Proceedings of the 17th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE 2017
4–8 July, 2017.

A parallel genetic algorithm for continuous and pattern-free heliostat field optimization

N.C. Cruz[1], S. Salhi[2], J.L. Redondo[1], J.D. Álvarez[1], M. Berenguel[1] and P.M. Ortigosa[1]

[1] *Dpt. of Informatics, ceiA3-CIESOL, University of Almería, Spain*

[2] *Centre for Logistics and Heuristic Optimisation (CLHO), University of Kent, Canterbury, UK*

emails: `ncalvocruz@ual.es`, `s.salhi@kent.ac.uk`, `jlredondo@ual.es`, `jhervas@ual.es`, `beren@ual.es`, `ortigosa@ual.es`

Abstract

The heliostat field of a solar power tower system, considering both its deployment cost and potential energy loss at operation, must be carefully designed. This procedure implies facing a complex continuous, constrained and large-scale optimization problem. Hence, its resolution is generally wrapped by extra distribution patterns or layouts with a reduced set of parameters. Griding the available surface is also an useful strategy. However, those approaches limit the degrees of freedom at optimization. In this context, the authors of this work are working on a new meta-heuristic for heliostat field optimization by directly addressing the underlying problem. Attention is also given to the benefits of modern High-Performance Computing (HPC) to allow a wider exploration of the search-space. Thus, a parallel genetic optimizer has been designed for direct heliostat field optimization. It relies on elitism, uniform crossover, static penalization of infeasible solutions and tournament selection.

Key words: heliostat field optimization, genetic algorithm, parallelization

1 Introduction

Solar central receiver systems, SCRS in what follows, are one the most promising flagships in the field of solar energy for large-scale electricity production. This is mainly due to the high thermodynamic efficiency and power output stability that can be achieved by these

 ISBN: 978-84-617-8694-7

systems [2, 6]. For the scope of this work, SCRS can be defined as a large set of high reflectance mirrors, called 'heliostats', and a radiation receiver on top of a tower. Heliostats feature an orientable structure and a tracking system that make them follow the apparent solar movement throughout days while also reflecting the incident radiation on the receiver. This energy is then progressively transferred to a working fluid which circulates inside the receiver. Finally, once the temperature of the fluid is high enough, it can be ultimately used in a turbine cycle to generate electricity. In Fig. 1, an illustrative depiction of this kind of systems is shown. Further information of SCRS can be found in [1, 16].

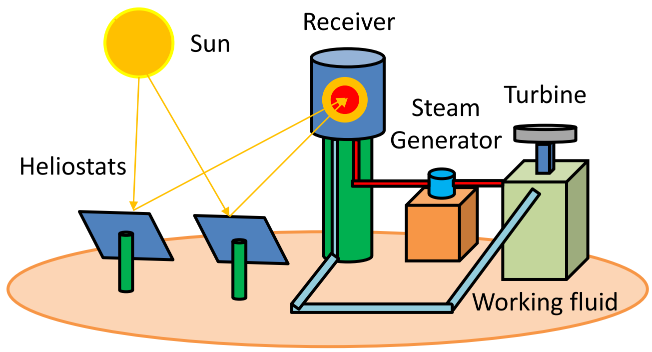

Figure 1: Scheme of a solar tower power plant.

The heliostat field supposes up to 50% of the initial investment costs and can cause up to 40% of energetic loss at operation [3, 6]. Consequently, its design must be carefully optimized. Some valid criteria to do so are the total intercepted power [8] (which is the objective chosen in this work), optical efficiency [3, 10] (which can be seen as a ratio between the really intercepted power and the theoretical maximum), land use [10, 15] and price of production [12]. Heliostat field optimization is a complex problem in which [5]: i) the coordinates of every heliostat should be directly considered (and commercial fields have, at least, several hundreds of heliostats) with placement constraints and ii) the objective function is computationally expensive and multi-modal without a direct solid approach. According to the concise and clear analysis done in [5, 8], most design methodologies relies on a) well-known parametric distribution patterns and b) selection of positions. Besides, these approaches could also be combined and expanded with what could be called 'special strategies'. Regarding known distribution patterns, there are some classic ones such as the popular radial-staggered scheme (used in [6, 10, 11]). There are also some recent and interesting proposals such as the parametric layout described in [12], which mainly expands the concept of radial staggering with more degrees of freedom. In [10], apart from trying to optimize a radial-stagger distribution, an innovative pattern is also proposed, a bio-inspired spiral later used in [3]. In relation to greedy selection of positions, the method of [15], in

