Hector, a new methodology for continuous and pattern-free heliostat field optimization

N.C. Cruz[a,*], S. Salhi[b], J.L. Redondo[a], J.D. Álvarez[a], M. Berenguel[a], P.M. Ortigosa[a]

[a]*Dpto. de Informática, ceiA3-CIESOL, Universidad de Almería (Spain)*
[b]*Centre for Logistics and Heuristic Optimization (CLHO), University of Kent (UK)*

Abstract

In the framework of central receiver solar plants, the heliostat field can take up to 50% of the initial investment and cause up to 40% of energy loss. The most popular design strategies are based on: i) forcing heliostats to follow known distribution patterns and ii) iterative selection of positions. However, these methods might produce suboptimal solutions. The evolution of computational platforms allows the development of more flexible approaches. In this work, Hector, a new meta-heuristic aimed at facilitating coordinate-based optimization, is presented. First, since East-West symmetry is imposed, one of those regions is ignored and the number of heliostats to be placed is halved. Second, the selected region is split into separate circular sectors around the receiver. Next, at every iteration, a new heliostat is added to the most promising sector. Then, it is optimized by a user-selected algorithm, as an independent problem, in a continuous search-space. This procedure is repeated until all the required heliostats have been deployed. The computed half is finally cloned into the other one. Two versions of this strategy are proposed. Our empirical results show that, for a given optimizer, better fields are obtained with Hector. The second version yields the best fields but requires more runtime.

Keywords: Concentrated solar power, Central receiver systems, Heliostat field design, Optimization, Meta-heuristic for problem division

*Corresponding author

Email addresses: ncalvocruz@ual.es (N.C. Cruz), s.salhi@kent.ac.uk (S. Salhi), jlredondo@ual.es (J.L. Redondo), jhervas@ual.es (J.D. Álvarez), beren@ual.es (M. Berenguel), ortigosa@ual.es (P.M. Ortigosa)

1. Introduction

Central receiver solar plants (CRSP) are interesting facilities for renewable electricity generation [1, 2, 3]. For the scope of this work, they consist of an array of solar-tracking mirrors called 'heliostats' and a radiation receiver on top of a tower. Heliostats reflect and concentrate the incident solar radiation on the receiver in days. This energy is progressively transfered to a working fluid which flows inside the receiver to be heated. When the fluid reaches the temperature required, it can be applied to a turbine cycle to generate electricity.

The heliostat field of CRSP, which can take up to 50% of initial investment, can also cause up to 40% of energy loss at operation stage [4, 5]. Therefore, the design of optimal heliostat fields is of great importance when deploying this kind of facilities [6]. Different optimization criteria can be considered: yearly efficiency [4, 7], investment cost [8], production price [9], land use [10], energy storage dispatch [11], etc. In this work, as in [12], the objective is to maximize the total power concentrated by the field. However, finding the best distribution of a large number of heliostats is a challenging problem [8]. The optical behavior of fields, which determines their power contribution and efficiency, depends on the heliostats, their interaction and time [13]. Besides, commercial plants have hundreds of heliostats, and their complete optimization requires computationally expensive simulations. These aspects, and the lack of details about optimized fields highlighted in [13], make heliostat field optimization an open and widely studied problem [12]. As summarized in [8, 12], the two fundamental field design strategies are i) the use of geometrical patterns to place heliostats and ii) their iterative deployment.

Making heliostats follow geometrical patterns reduces the problem dimensionality to the parameters of the pattern [14] and simplifies the analysis of fields [15]. Several patterns have been developed throughout the years [16]. A classic and popular one is radial staggering [17], which inspires some recent proposals [13, 9, 18]. An example of modern pattern is the spiral proposed in [7], which has attracted the interest of several researchers [19, 20, 21]. Regarding the iterative deployment of heliostats, the work in [10] is one of the most popular ones. The work in [22] continues in that way and tries to improve the precision of field evaluations. The method developed in [8, 23] is

iterative too. However, instead of simulating a pattern-free deployment with a dense grid like [10, 22], it adjusts the coordinates of every heliostat directly on the ground. Nevertheless, most methods opt for concatenating several steps. For instance, the use of patterns is usually linked to oversizing fields before selecting the best positions [7, 4, 24, 9]. The proposal in [5, 13] starts by creating a dense radially staggered field which is progressively expanded to balance different energy losses. In [25, 26], an initial field is generated with any pattern-based approach. Next, every heliostat is moved around its zone, within a discrete grid of positions, to improve the field. The work in [27] extends that in [26] by dividing the field into cells and decreasing the griding size progressively. Recently, powered by modern computational resources, several authors have studied how to optimize the coordinates of a whole field as a continuous optimization problem. In [12], a gradient-based optimizer is applied with promising results over traditional methods. In [28], a genetic algorithm is designed for this optimization approach. The proposal in [29] is also a genetic algorithm, but it allows infeasible solutions during the search.

This work proposes a new meta-heuristic to support continuous field optimization with any optimizer selected. Its name is 'Hector', which results from replacing the first letter in the word 'sector' with 'h' (from 'heliostat'). Its novelty lies in offering a mid-term solution between handling all the coordinates directly [29, 12, 28], and fixing the field after every deployment [8]. The greedy approach in [8] is limited due to its local scope. The methods in [29, 12, 28] have a global scope, but they try to solve such a difficult problem that the results achieved are likely to be suboptimal too. Hector assumes that, in practical terms, the same optimizer can obtain better results by addressing simpler instances, yet without limiting to a greedy strategy. Thus, it states a division method with iterative deployment that is based on solving subproblems with high quality. The only previous division methods are linked to patterns [15, 30, 18] and discrete positions [27], not to continuous field optimization as in this work. Hector forces East-West symmetry, which makes it possible to focus on a single side and halves the number of heliostats to place. Second, the chosen side is split into a user-given number of circular sectors. At every cycle, a new heliostat is added to the most promising sector. After that, the optimizer is launched to optimize the sector as an independent field. Finally, all the sectors are linked and cloned into the other side. The effectiveness of Hector has been proved by studying its effect on the optimizer designed in [29] by the authors of this work.

The remainder of this paper is organized as follows: Sec. 2 states the

optimization problem and defines the heliostat field model. Sec. 3 describes the two versions of the meta-heuristic proposed. Sec. 4 and Sec. 5 contain the experimentation and conclusions, respectively. Finally, Appendix A has a nomenclature table, and Appendix B describes the optimizer used to test Hector.

2. Field optimization and modeling

2.1. Optimization problem

The receiver and heliostats are considered to be on a flat ground. The receiver is assumed cylindrical, with height γ and diameter ρ. Its central point is at a height λ over the ground. Its supporting structure is a cylinder with the same diameter and height $\lambda - \gamma/2$. In this context, a Cartesian coordinate system that consists of directions East (X), North (Y) and Zenith (Z) is defined. Its origin is the central point of the tower base.

