Data Clustering and Genetic Algorithm for the Design Optimization of a Hybrid Concentrated Solar System for SHIP

Emilie Catel[1,2], Simon Kamerling[2,3] and Valery Vuillerme[2]

[1] Ecole Polytechnique (France)

[2] Université Grenoble Alpes, CEA, LITEN, INES (France)

[3] Université Savoie Mont-Blanc, USMB (France)

Abstract

Solar thermal systems for industrial process heat could greatly contribute to reduce global greenhouse gas emissions. Such installations should be cost-effective and energy-efficient, which is why this paper investigates a new algorithm for design optimization. Design optimization first requires to accurately simulate the system, which can be very time consuming when systems are as complex as solar thermal plants with thermal storage and variable load demands. Instead of simplifying models and losing in precision, it has been chosen here to use data clustering to reduce the computational time of simulations. A clustering algorithm of meteorological data originally developed for buildings heating is adapted for solar thermal energy production and the well-spread multi-objective genetic algorithm NSGA2 is performed to find the optimal sizing. The iterative use of both algorithms is investigated to gain in accuracy.

Keywords: SHIP, solar thermal energy, industrial heat, genetic algorithm, data clustering, design optimization, control strategy

1. Introduction

The industrial sector consumes approximately one third of global world-used energy, and is responsible for about 37% of greenhouse gases emissions. Almost three quarters of the industrial energy consumption is used for heat or cold production, most often produced with electricity or combustion of oil products as explain Kumar et al. (2019).

Tilahun et al. (2019) assert that installing solar thermal energy production technologies on industrial sites can significantly reduce greenhouse gas emissions and operating costs, while saving on electricity and fuel purchases. These systems consist of solar thermal collectors that heat a Heat Transfer Fluid (HTF), as well as a thermal storage system and an auxiliary heating that take over when solar energy is not sufficient to meet process heat requirements. Solar heat production in industrial processes is generally economically viable. However, it is necessary to optimize the systems design and control strategy in order to make investments even more attractive and reduce emissions as much as possible. Indeed, as pointed out by Thiel and Stark (2021), the solar thermal plant behaves as a complex system influenced and limited by the intermittency and low areal density of the available solar radiation as well as the variability of the temperature required for the processes. In this paper, the optimization design consists in the choice of the solar technology and the type of storage to be installed as well as their dimensioning: solar collectors area and tank volume, all to deliver maximal solar energy to the industrial process while minimizing the investment costs.

Design optimization algorithms require to run performance simulations on many different plant designs, which means that these simulations should be time constrained as explains Klemeš (2011). Physical models are used for simulating the energy production of solar thermal systems, and the more detailed models are the longer is the computational time of their simulations. A compromise between model accuracy and reasonable simulation time is often made, leading to a loss of quality of the simulation results. It is therefore interesting to look for new methods which allow time simulation reduction without degradation of model accuracy. One solution would be simulating the model on a few days instead of one full year, which implies choosing days that accurately represent the meteorological year, or replicate the annual performance profile of one important performance criterion: Sayegh (2020) shows that it can be done with clustering algorithms. Indeed, clustering algorithms are methods to find relevant groups in data and determine representative data points in each group, in order to use these particular points instead of the whole dataset and reduce the length of the simulations.

As building a solar thermal plant follows multiple objectives, such as minimizing gas emissions, maximizing the solar fraction, or minimizing the investment costs, it has been chosen to use a multi-objective algorithm for design

doi:10.18086/eurosun.2022.16.13 Available at http://proceedings.ises.org

optimization. The well-spread NSGA-II (Non Dominated Sorting Algorithm) developed by Deb et al. (2002) is used, and the simulations are run with the formerly defined short sequences of days instead of the whole year. A comparison with NSGA-II run on annual simulations is made to validate the model, and modifications of the algorithm were needed in order to meet the accuracy requirements.