 ISBN: 978-84-617-8694-7

which a grid is formed on the available surface, is widely known. The approach proposed in [5] is similar to that strategy. In fact, the concept of selection is also generally applied to pattern-based fields by generating larger layouts than needed and ultimately selecting the best heliostats as investigated in [3, 10, 11, 12]. Hence, layouts could be even seen as 'smart' ways of generating available positions. This is a good way to partially overcome the constrained perspective of parametric layouts as commented in [12]. Additionally, what has been tagged as 'special strategies', without considering some of the previous ones which also combine several ideas/steps, could contain the methods proposed in [4, 8] for their enhanced flexibility. In [4], an interesting procedure in which the refinement of a preliminary design is presented. In [8], based on some works exposing the sub-optimality of parametric approaches, the whole complexity of the problem is directly addressed with a classical optimization approach and a gradient-based method.

The authors of this work have defined a new meta-heuristic for heliostat field design. It aims to address the continuous and constrained large-scale problem of adjusting the coordinates of very heliostat through gradually altering the search-space shown to any selected optimizer. In this paper, the design of an optimizer specially designed for the problem at hand is described and commented. It is a genetic algorithm that has been designed with the main aim of keeping a simple but robust and parallel structure. We are aware that this type of algorithms have already been used for heliostat field optimization but i) their interesting theoretical principles worth its application in our specific context and, ii) their use is commonly focused on relatively reduced set of parameters as in [3, 4, 11, 12]. The underlying premise of design is: 'The more solutions can be explored per unit of time, the better results can be obtained'. In fact, the objective function is complex and requires simulating several candidate fields, which can be computationally expensive. Furthermore, as directly working with coordinates, it is expected that numerous cycles will be needed to achieve good solutions. Nevertheless, the procedure also relies on known principles to properly converge. Specifically, elitism, tournament selection and uniform crossover. Besides, points of potential knowledge-injection are also highlighted. This paper is organized as follows: In Section 2, the optimization problem is formally described in terms of maximizing the total power reflected on the receiver. Then, in Section 3, the genetic optimizer is described in detail. Finally, in Section 4, conclusions are drawn and future work is planned.

2 Problem statement

As introduced, the problem at hand consist in placing a certain number of heliostats on a flat ground so that, they reflect the maximum power on a known receiver. Let H be the total number of heliostats to deploy. All are assumed to be of the same size and specifications (as usual, to benefit from large-scale production). Specifically, their reflective surface is

 ISBN: 978-84-617-8694-7

rectangular and has a size of $l \times w$ (from length and width, respectively). Every heliostat h can be identified on the field by its central point, $C_h = (x_h, y_h)$ (assuming Cartesian coordinates). Hence, a field of H heliostats can be defined as a vector $F = (C_1, \cdots, C_H) = (x_1, y_1, \cdots, x_H, y_H)$ in $\mathbb{R}^{2H}$.

Considering the previous definitions, the problem to solve will have $2H$ dimensions, i.e., two coordinates per heliostat. Let $P_T(F)$ be the total power effectively reflected by a certain field F on the receiver throughout a fixed set of T instants of interest (i.e., defined apparent solar positions), $T = \{t_1, \cdots, t_T\}$. Depending on the final application requirements, T can vary from a single one (for design-point optimization) to many ones encompassing, for instance, a whole year (which increase the complexity and, specially, the computational cost). $P_T(F)$, which will be the objective function, can be analytically defined as expressed in Eq. (1) [8, 9].

$$P_T(F) = A \sum_{t=t_1}^{T} I_t \left(\sum_{h=h_1}^{H} \eta_h(t) \right) \tag{1}$$