Let H be the total number of heliostats in the field. All of them feature the same rectangular reflective surface of size $l \cdot w$ (height and width, respectively), and diagonal $c = \sqrt{l^2 + w^2}$. That surface is directly mounted on a supporting structure of height z over the ground. Thus, every heliostat h is identifiable in the flat ground around the receiver by its central point, (x_h, y_h). In this context, the design of a whole heliostat field can be defined by a vector $F = ((x_1, y_1), \cdots, (x_H, y_H))$ in $\mathbb{R}^{2H}$.

The target optimization problem is formulated as follows:

$$\begin{cases} \underset{F}{\text{maximize}} & P_T(F) \\ \text{subject to} & R_{min} + \dfrac{c}{2} \leq \sqrt{x_h^2 + y_h^2} \leq R_{max} - \dfrac{c}{2},\ \forall h \\ & \sqrt{(x_h - x_i)^2 + (y_h - y_i)^2} > c,\ \forall h \neq i \\ & atan(|x_h|, y_h) \leq \left(\beta - asin\left(\dfrac{c}{2\sqrt{x_h^2 + y_h^2}}\right)\right),\ \forall h \end{cases} \tag{1}$$

where $P_T(F)$ is the power concentrated by a field, defined by a vector F, on its receiver throughout T instants. The first constraint establishes that no heliostat can trespass the region defined by R_{min} and R_{max}. The maximum area taken by a heliostat on the plane is defined by its diagonal c, which is the diameter of the circumference that contains its reflective surface when it

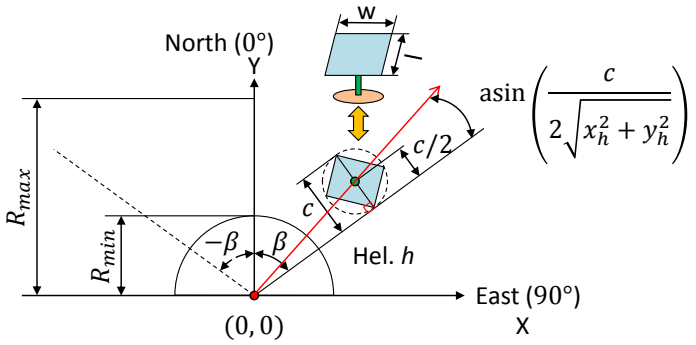

Figure 1: Graphical definition of the valid region for heliostat deployment

is parallel to the ground. Thus, to avoid trespassing, the Euclidean distance from every heliostat h to the origin of coordinates, i.e., $\sqrt{x_h^2 + y_h^2}$, must be greater than $R_{min} + \frac{c}{2}$ and less than $R_{max} - \frac{c}{2}$. The second constraint requires that all the heliostats are separated enough to move freely. Hence, the Euclidean distance between the center of any two heliostats must be greater than c. The third constraint requires that no heliostat trespasses an imaginary line that starts at (0, 0) and forms an angle β with the North direction. For $\beta \leq 180°$, this constraint is applied to the East and West sides. Function *atan* returns the angle defined by a heliostat and point (0, 0) with the North. This value is compared to β modified by the angle that the heliostat would take without trespassing, which depends on c and its distance to (0,0) (see Fig. 1).

2.2. Heliostat field model

In practical terms, it is necessary to build a field model that can be used to evaluate and compare fields [12]. The sample field model that has been implemented in this work is described next. The interested reader can find further information about its components in [31, 32, 33], which are publicly available. However, notice that the proposal of this paper is not linked to it.

The objective function, i.e., $P_T(F)$, is defined as follows [31, 12]:

$$P_T(F) = A \sum_{t=t_1}^{T} I_t \left(\sum_{h=h_1}^{H} \eta_h(t) \right) \tag{2}$$

where A is the reflective surface linked to the heliostat type, i.e., $\approx l \cdot w$, and I_t is the incident solar radiation density at every instant t. $\eta_h(t)$ is the

optical efficiency of a heliostat h at instant t. It is defined as [4, 12]:

$$\eta_h = \eta_{cos} \cdot \eta_{sb} \cdot \eta_{itc} \cdot \eta_{aa} \cdot \eta_{ref} \tag{3}$$

where η_{cos}, η_{sb}, η_{itc}, η_{aa}, η_{ref} are referred to the cosine, shading and blocking, interception, atmospheric attenuation and reflectivity factors, respectively [31, 7]. They are defined between 0 (minimum) and 1 (maximum). Note that not all these factors depend on time. Thus, the instant-related notation has been omitted for simplicity.

As summarized in [33], any heliostat is oriented so that the normal vector to its surface bisects the angle formed by the directions of solar radiation, $\hat{s}$, and that from the heliostat center to its aim point at $(0, 0, \lambda)$, $\hat{r}$. η_{cos} models how its effective reflective area is reduced because of its orientation. This is the most important factor in terms of energy loss for fields [33]. The cosine of the incidence angle of radiation on the heliostat, θ_i, is the reduction factor, i.e., η_{cos}. It can be calculated from Eq. (4) [33]:

$$\cos 2\theta_i = \hat{s} \cdot \hat{r} \tag{4}$$

Although the direction from every heliostat center to the aim point is considered constant, the solar ray direction depends on t. Hence, η_{cos} is recomputed for every considered instant. If the apparent solar position at t is defined by the angles of altitude from the ground, α, and azimuth from the North, a, (0° North, 90° East), $\hat{s}$ can be computed according to Eq. (5) [33]:

$$\hat{s} = (\cos\alpha \sin a, \cos\alpha \cos a, \sin\alpha) \tag{5}$$

η_{sb} models that heliostats might obstruct incident and/or reflected radiation from each other, which reduce their effective reflective area. The first phenomenon, called 'shading', and the second one, known as 'blocking', are grouped in factor η_{sb}. For a certain heliostat, η_{sb} is defined as the ratio between its neither blocked nor shadowed reflective area and the total one [31, 7]. Its calculation has been modeled as a polygon clipping problem as proposed in [32]. For every studied heliostat, the four vertexes of the reflective surface of any other one that could affect it are projected on its plane. Next, the resulting quadrilaterals are progressively subtracted to that of the reflective surface of the studied heliostat. η_{sb} is finally obtained by dividing the remaining area by the total one. This computation has been

implemented with the open-source library Clipper [34]. This factor depends on the instantaneous orientation of every heliostat and must be recomputed every time.

Regarding η_{itc}, it models that flux maps reflected by heliostats might not entirely fall on the appropriate zone of the receiver. This factor is calculated with the analytical model proposed in [31]. It assumes that the flux map reflected by every heliostat on the receiver is elliptic. Heliostats are considered error-free and perfectly canted, i.e., with a focal length equal to its distance to the receiver [31, 32]. Considering 9.3 mrad of deviation in solar rays [33], the horizontal axis of the elliptic flux map, D_{image}, can be computed as follows [31]:

$$D_{image} = 0.0093d \tag{6}$$

where d is the distance between the studied heliostat and its aim point. The vertical axis of this ellipse, L_v, can be computed from D_{image} as follows [31]:

$$L_v = \frac{D_{image}}{d_{xy}/d} \tag{7}$$

where d_{xy} is the image from the studied heliostat and the receiver on the ground, i.e., only considering directions X and Y. η_{itc} is finally computed as follows [31]:

$$\eta_{itc} = \frac{\frac{\pi}{4} L_v D_{image} - \left(\frac{(L_v - \gamma) D_{image}}{1.284} + \frac{(D_{image} - \rho) L_v}{1.284} \right)}{\frac{\pi}{4} L_v D_{image}} \tag{8}$$

where $L_v - \gamma$ and $D_{image} - \rho$ are zeroed when the difference is negative (as the ellipse is not larger than the corresponding receiver dimension). It is only necessary to compute this model once per heliostat.

