossRef]

18. Roch-Dupré, D.; Gonsalves, T.; Cucala, A.P.; Pecharromán, R.R.; López-López, A.J.; Fernández-Cardador, A. Determining the optimum installation of energy storage systems in railway electrical infrastructures by means of swarm and evolutionary optimization algorithms. *Int. J. Electr. Power Energy Syst.* **2021**, *124*, 106295. [CrossRef]

19. Maier, H.; Razavi, S.; Kapelan, Z.; Matott, L.; Kasprzyk, J.; Tolson, B. Introductory overview: Optimization using evolutionary algorithms and other metaheuristics. *Environ. Model. Softw.* **2019**, *114*, 195–213. [CrossRef]

20. García-Álvarez, J.; González, M.A.; Vela, C.R. Metaheuristics for solving a real-world electric vehicle charging scheduling problem. *Appl. Soft Comput.* **2018**, *65*, 292–306. [CrossRef]

21. Pellerin, R.; Perrier, N.; Berthaut, F. A survey of hybrid metaheuristics for the resource-constrained project scheduling problem. *Eur. J. Oper. Res.* **2020**, *280*, 395–416. [CrossRef]

22. Osaba, E.; Villar-Rodriguez, E.; Del Ser, J.; Nebro, A.J.; Molina, D.; LaTorre, A.; Suganthan, P.N.; Coello Coello, C.A.; Herrera, F. A Tutorial On the design, experimentation and application of metaheuristic algorithms to real-World optimization problems. *Swarm Evol. Comput.* **2021**, *64*, 100888. [CrossRef]

23. Wu, G.; Mallipeddi, R.; Suganthan, P.N. Ensemble strategies for population-based optimization algorithms—A survey. *Swarm Evol. Comput.* **2019**, *44*, 695–711. [CrossRef]

24. Wang, X.; Li, C.; Zhu, J.; Meng, Q. L-SHADE-E: Ensemble of two differential evolution algorithms originating from L-SHADE. *Inf. Sci.* **2021**, *552*, 201–219. [CrossRef]

25. Dokeroglu, T.; Sevinc, E. Memetic Teaching–Learning-Based Optimization algorithms for large graph coloring problems. *Eng. Appl. Artif. Intell.* **2021**, *102*, 104282. [CrossRef]

26. Deng, W.; Shang, S.; Cai, X.; Zhao, H.; Zhou, Y.; Chen, H.; Deng, W. Quantum differential evolution with cooperative coevolution framework and hybrid mutation strategy for large scale optimization. *Knowl.-Based Syst.* **2021**, *224*, 107080. [CrossRef]

27. Zeng, N.; Song, D.; Li, H.; You, Y.; Liu, Y.; Alsaadi, F.E. A competitive mechanism integrated multi-objective whale optimization algorithm with differential evolution. *Neurocomputing* **2021**, *432*, 170–182. [CrossRef]

28. Lopes Silva, M.A.; de Souza, S.R.; Freitas Souza, M.J.; de França Filho, M.F. Hybrid metaheuristics and multi-agent systems for solving optimization problems: A review of frameworks and a comparative analysis. *Appl. Soft Comput.* **2018**, *71*, 433–459. [CrossRef]

29. Salcedo-Sanz, S.; Muñoz-Bulnes, J.; Vermeij, M.A. New coral reefs-based approaches for the model type selection problem: A novel method to predict a nation's future energy demand. *Int. J. Bio-Inspired Comput.* **2017**, *10*, 145–158. [CrossRef]

30. Salcedo-Sanz, S. A review on the coral reefs optimization algorithm: New development lines and current applications. *Prog. Artif. Intell.* **2017**, *6*, 1–15. [CrossRef]

31. Salcedo-Sanz, S.; Ser, J.D.; Landa-Torres, I.; Gil-López, S.; Portilla-Figueras, J.A. The Coral Reefs Optimization Algorithm: A Novel Metaheuristic for Efficiently Solving Optimization Problems. *Sci. World J.* **2014**, *2014*, 739768. [CrossRef] [PubMed]

32. Garcia-Hernandez, L.; Salas-Morera, L.; Garcia-Hernandez, J.; Salcedo-Sanz, S.; Valente de Oliveira, J. Applying the coral reefs optimization algorithm for solving unequal area facility layout problems. *Expert Syst. Appl.* **2019**, *138*, 112819. [CrossRef]