2. The Typical Short Sequence Algorithm

One challenge of design optimization is to evaluate the objective function on different relevant weather data, to obtain optimized solutions that cover all possible climatic conditions of the site, while limiting data to restrain computational time simulations. Reducing data without losing accuracy is necessary in multiple scientific fields and many data grouping or clustering methods have been developed in the last decades to address the problem.

2.1. Data clustering

Data clustering is the art of finding groups in data (Kaufman and Rousseeuw, 2005). One of the most spread clustering algorithms used in literature is the k-means algorithm, which is implemented in Python in the Scikit-learn module detailed by Pedregosa et al. . It divides a set of data points into a predefined number k of disjoint clusters, each represented by one *centroid*, calculated as the mean of the samples in the cluster. This implies that the centroid is not necessarily a data point in the cluster, and it allows a faster computation. However, it can be better to select data points as representative instead of imaginary points, as in the k-medoids algorithm: the *medoid* chosen is the data point that has the least total distance to the other samples of its clusters. Fig. 1 presents the illustration of a partitioning around medoids methods with 20 data points and $k = 3$ clusters.

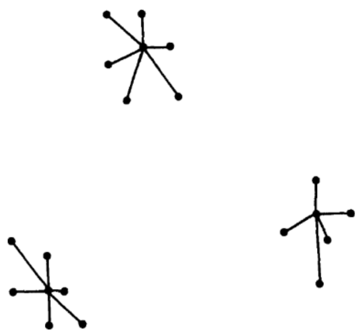

Fig. 1: Illustration of partitioning around medoids. Credit : Kaufman and Rousseeuw(2005).

Wallerand et al.(2018) used clustering methods in the design optimization of solar-assisted industrial processes. They chose the MATLAB k-medoids algorithm to obtain a reduced set of meteorological data that would be a subset of the original data. Tilahun et al. (2019) also use the MATLAB k-medoids algorithm for typical meteorological days selection, in order to optimize solar thermal energy systems design for industry. The k-medoids algorithm presents two disadvantages: it selects representative days directly on their meteorological parameters, and it is time consuming. A method that would select days according to their representativeness of the studied system performance instead of diverse meteorological parameters is more interesting, and was developed by Hasan Sayegh during his PhD: the *Typical Short Sequence* algorithm (*TypSS*).

2.2. Typical Short Sequence Algorithm

TypSS is a clustering algorithm developed by Sayegh (2020) which determines a series of typical days in order to obtain a short simulation sequence predicting the behaviour of a given system. It was developed to study buildings thermal performance, and was adapted in this paper for solar thermal industrial heat systems.
The mathematical and physical model used for simulating the systems performance is an intern tool developed by Kamerling et al. (2021) described in section 4.1.
The algorithm divides the year into distinct chronological periods iteratively and selects a representative day for each, trying to generate a reduced simulation sequence that replicates the annual sequence accurately. The code is divided into three main parts: the *initialization phase* where initial variables are calculated; the *period setting phase* where the year is iteratively divided into a predefined number of periods; and the *typical days' enhancement phase* where better representative days are searched for each period based on global performance values. During the period setting phase, at each step the performance of each period is evaluated by comparing the performance criterion value to the one of the annual simulation, and the worse performing period is divided into two halves. The global procedure of TypSS is presented in Fig. 2.