Regarding η_{aa}, it models how the atmosphere attenuates the power reflected by heliostats. It is computed with the following expression [4, 7]:

$$\eta_{aa} = \begin{cases} 0.99321 - 0.0001176d + 1.97 \cdot 10^{-8} d^2, & \text{if } d \leq 1000\, m \\ \exp\left(-0.0001106d\right), & \text{otherwise} \end{cases} \tag{9}$$

where d is the same distance previously defined. As the previous one, this model does not depend on time.

Finally, η_{ref} models energy loss at reflection on the reflective surface. This factor is considered a common construction constant as in [31, 7].

2.3. Model validation

As described, the field model consists of several components from the open literature. They have already been validated conceptually by their original authors. Thus, in practical terms, it is only necessary to check their implementation.

The sample problem proposed in [33] was solved to validate the computation of η_{cos}. The reference value is 0.9510, and the model implemented returns 0.9510. This computation implicitly validates the solar and target vector calculation. Regarding η_{sb}, the benchmarks included in [32] have been replicated with the results shown in Tab. 1. The small differences must be caused by the fact that the solar positions are not given explicitly, and they have been derived from the date and time with the models in [33]. Moreover, the clipping algorithm used is different, and the selection of potentially affecting heliostats is not covered in the original source. Hence, the implementation is considered correct. This is the most sophisticated part of the model, and its validation serves to confirm the correct behavior of aspects such as the proper orientation of heliostats. In relation to η_{itc}, the result of the field model was compared to that obtained in [31]. The reference value is 0.9502, and the model implementation yields 0.9501. Regarding η_{aa}, the result of the field model was compared to that of [31]. The reference value is 0.9887, and the model implemented returns 0.9887. Finally, the use of η_{ref} as a constant was checked too. Consequently, all the components of the field model work as expected.

Table 1: Validation results of η_{sb}

Test name	A	B	B	C	D
Reference (%)	0.7610	0.3080	0.8581	0.9614	0.5212
Obtained (%)	0.7629	0.3037	0.8440	0.9607	0.5034

3. Description of the meta-heuristic Hector

3.1. Introduction

As introduced, the problem defined in Sec. 2.1 is extremely complex. First, even after considering a flat ground, there are still $2H$ variables to handle. Second, taking into account that the distance between any pair of points is the same in both directions as well as the three types of zone limits

per heliostat, there are $(H^2-H)/2+3H = (H^2+5H)/2$ non-linear constraints to fulfill. Third and lastly, the objective function is only conceptually known. In this context, optimizers are unlikely to converge to a global optimum.

Hector is a new meta-heuristic which assumes that facing such a complex problem is not practical. Therefore, and inspired by the classic Divide and Conquer paradigm, it tries to show a layered version of the problem to any selected optimizer, O. This approach renounces the certainty of being able to achieve an optimal result, which is considered to be remote and difficult to prove. It relies on the idea that O will ultimately obtain better results by working on a sliced version of the problem than directly facing its whole complexity. Convergence would hence occur in better conditions with simpler problems. Moreover, as detailed later, the underlying magnitudes feature a certain degree of separability.

There are two versions of Hector, the standard one detailed in Sec. 3.2 and the enhanced one described in Sec. 3.3. Both of them share the same underlying principles summarized next:

- Assumption of East-West symmetry and focus on a single side.
- Division of the selected side into circular sectors.
- Iterative deployment of heliostats in the most promising sector.
- Coordinate-based optimization of every altered sector.

3.2. Standard Hector (SH)

Visually, the sun moves from sunrise to sunset at symmetrical times around noon. In solar time, its azimuth is 180° at noon and moves at a constant rate of 15 degrees per hour [33]. Similarly, its altitude angles can also be considered symmetrical in relation to noon. Thus, in optical terms, reflection angles of heliostats in the East and West zones should be swapped after noon. Moreover, solar radiation density depends on solar altitude [33]. In this context, an optimal heliostat distribution on the East side of the field, using the North direction as symmetry axis, should also be optimal for the West zone. According to these ideas, it is possible to focus on a single side and avoid East-West trespassing. Hence, only $H/2$ heliostats need to be placed, i.e., H optimization variables. A straightforward way to consider this situation in Eq. (1) is the removal of the absolute value in the last constraint. Obviously, this assumption forces the inclusion of an additional step: Once

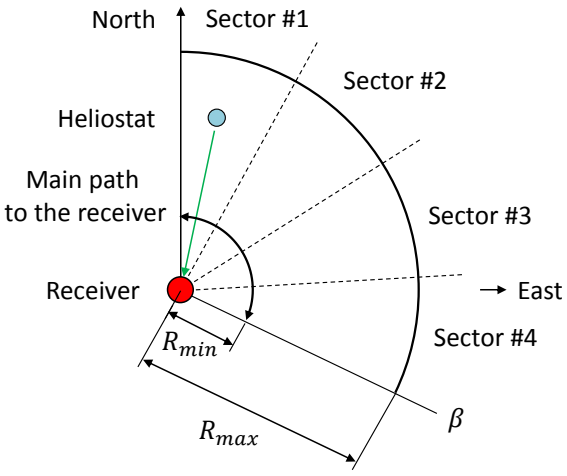

Figure 2: Division of the selected side into circular sectors

the selected side has been designed, it must be cloned to the other one with the North direction as the symmetry axis.

In spite of halving the number of decision variables, it is still relatively large. In order to further reduce the complexity of the problem, the selected side will be divided into S equal circular sectors as depicted in Fig. 2, where S is a user-defined parameter. As commented in [5, 10], it can be said that most energy losses that affect the performance of a certain heliostat do not depend on its neighbors. Only shading and blocking are based on their interaction (see Sec. 2.2). In fact, the proposed division would contain the main path between every heliostat and the receiver, which would still make possible an acceptable estimation of blocking. Moreover, as real shading and blocking problems depend on the nearest neighbors [4, 5], most of them would be included in the same sector. To contribute to this principle, the no-trespassing condition of the initial region is inherited by all sectors (see Fig. 3). Therefore, considering every sector as a different field should suppose a small precision loss while reducing complexity by an extra S factor. Only two considerations must be highlighted. First, an additional constraint must be included for the new angular limit defining every sector (minimum and maximum angles). It can be simply derived from that for β in Eq. (1). Second and last, before generating its symmetric side, all the sectors are combined to form the selected hemisphere.