33. Salcedo-Sanz, S.; García-Díaz, P.; Del Ser, J.; Bilbao, M.N.; Portilla-Figueras, J.A. A novel Grouping Coral Reefs Optimization algorithm for optimal mobile network deployment problems under electromagnetic pollution and capacity control criteria. *Expert Syst. Appl.* **2016**, *55*, 388–402. [CrossRef]

34. Salcedo-Sanz, S.; Camacho-Gómez, C.; Mallol-Poyato, R.; Jiménez-Fernández, S.; DelSer, J. A novel Coral Reefs Optimization algorithm with substrate layers for optimal battery scheduling optimization in micro-grids. *Soft Comput.* **2016**, *20*, 4287–4300. [CrossRef]

35. Jiménez-Fernández, S.; Camacho-Gómez, C.; Mallol-Poyato, R.; Fernández, J.C.; Ser, J.D.; Portilla-Figueras, A.; Salcedo-Sanz, S. Optimal microgrid topology design and siting of distributed generation sources using a multi-objective substrate layer Coral Reefs Optimization algorithm. *Sustainability* **2019**, *11*, 169. [CrossRef]

36. Tsai, C.W.; Chang, W.Y.; Wang, Y.C.; Chen, H. A high-performance parallel coral reef optimization for data clustering. *Soft Comput.* **2019**, *23*, 9327–9340. [CrossRef]

37. Salcedo-Sanz, S.; Camacho-Gómez, C.; Magdaleno, A.; Pereira, E.; Lorenzana, A. Structures vibration control via Tuned Mass Dampers using a co-evolution Coral Reefs Optimization algorithm. *J. Sound Vib.* **2017**, *393*, 62–75. [CrossRef]

38. López-López, A.; Pecharromán, R.; Fernández-Cardador, A.; Cucala, A. Assessment of energy-saving techniques in direct-current-electrified mass transit systems. *Transp. Res. Part C* **2014**, *38*, 85–100. [CrossRef]

39. Salcedo-Sanz, S.; Camacho-Gómez, C.; Molina, D.; Herrera, F. A coral reefs optimization algorithm with substrate layers and local search for large scale global optimization. In Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada, 24–29 July 2016; pp. 3574–3581.

40. Sánchez-Montero, R.; Camacho-Gómez, C.; López-Espí, P.L.; Salcedo-Sanz, S. Optimal Design of a Planar Textile Antenna for Industrial Scientific Medical (ISM) 2.4 GHz Wireless Body Area Networks (WBAN) with the CRO-SL Algorithm. *Sensors* **2018**, *18*, 1982. [CrossRef]

41. Camacho-Gómez, C.; Sanchez-Montero, R.; Martínez-Villanueva, D.; López-Espí, P.; Salcedo-Sanz, S. Design of a Multi-Band Microstrip textile patch antenna for LTE and 5G services with the CRO-SL ensemble. *Appl. Sci.* **2020**, *10*, 1168. [CrossRef]

42. Bermejo, E.; Chica, M.; Damas, S.; Salcedo-Sanz, S.; Cordón, O. Coral Reef Optimization with substrate layers for medical Image Registration. *Swarm Evol. Comput.* **2018**, *42*, 138–159. [CrossRef]

43. Gee, Z.W.; Kim, J.H.; Loganathan, G. A New Heuristic Optimization Algorithm: Harmony Search. *Simulation* **2001**, *76*, 60–68.

44. Storn, R.; Price, K. Differential Evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]

45. Eiben, A.E.; Smith, J.E. *Introduction to Evolutionary Computing*; Springer: Berlin, Germany, 2003.

46. Roch-Dupré, D.; López-López, A.; Pecharromán, R.; Cucala, A.; Fernández-Cardador, A. Analysis of the demand charge in DC railway systems and reduction of its economic impact with Energy Storage Systems. *Int. J. Electr. Power Energy Syst.* **2017**, *93*, 459–467. [CrossRef]

47. Ralon, P. Electricity Storage and Renewables: Costs and Markets to 2030. Technical Report. International Renewable Energy Agency. 2017. Available online: www.irena.org (accessed on 4 August 2021).