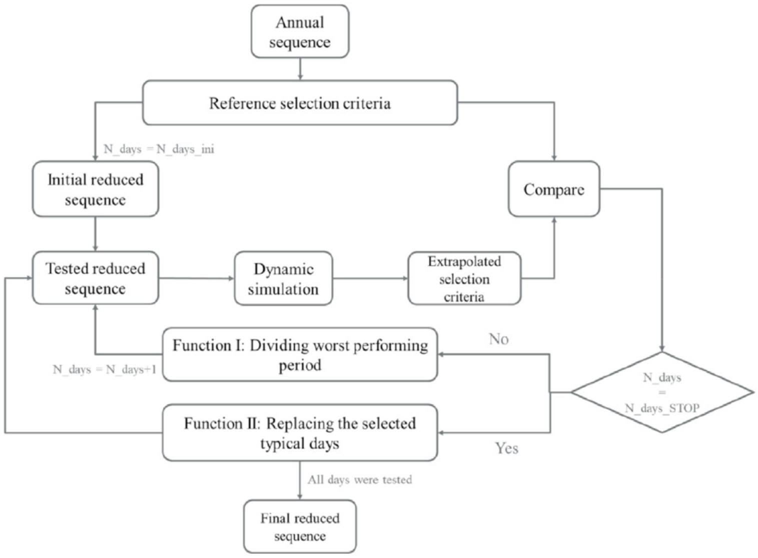

Fig. 2: Scheme of TypSS. Credit : Sayegh et al. (2022).

The parameters include all the data specified by the operator to the model and algorithm that are needed to operate. They are:

- The parametric configuration of an individual, e.g. the characteristics of the solar thermal plant tested common to all individuals and the decision variables specific to each one.
- *n_indv* the number of tested individuals.
- The weather data.
- *n_START* and *n_STOP* the lengths of the initial and final sequence.
- The chosen performance criterion;
- The days generation coefficient *coef_gen* corresponding to the percentage of days in each cluster tested by the algorithm to be the representative day.

3. Design Optimization

3.1 Optimization algorithms

According to Klemeš (2011) definition, an optimization problem consists of an objective function that corresponds to a performance criterion to be minimized or maximized (initial investment costs, fuel savings achievements...), given input parameters characterizing the model (e.g. in our case study the technologies used, typical meteorological data, thermal losses of the installations...), and decision variables whose optimal values have to be determined (e.g. total area of collectors, storage capacity...), as well as the equations modeling the underlying system.
There are a large number of optimization algorithms, which can be divided into two categories: algorithms based on the simulation and solving of a physical model, and those performing optimization via data such as machine learning. This paper investigated only the first category of algorithms, as the available data on the performance and costs of solar thermal power plants is largely insufficient to consider the second one.
Focusing on the design of a solar thermal power plant where the objective function is non-linear and highly unstable, Cabello et al. (2009) demonstrates that classical gradient-based optimization techniques such as MILP have less accurate results than the genetic algorithm he proposes.

3.2 Evolutionary algorithms

A genetic algorithm is inspired by natural evolution : Kalogirou (2004) explains that its principle rests on the evolution of a fixed-size population of solutions evolving in time, the fittest ones being selected for reproduction. It uses three principal operators: *selection*, *crossover* and *mutation*. Selection is performed thanks to a *fitness function* defined by the user for each problem: the solutions scoring best are given a higher probability of being chosen in the mating pool, where crossover happens to produce an *offspring*. Part of the new generation undergoes mutations to assure the exploration of diversified solutions. The initial population is generated randomly, and the algorithm stops when the maximal number of generations is reached or when some performance criteria are met. The *adjustable chromosomes* of the individuals are the variables that are modified during the algorithm, in this work they are: the number of loops of the solar installation and the number of storage hours.

3.3 Multi-objective optimization

In this paper, it has been chosen to optimize two objective functions : maximizing the annual solar fraction of the plant and minimizing its investment costs. A solution dominates another one if the first solution has at least one criterion strictly superior to the second solution, and no criteria strictly inferior. Mahmoudimehr and Sebghati (2019) defines an optimal or *non-dominated* solution as a solution for which there are no other possible solutions that dominate it : the solution then belongs to the Pareto front. In Fig. 3 below; the solutions A, B, C and D are not dominated by any solutions of the Solution domain, therefore they belong to the Pareto front.

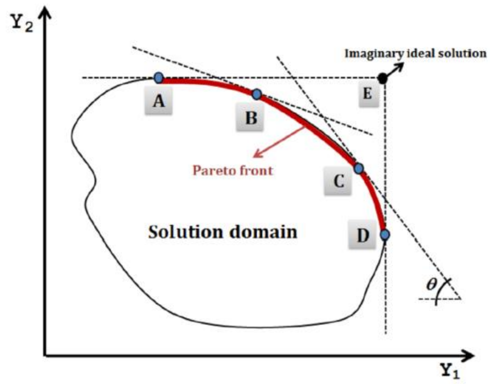

Fig. 3: Solution domain of a typical two-objective optimization problem. Credit : Mahmoudimehr and Sebghati (2019).