At this point, the way in which $H/2$ heliostats are distributed among

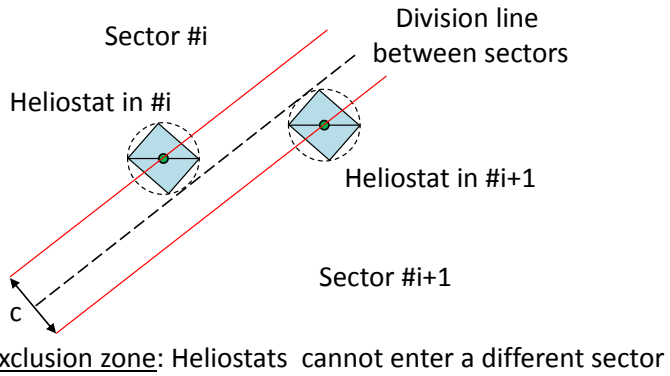

Figure 3: Exclusion zone between heliostat to avoid trespassing

S empty sectors must be defined. Every sector features a certain factor of attractiveness which is 1 (maximum) when it is empty and 0 (null) when it is full. When a new heliostat is going to be added, it is deployed in the most attractive sector. Under equal conditions, if the studied field is in the Northern Hemisphere, that which is closer to the North is selected. This heuristic is based on the fact that better reflection angles can be achieved there [33]. The opposite decision should be made for a field in the Southern Hemisphere, though. Consequently, the first sector to be used by Hector is easy to determine. In fact, as any empty sector is more efficient than any non-empty one, the first S heliostats will be consecutively deployed in the S sectors.

Every time a heliostat is linked to a certain sector, S_i, optimizer O is launched to find its best design. After this procedure ends, the attractiveness of that sector is updated according to Eq. (10), where $|S_i|$ is the number of heliostats in it. Conceptually, Eq. (10) can be seen as the ratio between the power reflected by that set of heliostats and their theoretical maximum, i.e., the overall efficiency factor [7, 28], divided by their number. This latter division aims to penalize the overuse of sectors: If all of them had the same number of heliostats, the one with the higher efficiency would be more attractive. However, if all the sectors had the same efficiency factor, the one with the fewer heliostats would be better considered.

$$attractiveness(S_i) = \frac{P_T(S_i)}{|S_i|^2 \, A \sum_{t=t_1}^{T} I_t} \tag{10}$$

From the previous statements, it is possible to precisely define the proce-

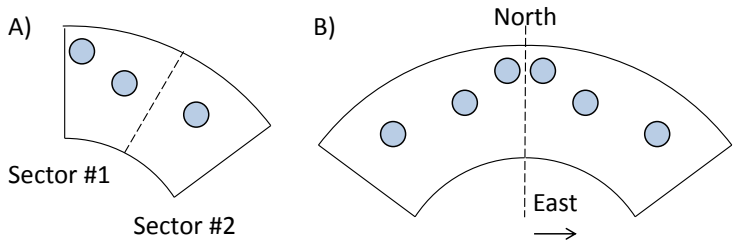

Figure 4: Final steps: composition of sectors and cloning

dure executed by Hector according to Alg. 1. For readability, explicit input is focused on the most descriptive variables. However, details such as the size of heliostats and the configuration of O are assumed to be available. The procedure starts by creating S empty circular sectors of equal size (line 1). By default, the preferred field side to work in is the East one. According to symmetry, this decision should not be relevant, though. Then, at line 3, an iterative loop is executed until $H/2$ are placed in the chosen side (with independence of the sectors). At every iteration, the best sector, i.e. that with higher attractiveness, is selected (line 4). If all of them have the same attractiveness, as Northern Hemisphere is assumed, that closer to the north of the field is preferred. It must be noted that function *PickTheBestSector* is expected to return the best sector which can be used, i.e., not considered full. If such a sector cannot be found it yields $\emptyset$, as checked at line 5. That situation could lead to stop without having $H/2$ heliostats deployed (line 14). In fact, Hector cannot grant that any requested deployment is achievable by using O or that it is even feasible. When it launches O at line 8, the number of heliostats in the chosen sector was previously removed from the count at line 6. Once O ends, Hector takes its output and ensures that it is a valid sector, which also defines the value of variable *success*. It is *true* if *bestSector* has all the heliostats it had before including the new one at line 7 plus that one. However, if it has been necessary to remove heliostats in order to build a feasible sector, *success* is set to $false$. The selected sector is then tagged as full (lines 9 to 11) and it will not be modified anymore. The iteration ends by updating the number of heliostats successfully deployed in *bestSector* at line 12. After that loop, the procedure merges all the sectors into a single one and clones it to the ignored side at line 17 (see Fig. 4). Finally, the resulting field is returned at line 18.

The addition of a new heliostat is performed in line 7. Its coordinates are

randomly generated inside $bestSector$, where feasibility is promoted by making a maximum number of attempts to get a collision-free position (which is considered a contextual parameter too). If that cannot be achieved, responsibility is inherited by O assuming that it performs better than randomness. Similarly, when parsing the output of O to set $success$ (line 8), the correctness of every heliostat is checked both for trespassing and collisions (only against heliostats already processed). Upon infeasibility, Hector tries to randomly move the problematic heliostat to a valid position for the same maximum number of attempts. However, in contrast to the previous permissiveness, processing ends if it cannot find a feasible position for that heliostat. Then, $success$ would be set to $false$ and only successfully checked heliostats would be maintained in the sector (which would ultimately be tagged as full at line 10).

Finally, it is important to highlight that the positions of heliostats are not fixed after line 8. Hence, the distribution of a certain sector can be totally redesigned the next time it is selected. A sector with a new heliostat is considered a different problem and O will try to solve it as well as possible. Nevertheless, making an effort to solve every sector when it is extended is far from being useless. First, the attractiveness of sectors is kept precisely updated. Second and last, Hector provides the selected sector to O (see line 8), which is encouraged to use that input information. For instance, if O relies on a population of solutions such as genetic optimizers, that one could be loaded as part of its initial population [29]. Similarly, a gradient-based optimizer could try to use it as a possible starting point. Consequently, this work-flow can be considered as an implicit multi-start component [8].

3.3. Enhanced Hector (EH)

A key principle of Hector is that dividing the whole problem into smaller units can lead any optimizer to obtaining better results. However, as mentioned, there is some loss of precision when dividing the available side. Generally speaking, a trade-off between sectoring and the size of sub-problems must be found in order not to be counterproductive. In SH, slicing the objective function and the inherent reduction of mobility during the search are potential problems. As exclusion zones are left between sectors to ensure separability (see Fig. 3), those areas cannot contain heliostats. Hence, the more sectors that are created, the more zones that are blocked.