In relation to A, it is the reflective area of the heliostat model (approximately $l \times w$ (m^2)) and I_t is the incident radiation density at instant t ($\mathrm{kW/m}^2$). Regarding $\eta_h(t)$, it is the instantaneous efficiency factor of heliostat h at instant t (from 0 to 1, minimum and maximum efficiencies respectively). This factor depends on i) the instant, ii) the position of heliostat h in relation to the receiver and iii) the other heliostats due to potential interactions. Its computation implies both simulating and analyzing the behavior of the candidate field. As clearly explained in [3, 10, 16], this factor models different sources of energy loss at operation. In fact, it is composed by different sub-factors (also in range [0,1]). The abstract definition selected in this work is shown in Eq. (2), and it is the same selected in [3, 9, 10].

$$\eta_h = \eta_{cos} \eta_{sb} \eta_{itc} \eta_{aa} \eta_{ref} \tag{2}$$

A brief summary of the components of Eq. (2) and the way in which they are computed is shown next:

- η_{cos} **(Cosine loss)**: The effective reflective area of a heliostat is reduced by the cosine of angle of incidence of solar radiation. It is computed as described in [16].

- η_{sb} **(Shading and blocking loss)**: Every heliostat can partially obstruct the radiation either incident (shading) or reflected (blocking) from any other one. It is computed as recently proposed in [13] with the method for candidate filtering used in [10]. This is the most computationally expensive part pf the function.

- η_{itc} **(Interception loss)**: The reflected flux map of heliostats might not perfectly fit the desired zone of the receiver. It is computed according to the model proposed in [9] (which avoids its temporal component for computational efficiency).

 ISBN: 978-84-617-8694-7

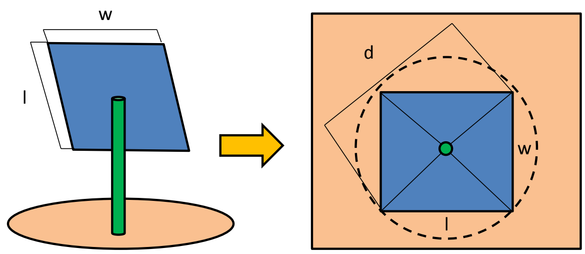

Figure 2: Characteristic safety distance for heliostats.

- η_{aa} **(Atmospheric attenuation loss)**: The atmosphere attenuates reflected radiation from heliostats along its trajectory. It is estimated by using the same model applied in [10] (which is also non-instant dependent).
- η_{ref} **(Reflectivity loss)**: Heliostats cannot grant a lossless reflection phenomenon. It is considered as a common fabrication constant as in [10].

Additionally, it must also be considered that: i) the receiver base will be at the origin of coordinates, ii) no heliostats can be neither nearer the receiver than R_{min} nor further than R_{max} and iii) heliostats should be able to freely move without colliding each other, where the safety distance d can be computed as $d = \sqrt{l^2 + w^2}$ (see Fig.2). At this point, the target optimization problem expressed in Eq. (3). Therefore, there are $H(H-1)/2 + 2H$ constrains to satisfy (as distance from any point a to b is the same that from b to a) and the potential problem dimensionality is really large.

$$\begin{aligned}
\underset{F}{\text{maximize}} \quad & P_T(F) \\
\text{subject to} \quad & \sqrt{x^2+y^2} \geq R_{min} + \frac{d}{2}, \forall C = (x,y) \\
& \sqrt{x^2+y^2} \leq R_{max} - \frac{d}{2}, \forall C = (x,y) \\
& \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq d, \forall C_i = (x_i, y_i), C_j = (x_j, y_j) : (i \neq j)
\end{aligned} \tag{3}$$

3 Method description

Genetic algorithms (GA), proposed by Holland in late seventies [7], are commonly used for complex global optimization problems. This is because their underlying theoretical principle is not linked to any particular problem but to the abstract evolution of species. Specifically,

 ISBN: 978-84-617-8694-7

a population of candidate solutions ('individuals') is generated and simulated to evolve (including interaction) until a certain halt condition met. In fact, as commented in Sec.1, they are usually considered for heliostat field optimization. In any case, the concept of GA (and even evolutionary computing in general) is quite wide and abstract. Hence, this kind of methods is ultimately adapted to the target problem, which usually determines both the selection of its base operators and their scope. Further information on population-based heuristic including GA, see [14].