3.4 NSGA-II

NSGA-II is a well-spread multi-objective optimization algorithm proposed by Deb et al. (2002) that belongs to the genetic algorithms category. In this case, the fitness function is evaluated on individuals according to the two objective functions, and each solution is attributed to a Pareto Front (first front for Pareto optimal solutions, second front for the second best solutions that are only dominated by one solution and so on). They are then also sorted by their crowding distance, which is their Euclidean distance to the mean of their Pareto front. The algorithm selects for reproduction the solutions belonging to the firsts Pareto fronts and that have the highest crowding distance as can be seen in Fig. 4. This third parameter encourages the algorithm to select diversified solutions that allow better global exploration of the space solutions.

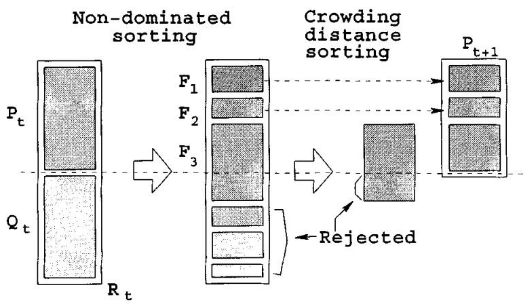

Fig. 4: NSGA-II sorting procedure. Credit : Deb et al. (2002).

Mahmoudimehr (2018) chose the NSGA-II algorithm for design optimization of a hybrid photovoltaic hydroelectric standalone energy system in Iran, with two objective functions being the investment cost and the loss of power supply probability. Silva et al. (2014) decided to apply another type of evolutionary algorithm : a memetic algorithm, which combines a genetic algorithm with a fast-local search algorithm. However, the algorithm is only single objective and three objective functions are successively studied, leading in very different optimal solutions.
In this paper, the NSGA-II algorithm is run in two different ways: the first one with evaluating the fitness functions on annual simulations for each individual, the second one with the evaluation performed on short sequences simulations, which reduces drastically the time computation. The algorithm is illustrated in Fig. 5.

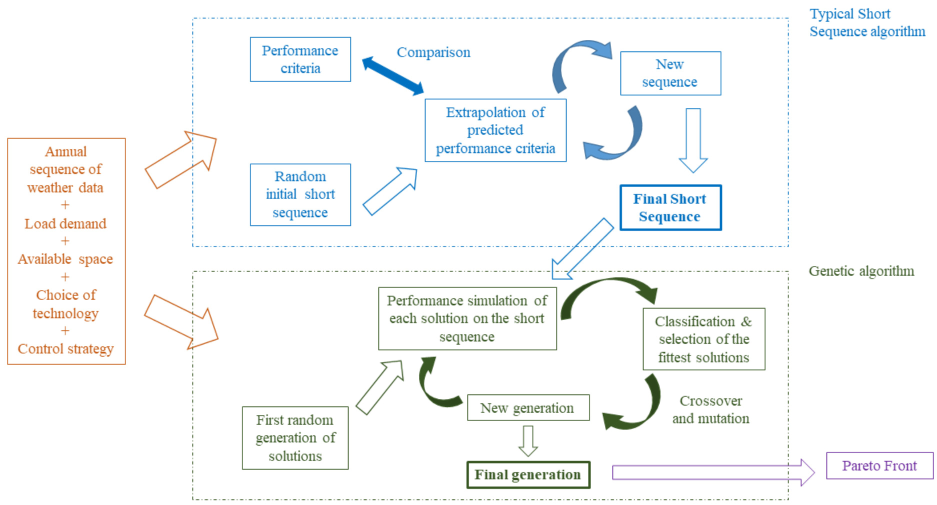

Fig. 5: NSGA-II combined with TypSS procedure.