EH tries to overcome this situation by slightly altering the definition of sectors. The procedure explained in Sec. 3.2 and Alg. 1 remains almost

Algorithm 1: Hector meta-heuristic for heliostat field optimization

Input: Real: R_{min}, R_{max}, β; Int: S, H; Optimizer: O

SectorSet $sectors$ = CreateSectors(R_{min}, R_{max}, β, S, Side=East);
Int $numHel = 0$;
while $numHel < H/2$ **do**
    Sector $bestSector$ = PickTheBestSector($sectors$, Preferred=North);
    if $bestSector \neq \emptyset$ **then**
        $numHel = numHel$ – HeliostatsIn($bestSector$);
        AddHeliostat($bestSector$, 1);
        Bool $success$ = RunOptimizer(O, $bestSector$);
        if *!success* **then**
            SetFull($bestSector$);
        end
        $numHel = numHel$ + HeliostatsIn($bestSector$);
    else
        break;
    end
end
CoordinateSet $finalField$ = Compose($sectors$, Add_Symmetry=True);
return $finalField$;

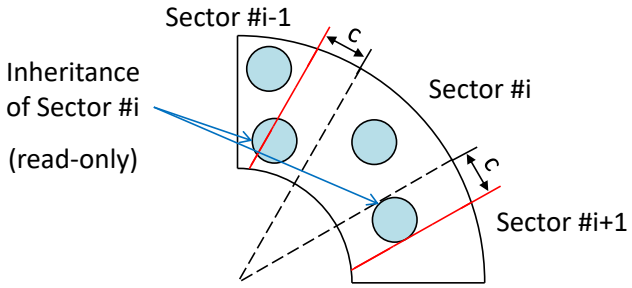

Figure 5: Graphical definition of inheritance for a certain sector

unaltered. Heliostats do not have to keep any exclusion zone between sectors anymore (apart from the last one limited by β). That can be achieved by removing the *asin* correction factor used to avoid trespassing the minimum and maximum angles defining sectors (see Eq. (1)). Constraints linked to R_{min} and R_{max} are maintained, though. However, if exclusion zones are simply ignored in the problems shown to O for every sector, their composition to form the final side might be infeasible. Hence, regarding Alg. 1, if *bestSector* is sector S_i, a read-only set of heliostats from both S_{i-1} and S_{i+1} is appended to it. This set, called 'inheritance' does not contain all heliostats from S_{i-1} and S_{i+1} but only those up to c from the angular limits. Although inherited heliostats cannot be moved by O, it considers them to evaluate its candidate solutions. Thus, it knows if its sector is feasible or not and, when it is, neighboring heliostats affect the performance of those in the optimized sector. These ideas are depicted in Fig. 5. In practical terms, the inheritance of *bestSector* should be loaded between lines 6 and 7 in Alg. 1 (e.g., *LoadInHeritance(bestSector, bestSector.left, bestSector.right)*).

On the one hand, EH virtually solves the reduction of feasible surface linked to the number of sectors in SH. Their evaluation as independent fields is also more precise because blocking and shading, caused by heliostats near the borders, can be better considered (while they were neglected by SH). Moreover, the number of decision variables to be handled at every sector remains unaltered. On the other hand, slightly more overlapping constraints must be fulfilled due to inheritance. Besides, even if the logical configuration of O is the same for SH and EH, the runtime required to evaluate sectors is likely to be higher with EH. This is due to the fact that inherited heliostats must be loaded to compute $P_T(F)$ too. Additionally, if some heliostats were left too near the symmetry axis, they could not be cloned to the other side.

Hence, cloning might not turn $H/2$ into H heliostats as required. When needed, EH handles this situation by creating a final sector covering the whole field and whose read-only inheritance is the cloned field. After that, O is finally launched to place the remaining number of heliostats without colliding with those already deployed. This logic would be included at line 17 in Alg. 1.

4. Experimentation and results

EnGA [29] is the continuous field optimizer selected to test the effectiveness of Hector, i.e., its module O. Appendix B includes a brief description of this genetic optimizer for the sake of completeness. This method, also proposed by the authors of this work, has been selected because of i) its capabilities to work with unfeasible solutions to maximize the search flexibility and ii) its direct use of parallel computing. EnGA, the objective function and Hector have been implemented in C++. The experimentation platform is a computer with 2 Intel Xeon E5 2650 processors (16 cores in total), and 64 GB of RAM. EnGA can spawn 16 threads to handle its population. The problem context consists of designing a field of 300 heliostats. It is inspired by the real one called CESA-I. It is part of the *Plataforma Solar de Almería* (PSA), the largest concentrating solar technology research center in Europe, in the south of Spain (latitude 37.083° N, longitude 2.35° W) [13]. Thus, the receiver center is at 86.60 m over the ground, has a vertical height of 2.45 m and a width of 2.25 m. The heliostats have reflective surfaces with 6.600 m of height, 6.616 m of width, and a reflectivity factor of 0.8. Their mount point height is 3.65 m. The available surface is defined by $R_{min} = 20.0$ m, $R_{max} = 300.0$ m and $\beta = 90°$.

In this framework, a design-point case was preliminarily studied to confirm the viability of Hector. The results obtained, which were positive, can be found in [35]. After this initial trial, a more realistic case has been studied. Instead of an isolated instant, the 21^{st} of every month has been selected. This is a descriptive enough selection of days to study the yearly performance of fields [4] and makes sense for real application. From them, a set of $T = 36$ instants has been defined by considering the solar position at 9:00, noon and 15:00 (solar time). The model explained in [36] has been used to estimate the direct solar radiation at every one of them. In this context, EnGA has been executed alone to set the time reference. Its population size is 1200, and it starts with 5 staggered-based solutions to ensure its proper conver-

Table 2: Configuration of EnGA for SH depending on S

	EnGA	*SH (1)*	*SH (2)*	*SH (3)*	*SH (4)*
Pairs	600	600	600	640	720
Cycles	2000	150	310	400	560

gence. At every cycle, it forms 600 reproduction couples. Progenitors, like surviving individuals at the end of cycles, are selected from tournaments of 4 individuals. Descendants have an overall mutation probability of 0.3 and, once started, every heliostat can be randomly repositioned with a probability of 0.05. At the end of every cycle, the best 30 candidate solutions in the population are directly selected to survive. EnGA will execute 2000 cycles in total. However, this configuration has been adapted to run within SH and take a runtime similar to the reference. Apart from removing the extra help of loading initial staggered-based individuals, only the number of pairs and cycles have been changed. Tab. 2 includes how they have been varied from the exclusive execution of EnGA (first column) to the use of 1 to 4 sectors. As can be seen, the iterative execution of SH makes the runtime with few pairs and cycles similar to that of EnGA alone. Note that the use of sectors results in a simpler and less computationally demanding objective function. Thus, the more sectors that are created, the more cycles and pairs that are needed to approach to the runtime required.