In this work, a GA has been designed for the problem defined at Sec.2. Besides, as large populations are expected to be used it has the underlying aim of being able to run in parallel too. This strategy adapts the whole procedure to exploit modern high-performance environments, which is specially valuable to attenuate the potential cost of the objective function (specially for large fields and/or sets of instants).

The structure defined for the individuals is quite simple: They are vectors of length $2H$ with an additional field to record the fitness of that field design, i.e., the total power that it reflected on the receiver after simulation (evaluation of Eq. (1)). In Fig.3, the structure of individuals is depicted. Besides, it is important to remember, as included in that figure, that every pair of coordinates is linked to a certain heliostat. However, GA are mainly suited to unconstrained optimization [17] and as the problem at hand is constrained, some adaptations must be done. Specifically, the problem is treated as an unconstrained one and any unfeasible solution, i.e., those that does not respect all constraints, will be penalized with very low fitness. Penalization will only depend on the degree of violation, i.e., the more constraints are not respected (number and amount), the worse fitness is associated. This approach, which is quite common to handle constrained optimization problems with GA, is called 'static penalization' [17]. Thus, the designed method ignores the constraints shown in Eq. (3) but alter the evaluation of any candidate solution, F, as defined in Eq. 4. In that expression, m_c is the distance between heliostat c and the tower base, i.e., $m = \sqrt{x_c^2 + y_c^2}$, and I_T is the summation of solar radiation density at every instant in T. Finally, V is an abstract set which contains a record for every heliostat c in F and the constraints it violates (to compute only those factors). Every heliostat in V has also a special set linked, V_c listing any other heliostat z that is too near.

$$eval(F) = \begin{cases} P_T(F), & \text{if } V = \emptyset \\ 0 - AI_T \left(\frac{(R_{min}+d/2)-m_c}{(R_{min}+d/2)} + \frac{m_c-(R_{max}-d/2)}{m_c} \right. & \\ \qquad \left. + \frac{d-dist(C_c,C_z)}{d} \right) \;;\forall c \in V, \forall z \in V_c & \text{otherwise} \end{cases} \tag{4}$$

Once the definition of candidate solutions and how constraints are handled, the genetic procedure can be described. It takes the input parameters listed below:

- pop_{size}: the population size, which will be kept constant during the search.
- num_{pairs}: the number of pairs to form for crossover.

 ISBN: 978-84-617-8694-7

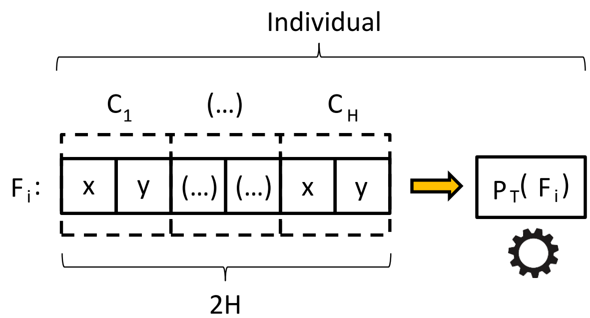

Figure 3: Diagram of an individual.

- $tourn_{size}$: the tournament size at any selection (both for crossover and replacement).
- ov_{mut}: the overall probability mutating a descendant.
- per_{mut}: the probability of altering every heliostat of a descendant once mutation started.
- *cycles*: the number of cycles, i.e., generations to run.

It must be noted that contextual information such as the description of the field, which is necessary to evaluate candidate solutions, is assumed implicitly available. With all this information, the algorithm described in Alg.1 is executed. As can be seen, it is a common evolutionary loop in which the parts that involve evaluating the objective function is distributed among concurrent threads (in a shared-memory environment). In fact, they all are forced to wait for the master to define the population of the next cycle before starting at line 12. By proceeding this way, threads can share the cost of evaluating the objective function (which is required at initialization (line 5), reproduction (line 9) and mutation (line 10)) while a consistent common population is maintained. Functions *createThreads*, *runInParallel* and *getChunkSize*, they are referred to the way in which a thread pool can be created and launched to work on different ranges of the population matrix. Similarly, tag *synchr* simple indicates that the update of that variable must be consistent. The tag *barrier_master_do* declares that the operation must be executed by the master while any other thread is forced to wait for him. Regarding the algorithmic behavior of the proposed GA, it is described next.