3.5 OptiTypSS

As detailed in the PhD report of Sayegh (2020), running the NSGA-II algorithm with simulations applied on the predetermined short sequence does not give a good approximation of the Pareto Front. Indeed, the short sequence created by TypSS is designed to extrapolate the annual performance of specific individuals, chosen randomly at the start of the algorithm; therefore, it will not necessarily extrapolate well individuals near the Pareto Front. This problem was addressed by running iteratively the NSGA-II algorithm on a new short sequence determined by TypSS with added individuals chosen among the set of optimal solutions returned by the precedent iteration of NSGA-II, as shown in Fig. 6 below. Indeed, with this new approach, the second short sequence generated will have lower global accuracy but will be more precise for individuals around the first approximated Pareto front, which is supposed to be near the real Pareto front, and the second run of NSGA2 will therefore give a better approximated Pareto front, and so on.

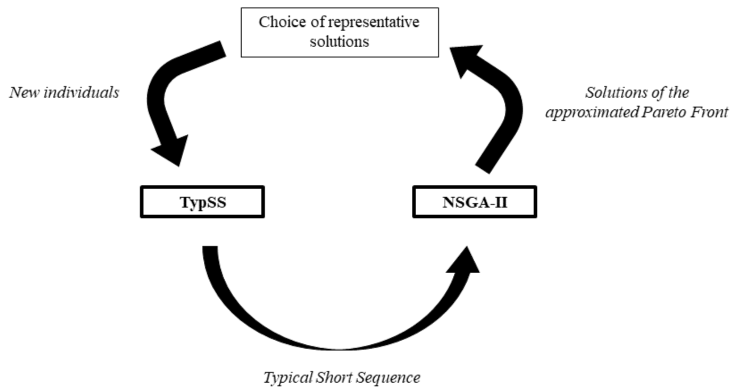

Fig. 6: OptiTypSS procedure.

4. Application to the Design Optimization of a Hybrid SHIP System

4.1 Hybrid SHIP System modeling

The Hybrid SHIP System chosen for the study is described by Kamerling et al. (2021) and illustrated in Fig. 7. It consists of a solar field, a two-tank storage and a boiler in series. Mass flow rate from the solar field ($\dot{m}_{SF}$) at the Solar Field's outlet temperature T_{SF} goes to the storage ($\dot{m}_{SF\rightarrow st}$) or to the boiler directly ($\dot{m}_{SF\rightarrow st}$). When needed, the Heat Transfer Fluid (HTF) comes from the hot tank ($\dot{m}_{st\rightarrow boiler}$) at the tank's temperature T_{st}. The storage state is characterized by the mass in the hot tank M_{st} and its temperature T_{st}, while the cold tank is not modeled. It is considered that the mass fluctuations in the hot tank are compensated by the cold tank. A third inlet coming from the cold tank permits to bypass the solar field and storage ($\dot{m}_{aux}$). The three mass flow rates at the inlet of the boiler must be equal to the process's demand mass flow rate ($\dot{m}_{demand}$), when no energy comes from the hot tank and the solar field, the boiler heats up the total HTF's mass flow rate in order to reach the process's demand temperature.