Tab. 3 contains the results of launching EnGA alone (first row) and linked to SH (remaining rows). Due to the stochasticity of EnGA, they are the average of five independent executions as in [35]. The first column contains the average power concentrated by the designed fields on their receivers, i.e., the maximized variable. The second column shows the efficiency of those values compared to the maximum power achievable, i.e., around 378 MW. It results from accumulating the maximum power that could concentrate the set of heliostats, without optical losses, at every instant ($T = 36$). Finally, the third column shows the average runtime. According to the results, there is not doubt about the positive effect of Hector: all the solutions of EnGA within SH are better than those of the optimizer alone. The best record is in bold font. It is important to highlight that, for this type of problem, small percentage differences are highly relevant [4, 7]. In contrast to the preliminary test, the positive trend of increasing S is maintained in all the cases. It might be because of the higher difficulty of the problem, which allows more divisions. Notice that the efficiency values are lower than those

Table 3: Average results of EnGA with and without SH

	Av. Power (MW)	*Av. Efficiency*	*Av. Runtime (s)*
EnGA	256.35	0.6779	76748
SH (1)	259.40	0.6860	66232
SH (2)	259.56	0.6864	61584
SH (3)	260.05	0.6877	51228
SH (4)	**260.50**	**0.6889**	**56187**

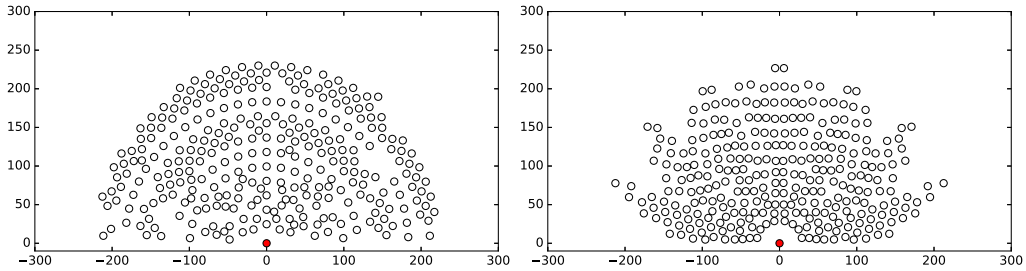

Figure 6: Best fields achieved by EnGA only (left) and with SH(4) (right)

of the preliminary test. It is because the problem is much more difficult: The fields are evaluated for several solar positions and it is not easy for heliostats to be productive for low solar altitudes. Finally, Fig. 6 includes the best fields achieved with EnGA alone (left) and within SH and 4 sectors (right), whose efficiencies are 0.6782 and 0.6894, respectively. As can be seen, the field of EnGA is based on one of its initial pattern-based solutions with haphazard improvements made during the evolutionary process. In contrast to it, that of SH features a compact design with perfect East-West symmetry. It concentrates more heliostats in the central zone, and has a progressive transition to further ones in order to attenuate the effect of shading and blocking at low solar altitudes. Thus, SH has been able to guide the same optimizer to achieve better results in the same (or less) time.

As can be seen in Fig. 6, fields designed with SH has a clear separation between sectors. Therefore, the previous experiments have been repeated with EH to allow more flexibility. The configuration given to EnGA within SH has not been changed. Tab. 4 contains the results obtained with EH in the same format previously used. The overall behavior of sectoring is positive again. Moreover, for the same number of sectors, the fields of EH are better than those of SH. These results confirm that EH is less affected by

Table 4: Average results of EnGA with and without EH

	Av. Power (kW)	*Av. Efficiency*	*Av. Runtime (s)*
EnGA	256.35	0.6779	76748
EH (1)	261.14	0.6906	69807
EH (2)	262.06	0.6930	88075
EH (3)	262.12	0.6932	93556
EH (4)	**262.46**	**0.6941**	**120504**

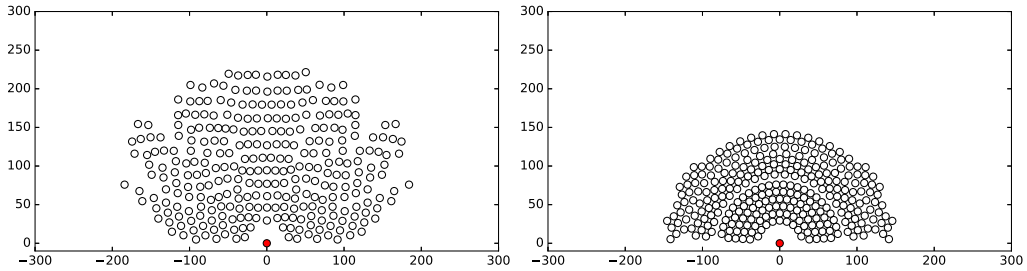

Figure 7: Best field achieved with EH (left), and field design of SAM [37, 38] (right)

the overhead of using sectors. In fact, EH has even outperformed SH with a single sector due to the lack of an exclusion zone along the North direction. Unfortunately, the negative effect predicted for EH can also be appreciated: Its runtime is significantly higher. It is because of i) the analysis of inheritance and ii) the additional step that adds the heliostats that could not be cloned. For instance, SH took 51228 s with three sectors on average while EH needed 93556 s. The peak is with 4 sectors due to the accumulated overhead. Finally, Fig. 7 (left) contains the best field obtained with EH, whose efficiency is 0.6949. It is similar to the one designed with SH and plotted in Fig. 6 (right). The design maintains the overall symmetry. However, it is not perfect anymore, which can be easily appreciated near the North. Moreover, as intended, the separation between sectors is slightly more difficult to track. In fact, EH achieves more homogeneity and density in front of the receiver.

Additionally, to compare these results with external tools, a field has been designed with the popular and free software package SAM (System Advisor Model) [37, 38], version 2017.9.5. This tool can design a radial-staggered pattern for a given region and receiver. The resulting field is shown in Fig. 7 (right). As can be seen, it tries to concentrate as many heliostats as possible near the receiver before increasing the spacing progressively. The efficiency

of that field according to the model implemented is 0.6597, which means around 249.44 MW on the receiver after the $T = 36$ instants. According to these results, the field of SAM is not the best option. Even the results of EnGA alone outperform it, probably because of the flexibility added to patterns by working with coordinates. Nevertheless, EnGA still converges to worse results when addressing the whole problem alone rather than within SH and EH.

5. Conclusions and future work

This work starts with a brief review of the main strategies for heliostat field optimization. After that, it focuses its interest on coordinate-based field optimization as the most powerful technique. This approach does not force heliostats to follow a pattern or belong to a set of positions, but it optimizes their coordinates on a continuous search space. However, this strategy potentially lacks applicability and scalability due to the underlying problem difficulty. The proposal of this work has been based on the hypotheses that most optimizers may not be able to find the global optimum due to the problem structure.

A new meta-heuristic, called Hector, has been designed to support coordinate-based field design with any user-given optimizer. It renounces the mathematical certainty of being able to obtain the global optimum. Instead, it splits the problem into smaller cases which are progressively shown to any selected optimizer. This method is based on the premise that, in practical terms, the same optimizer can find better fields in this way. Hector forces East-West symmetry, which reduces complexity by two. Moreover, the resulting sub-problem is further simplified by splitting one of the halves into different sectors to work with. Only half of the required heliostats are then iteratively distributed among them. The corresponding sector is re-designed while keeping its previous state as a reference. Two versions of Hector have been designed, the standard and the enhanced one. The former keeps exclusion zones between sectors to allow their independent design. The latter maintains the independence of optimization too. However, it considers the border of any surrounding sector as read-only information and does not require the use of exclusion zones. Thus, the evaluations of sectors within the enhanced version are more computationally expensive but also more precise.