Function *GenerateInitialPop* (line 5) is expected to create as many candidate solutions, i.e., F vectors, as required by the population size. However, their evaluation according to Eq.4 is also required to form complete individuals. In fact, heliostats at this point are only forced to respect the constraints involving R_{min} and R_{max}, just to limit their position, but

 ISBN: 978-84-617-8694-7

Algorithm 1: Genetic algorithm for the first layer of the problem.

Input: Int $pop_{size}, num_{pairs}, tourn_{size}, cycles$, Real ov_{mut}, per_{mut}
Output: Vector F in $\mathbb{R}^{2H}$

1 IndividualSet *pop*, Individual *bestIndividual*; `/* Shared among threads */`
2 ThreadTeam $threads = createThreads()$; `/* Create a team */`
3 $threads.runInParallel()$; `/* Thread-local below: */`
4 Integers $range = getChunkSize()$; `/* Get my zone of work as a thread */`
5 $pop = GenerateInitialPop(pop_{size}, range, INJECT?)$;
6 $bestIndividual < synchr >= UpdateBest(pop)$;
7 **for** $i = 1$ ***to*** *cycles* **do**
8   IndividualSet $progs = SelectProgenitors(pop, range < num_{pairs} >, tourn_{size})$;
9   IndividualSet $desc = Reproduce(progs)$;
10   IndividualSet $descMut = Mutate(desc, ov_{mut}, per_{mut})$;
11     $bestIndividual < synchr >= UpdateBest(desc, descMut)$;
12     $pop < barrier\_master\_do >= Replace(pop, descMut, KEEP\_BEST)$;
13 **end**
14 **return** $bestIndividual.F$

collisions are not avoided. Thus, partially unfeasible solutions are possible from the very beginning. Nevertheless, its special label '$INJECT?$' declares the possibility of including some 'special individuals'. Specifically, it is referred to adding some fields obtained from any robust distribution pattern such as the biomimetic spiral proposed in [10]. This is a two-bladed option because it injects solid knowledge to the population from its origin, but also induces a serious influence in it because the fitness of those individuals will be much more higher than the other ones. Hence, it could lead to an important genetic drift and premature convergence. This option should be avoided or, at least, minimized when possible.

Function *SelectProgenitors* (line 8) simply looks for two different progenitors to form every pair. To do so, tournament selection is performed. This method is one of the most popular for GA as it can easily combine uniformity of exploration with adjustable selection pressure. Thus, every progenitor of each pair is selected out of a sample of $tourn_{size}$ participants. This is also the procedure applied in function *Replace* (line 12), in which the master selects every surviving individual out of $pop_{size} - 1$ tournaments. The previous -1 is caused because the best solution known so far is always left as part of the population (at it would not be guaranteed to participate at any tournament otherwise). This is called 'elitism' in the field of GA, and it is based on the idea that the structure of a very good solution could orientate the other ones to better zones of the search-space.

Function *Reproduce* (line 9) takes every pair and gets two descendant from each one.

 ISBN: 978-84-617-8694-7

To do so, uniform crossover is applied. This method is very popular because it features a high rate of mixing that tends to more complete explorations of the search-space. Its procedure consists in these steps: First, an auxiliary crossover mask is randomly defined. It is a binary string of length H (one per heliostat, i.e., pair of coordinates C) in which every bit had the same probability to be either a 0 or a 1. Second, a first descendant is formed by taking the heliostats (coordinates x and y) of its progenitor i for every position in which the auxiliary mask has a 1 while they taken from the progenitor $i+1$ otherwise. Third, the mask is inverted and a second descendant is obtained by applying the same rules. Any new individual must be ultimately evaluated according to Eq. 4.

Function $Mutate$ (line 10) is expected to allow the population to reach completely new zones of the search-space. To do so, every descendant has a probability of ov_{mut} of suffering any kind of mutation. Specifically, their set of heliostats is crossed and any of them has a probability of per_{mut} of being randomly repositioned. It must be noted that altered individuals must be evaluated as new ones. Besides, when applied, mutation can override promising solutions and make them worse. This is why a copy of the non-mutated ones is maintained to update the global reference of the best solution known so far.

Finally, the method returns to the best vector of coordinates defining a field that has been found during the search. That vector would be the solution that our meta-heuristic would receive for further processing.