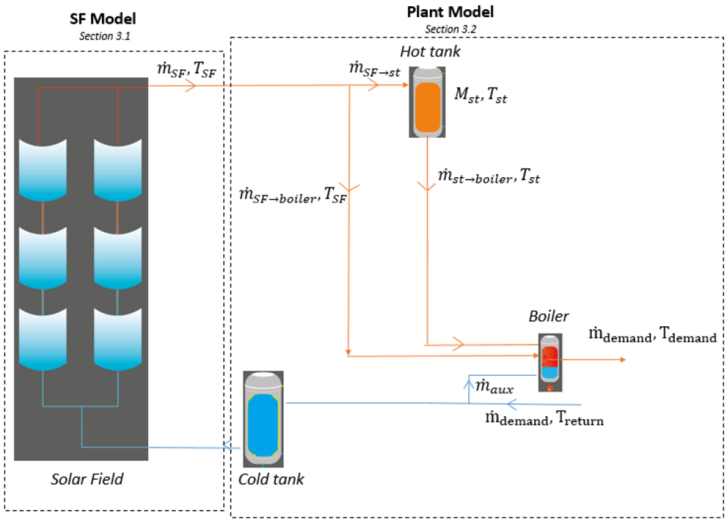

Fig. 7: Schematic of the Hybrid SHIP System of interest and control variables. Credit: Kamerling et al. (2021).

The solar field consists of a basic structure, called a *loop*, repeated a certain number of times in parallel. The loop contains low-cost parabolic trough collectors (PTC) in series with higher thermal efficiency Linear Fresnel Receivers (LFR). The two decision variables of the design optimization problem are the number of loops repeated (the number must be an integer) and the storage capacity in hours.
The algorithm *SwipeTools*, developed by S. Kamerling, calculates the solar production (i.e. heats and mass flow rates) during one year with hourly steps. It takes as input the process demand curve (hourly mass flow rates and return temperature), the meteorological data and the system design, and returns the detailed annual solar production. The main parameters taken into account in this paper are the hourly energy provided by the boiler during the full year and the annual solar fraction.

4.2. TypSS parametrization.

Several parameters were chosen based on the sensitivity analysis of the algorithm entry parameters carried out by Sayegh (2020), and are summarized in Tab. 1 below.

Tab. 1: Parametrization of TypSS.

Parameters	Values
n_indv	3
n_START	6
n_STOP	12
Performance criterion	Solar Fraction
Weather data	One typical year (Grenoble, FR)

Several number of final periods *n_STOP* were tested to find the best suited value for this case study, this parameter varying greatly between models. It has finally been chosen to determine short sequences of 12 typical days, the number realizing the best compromise between accuracy of the results and time convergence of the algorithm. Initially, the performance criterion used to compare annual and short sequence simulations was the annual solar fraction of the plant, this parameter being one of the two objective criteria in the optimization problem. However, it proved to induce a bias in the initial periods division: it led the algorithm to choose the smaller periods as the ones to be divided again. As can be seen in Fig. 8 below, it resulted in an inconsistent division of periods: five clusters contain more than 60 days while eight others contain less than 10 days, and more than half of the typical days are in July and August. This typical days repartition cannot approximate well one year of meteorological data, as it focuses on two particular months in summer, which are not representative of the variability of the resource during one year. The choice of an extensive criterion, the annual energy consumed by the boiler, instead of an intensive one solved the problem and resulted in a more coherent annual division with an even distribution of typical days during the year, as Fig. 9 shows.

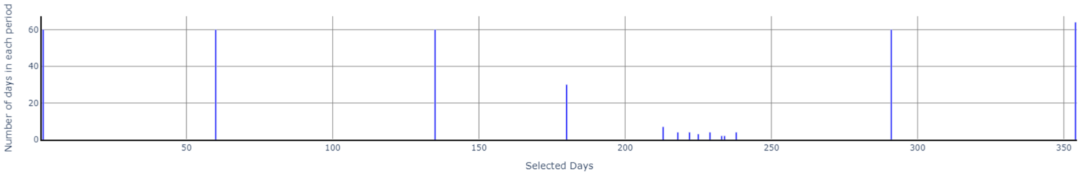

Fig. 8: Periods division and days selection for a short sequence of 14 typical days where the annual solar fraction is the performance criterion.