According to the experiments, both versions of Hector successfully help EnGA, which is a recent genetic optimizer for continuous heliostat field op-

timization, to find better fields. The standard version was given approximately the same runtime to achieve it while the enhanced one maintained the same logical configuration. The latter was the best performing version but at the expense of higher runtime. Thus, Hector seems interesting to support coordinate-based optimization with third-party optimizers. Its enhanced version is the best choice as long as its higher runtime can be assumed with large problems. Otherwise, the standard version is also capable of producing good designs, slightly worse, but with a reduced effort. Additionally, it is interesting to mention that both EnGA alone, and within Hector, outperform a field generated with SAM, a tool for pattern-based field design. This result confirms the potential benefits of continuous coordinate-based field optimization.

As future work, different optimizers could be considered for Hector. The analysis could be focused on designing larger fields with tighter configurations, i.e., not strictly focused on being compared to their independent execution. An automatic way to determine the best number of sectors could also be studied. Additionally, varying the number of heliostats deployed per iteration seems to be an interesting line of research.

Acknowledgments

This work has been funded by grants from the Spanish Ministry of Economy, Industry and Competitiveness (MTM2015-70260-P, TIN2015-66680-C2-1-R and ENERPRO DPI 2014-56364-C2-1-R), Junta de Andalucía (P12-TIC301). N.C. Cruz (FPU14/01728) is supported by an FPU Fellowship from the Spanish Ministry of Education. J.L. Redondo (RYC-2013-14174) and J.D. Álvarez (RYC-2013-14107) are fellows of the Spanish 'Ramón y Cajal' contract program, co-financed by the European Social Fund.

[1] S. Alexopoulos, B. Hoffschmidt, Advances in solar tower technology, Wiley Interdisciplinary Reviews: Energy and Environment 6 (1).

[2] J. Rea, C. Oshman, M. Olsen, C. Hardin, G. Glatzmaier, N. Siegel, P. Parilla, D. Ginley, E. Toberer, Performance modeling and techno-economic analysis of a modular concentrated solar power tower with latent heat storage, Applied Energy 217 (2018) 143–152.

[3] B. Zhao, M. Cheng, C. Liu, Z. Dai, Thermal performance and cost analysis of a multi-layered solid-PCM thermocline thermal energy storage for CSP tower plants, Applied Energy 178 (2016) 784 – 799.

[4] S. Besarati, D. Goswami, A computationally efficient method for the design of the heliostat field for solar power tower plant, Renewable Energy 69 (2014) 226–232.

[5] F. Collado, J. Guallar, Campo: Generation of regular heliostat fields, Renewable Energy 46 (2012) 49–59.

[6] A. Avila-Marin, J. Fernandez-Reche, F. Tellez, Evaluation of the potential of central receiver solar power plants: configuration, optimization and trends, Applied Energy 112 (2013) 274–288.

[7] C. Noone, M. Torrilhon, A. Mitsos, Heliostat field optimization: A new computationally efficient model and biomimetic layout, Solar Energy 86 (2) (2012) 792–803.

[8] E. Carrizosa, C. Domínguez-Bravo, E. Fernández-Cara, M. Quero, A heuristic method for simultaneous tower and pattern-free field optimization on solar power systems, Computers & Operations Research 57 (2015) 109–122.

[9] A. Ramos, F. Ramos, Strategies in tower solar power plant optimization, Solar Energy 86 (9) (2012) 2536–2548.

[10] M. Sánchez, M. Romero, Methodology for generation of heliostat field layout in central receiver systems based on yearly normalized energy surfaces, Solar Energy 80 (7) (2006) 861–874.

[11] M. Wagner, A. Newman, W. Hamilton, R. Braun, Optimized dispatch in a first-principles concentrating solar power production model, Applied Energy 203 (2017) 959 – 971.

[12] S. Lutchman, A. Groenwold, P. Gauché, S. Bode, On using a gradient-based method for heliostat field layout optimization, Energy Procedia 49 (2014) 1429–1438.

[13] F. Collado, J. Guallar, A review of optimized design layouts for solar power tower plants with campo code, Renewable and Sustainable Energy Reviews 20 (2013) 142–154.

[14] Y. Zhou, Y. Zhao, Heliostat field layout design for solar tower power plant based on GPU, IFAC Proceedings Volumes 47 (3) (2014) 4953–4958.

[15] F. Lipps, Theory of cellwise optimization for solar central receiver systems, Tech. rep., Houston Univ., TX (USA). Energy Lab. (1985).

[16] Z. Yao, Z. Wang, Z. Lu, X. Wei, Modeling and simulation of the pioneer 1MW solar thermal central receiver system in China, Renewable Energy 34 (11) (2009) 2437–2446.

[17] F. Lipps, L. Vant-Hull, A cellwise method for the optimization of large central receiver systems, Solar Energy 20 (6) (1978) 505–516.

[18] F. Siala, M. Elayeb, Mathematical formulation of a graphical method for a no-blocking heliostat field layout, Renewable Energy 23 (1) (2001) 77–92.

[19] M. Gadalla, M. Saghafifar, Thermo-economic and comparative analyses of two recently proposed optimization approaches for circular heliostat fields: Campo radial-staggered and biomimetic spiral, Solar Energy 136 (2016) 197–209.

[20] A. Mutuberria, J. Pascual, M. Guisado, F. Mallor, Comparison of heliostat field layout design methodologies and impact on power plant efficiency, Energy Procedia 69 (2015) 1360–1370.

[21] M. Zhang, L. Yang, C. Xu, X. Du, An efficient code to optimize the heliostat field and comparisons between the biomimetic spiral and staggered layout, Renewable Energy 87 (2016) 720–730.

[22] Y. Yao, Y. Hu, S. Gao, Heliostat field layout methodology in central receiver systems based on efficiency-related distribution, Solar Energy 117 (2015) 114–124.

[23] E. Carrizosa, C. Domínguez-Bravo, E. Fernández-Cara, M. Quero, An optimization tool to design the field of a solar power tower plant allowing heliostats of different sizes, International Journal of Energy Research 41 (8) (2017) 1096–1107.

[24] R. Pitz-Paal, N. Botero, A. Steinfeld, Heliostat field layout optimization for high-temperature solar thermochemical processing, Solar Energy 85 (2) (2011) 334–343.

[25] R. Buck, Heliostat field layout using non-restricted optimization, in: Proceedings of South African Solar Energy Conference, 2012.

[26] R. Buck, Heliostat field layout improvement by non-restricted refinement, Journal of Solar Energy Engineering 136 (2) (2014) 1–6.

[27] S. Kim, I. Lee, B. Lee, Development of performance analysis model for central receiver system and its application to pattern-free heliostat layout optimization, Solar Energy 153 (2017) 499–507.

[28] P. Richter, D. Laukamp, L. Gerdes, M. Frank, E. Ábrahám, Heliostat field layout optimization with evolutionary algorithms., in: 2nd Global Conference on Artificial Intelligence (GCAI), 2016, pp. 240–252.

[29] N. Cruz, S. Salhi, J. Redondo, J. Álvarez, M. Berenguel, P. Ortigosa, A parallel genetic algorithm for continuous and pattern-free heliostat field optimization, in: Proceedings of the 17th CMMSE, Vol. 2, Rota, Spain, 2017, pp. 684–694.