4 Conclusions and future work

In this work, the problem of heliostat field optimization has been presented and formally stated. The field is not tried to be described by a reduced set of design variables, but the complete set of continuous coordinates is directly addressed. This approach maximizes the degrees of freedom at designing the field with respect to the use of parametric patterns. Increasing the mobility at search has been proven to lead to better designs. In fact, any final pattern can be seen just as a special case of a continuous pattern-free optimization. In this context, the authors of this paper are working on a meta-heuristic that can reduce the complexity of a complete resolution without significantly losing the achieved flexibility. However, as a meta-heuristic, it must be coupled with an optimizer to effectively solve the problem. Hence, a minimalist and parallel genetic algorithm has been designed for that purpose. Its aim is to perform a wide exploration of the search-space by using high-performance computing environments while also guiding the search. It relies on i) elitism, ii) uniform crossover, iii) static penalization and v) tournament selection. Additionally, the possibility of injecting distribution patterns as initial individuals is also allowed. It should be used with extreme care to avoid strong genetic drift and premature convergence.

Regarding future work, the most immediate one is to enhance the current preliminary implementation of the method. After that, it will included in our meta-heuristic and its real

 ISBN: 978-84-617-8694-7

performance for the problem at hand will be analyzed in depth. Hence, that some aspects might have to be further adapted.

Acknowledgements

This work has been funded by grants from the Spanish Ministry of Economy and Competitiveness (TIN2015-66680-C2-1-R and ENERPRO DPI 2014-56364-C2-1-R), Junta de Andalucía (P11-TIC7176 and P12-TIC301). Nicolás Calvo Cruz (FPU14/01728) is supported by an FPU Fellowship from the Spanish Ministry of Education. Juana López Redondo (RYC-2013-14174) and José Domingo Álvarez (RYC-2013-14107) are fellows of the Spanish 'Ramón y Cajal' contract program, co-financed by the European Social Fund.

References

[1] S. Alexopoulos and B. Hoffschmidt, *Advances in solar tower technology*, WIREs Energy Environ. **6** (2017) 1–19.

[2] A. L. Avila-Marín, J. Fernández-Reche and F. M. Tellez, *Evaluation of the potential of central receiver solar power plants: Configuration, optimization and trends*, Appl. Energ. **112** (2013) 274–288.

[3] S. M. Besarati and D. Y. Goswami, *A computationally efficient method for the design of the heliostat field for solar power tower plant*, Renew. Energ. **69** (2014) 226–232.

[4] R. Buck, A. Pfahl, and T. H. Roos, *Target aligned heliostat field layout for nonlinear flat terrain*, In Proceedings of the First Southern African Solar Energy Conference (SASEC) (2012).

[5] E. Carrizosa, C. Domínguez-Bravo, E. Fernánded.Cara and M. Quero, *An optimization approach to the design of multi-size heliostat fields.*, Tech. rep., Institute of Mathematics of University of Seville (IMUS). (2014).

[6] F. J. Collado and J. Guallar, *Campo: Generation of regular heliostat fields*, Renew. Energ. **46** (2012) 49–59.

[7] J. H. Holland, *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*, Michigan Press (1975).

[8] S. L. Lutchman, A. A. Groenwold, P. Gauché and S. Bode, *On Using a Gradient-Based Method for Heliostat Field Layout Optimization*, Energ. Proc. **49** (2014) 1429–1438.

[9] S. L. Lutchman, *Heliostat field layout optimization for a central receiver*, Master Thesis at the University of Stellenbosch, 2014.

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

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

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

[13] A. Ramos and F. Ramos, *Heliostat blocking and shadowing efficiency in the video-game era*, arXiv **86(1402.1690)** (2014).

[14] S. Salhi, *Heuristic Search: The emerging science of problem solving*, Springer, Switzerland, 2017.

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

[16] W. B. Stine and M. Geyer, *Power from the Sun*, On-line book available on http://powerfromthesun.net/, 2001. (Last accessed: 13^{th} May, 2017)

[17] Ö. Yeniay, *Penalty function methods for constrained optimization with genetic algorithms*, Math. and Comp. App. **10(1)** (2005) 45–56.

 ISBN: 978-84-617-8694-7