[30] S. Lutchman, P. Gauché, A. Groenwold, On selecting a method for heliostat field layout optimization, in: Proceedings of South African Solar Energy Conference, 2014.

[31] S. Lutchman, Heliostat field layout optimization for a central receiver, Master's thesis, University of Stellenbosch, South Africa (2014).

[32] A. Ramos, F. Ramos, Heliostat blocking and shadowing efficiency in the video-game era, arXiv preprint arXiv:1402.1690.

[33] W. Stine, M. Geyer, Power from the Sun, Online e-book. Available from `http://powerfromthesun.net/` (Last access in April, 2018), 2001.

[34] A. Johnson, Clipper – An open-source polygon clipping library, Available from `http://www.angusj.com/delphi/clipper.php` (Last access in April, 2018).

[35] N. C. Cruz, S. Salhi, J. L. Redondo, J. D. Álvarez, M. Berenguel, P. M. Ortigosa, Complementary data of the paper 'Hector: Sector-based heliostat field optimization', `http://www.hpca.ual.es/~ncalvo/ae18app/` (2018).

[36] C. Honsberg, S. Bowden, Resources of the Photovoltaic Education (PVE) network, Online e-book. Available from `http://www.pveducation.org/pvcdrom/` (Last access in April, 2018), 2014.

[37] N. Blair, A. Dobos, J. Freeman, T. Neises, M. Wagner, T. Ferguson, P. Gilman, S. Janzou, System Advisor Model, (SAM): General description, Tech. rep., National Renewable Energy Laboratory. (2014).

[38] N. Cruz, J. Redondo, M. Berenguel, J. Álvarez, P. Ortigosa, Review of software for optical analyzing and optimizing heliostat fields, Renewable and Sustainable Energy Reviews 72 (2017) 1001–1018.

[39] S. Salhi, Heuristic Search: The Emerging Science of Problem Solving, Cham: Palgrave McMillan (published by Springer), 2017.

Appendix A. Nomenclature

Latin letters

a	Apparent solar azimuth at a certain instant, (°)
A	Reflective area of every heliostat, (m^2)
c	Characteristic diameter of the heliostat model, (m)
d	Distance between a certain heliostat and the aim point, (m)
$d_{x,y}$	Distance between a certain heliostat and the receiver on the plane, (m)
D_{image}	Width of the image reflected by a heliostat at a certain time, (m)
F	Vector defining the design of a heliostat field
h	Generic heliostat in a given field
H	Number of heliostats linked to a given field
I_t	Incident solar radiation density at instant t, (kW/m^2)
l	Height of the reflective surface of heliostats, (m)
L_v	Vertical length of the image by a heliostat at a certain time, (m)
O	Optimizer selected to be used within the Hector meta-heuristic
$P_T(F)$	Power concentrated on the receiver of a field F during T instants, (kW)
$\hat{r}$	Unit vector containing the direction from a heliostat to the aim point
R_{max}	Maximum radius of the feasible area for heliostat placement
R_{min}	Minimum radius of the feasible area for heliostat placement
S	Number of sectors created by Hector
S_i	Sector i created by Hector
$\|S_i\|$	Number of heliostats in a certain sector i
$\hat{s}$	Solar ray unit vector at a certain instant t
t	Generic instant of time, i.e., apparent solar position
T	Number of instants of interest at field evaluation
w	Width of the reflective surface of heliostats, (m)
x_h	East coordinate of heliostat h, (m)
X	East direction in the solar field
y_h	North coordinate of heliostat h, (m)
Y	North direction in the solar field
z	Mount point height of the reflective surface of heliostats, (m)
Z	Zenith direction in the solar field

Greek letters

α	Apparent solar altitude at a certain instant, (°)
β	Angular limit of the field measured from the North towards the East, (°)
γ	Height of the cylindrical receiver, (m)
η_h	Optical efficiency of heliostat h at a given instant, ([0-1])
η_{cos}	Cosine factor of a certain heliostat at a given instant, ([0-1])
η_{sb}	Shading and blocking factor of a certain heliostat at a given instant, ([0-1])
η_{itc}	Interception factor of a certain heliostat, ([0-1])
η_{aa}	Atmospheric attenuation factor of a certain heliostat, ([0-1])
η_{ref}	Reflectivity factor of heliostats, ([0-1])
θ_i	Angle of incidence of solar radiation on a heliostat at a given instant, (°)
λ	Height of the central point of the cylindrical receiver, (m)
ρ	Diameter of the cylindrical receiver, (m)

Appendix B. EnGA: A genetic algorithm for continuous heliostat field optimization

Genetic algorithms are stochastic population-based optimizers [39]. They generate a set of candidate solutions (individuals) to simulate their natural evolution. Thus, their aptitude as individuals, i.e., quality as solutions of the target problem, is improved progressively. Genetic algorithms are popular for complex and black-box optimization problems due to their generality. In fact, they have also been used for heliostat field optimization, either pattern-based [4, 9] or continuous [29, 28]. The proposal of this work, Hector, has been linked to the genetic optimizer EnGA [29] to show its effectiveness. This algorithm, which has been designed for continuous and pattern-free heliostat field design, is summarized in this section for the sake of completeness. The interested reader is referred to [29], which is publicly available, for further details.

Every individual of EnGA consists of: i) an array with the coordinates of each heliostat in the field (or the sector in the framework of Hector), and ii) its quality as field design. In general, this latter value results from computing P_T for the distribution of heliostats defined by the array. However, EnGA allows the existence of infeasible solutions, such as distributions with colliding heliostats, to perform broad explorations. Since they cannot have a valid value of P_T, their aptitude is negative: The more constraints that are not satisfied, the worse fitness that is associated (penalized evaluation, see [29]).

EnGA starts by creating an initial population. Specifically, it creates and evaluates as many random individuals as desired. Only the constraints that define the valid region (either whole field or sector) are respected. Thus, some of the initial solutions might be infeasible. Next, it executes the loop described below for a given number of cycles:

Selection: It selects a given number of individuals as progenitors. Every progenitor is selected as the best individual (highest aptitude) out of a random sample of the population. The sample size is a user-given parameter too.

Reproduction: The previous progenitors are grouped into a user-given number of couples. Next, two descendants are created from every couple. For this purpose, a random binary string with as many bits as heliostats is generated. After that, the first descendant is created by

taking those heliostats from its first progenitor in which the bit is 1, or from the second one otherwise. Finally, the string is inverted and the same rule is applied to obtain the second descendant.

Mutation: Every descendant has a user-given probability of being randomly altered. When this happens, its array of heliostats is traversed and every one of them has a user-given probability of being randomly repositioned.

Replacement: At the end of every cycle, EnGA selects the individuals for the population of the next cycle. They come from either the current population or the set of descendants. The same procedure applied for the selection of progenitors is repeated. Note that a user-given number of the best ones is directly selected to avoid the risk of losing them (elitism).

Finally, EnGA returns to the field design of the best individual found.