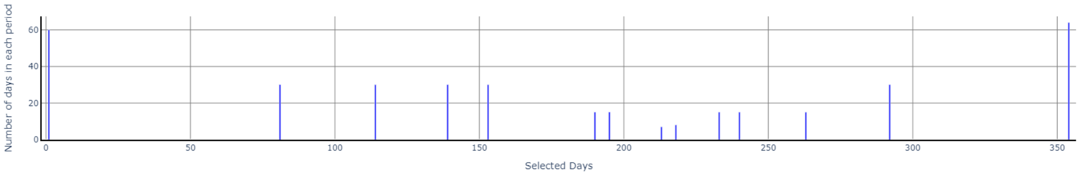

Fig. 9: Periods division and days selection for a short sequence of 14 typical days where the annual energy provided by the boiler is the performance criterion.

With these parametric configurations, the TypSS algorithm created typical short sequences that gave accurate approximation of the annual performance of the tested Hybrid SHIP System. One example of result obtain from the short sequence approximation is presented in Fig. 10. It corresponds to a plant with 88 loops and a storage capacity of 4.5 hours, belonging to the approximated Pareto front given by 3 iterations of OptiTypSS.

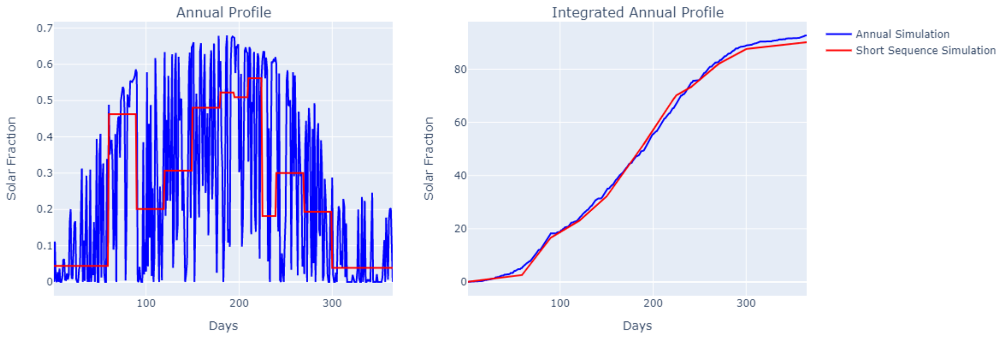

Fig. 10: Annual and integrated profiles of solar fraction with short sequence approximation

4.3. OptiTypSS Parametrization

The choice of parameters and limits for the decision variables are detailed in Tab. 2 below. The weather data was downloaded from the European Commission software PVGIS (PVGIS, 2019).

Tab. 2: Parametrization of OptiTypSS.

Decision variables	**Limits**
Number of loops (*integer*)	[|1,150|]
Storage capacity (h)	[0.5,10]
Parameters	**Values**
Population size	50
Number of parallel processes	20
Maximal number of identical generations	20
Weather data	One typical year (Grenoble, FR)

Fig. 11 presents the results of different design optimization tests with these parameters The tests were run on the same computer with 32 Go ram, with 20 sub-processes run in parallel. The NSGA-II algorithm was first run without use of short sequences: for each individual tested, the full annual system modeling was performed to calculate the annual solar fraction of the plant. It converged in 13h45 after 98 generations, and the Pareto front obtained is considered as the reference one - it is the blue curve in the figure below. After this, several iterations of OptiTypSS were tested to see the accuracy of the approximated Pareto front given. The first simulation corresponds to only one iteration of NSGA-II performed with the first typical sequence determined by TypSS on three random individuals. It is indeed further away from the reference curve than the others, with a quadratic error of 7% while the OptiTypSS algorithm performed with 2, 3 and 4 iterations reached a quadratic error of respectively 5.9%, 6% and 5%. The best approximation of the reference Pareto Front is therefore obtained by OptiTypSS with 4 iterations. However, as for certain optimization problems even 5% of quadratic error would be too much, another method for reducing time computation without losing in accuracy was investigated. An additional annual simulation was run with a different initial population : instead of generating random individuals, the initial population was taken as the resulting population of the first iteration of TypSS. The proximity of the initial population to the Pareto front led to a decrease in the number of generations needed for convergence : the algorithm stopped after 52 generations instead of 98, reducing the overall computation time of the algorithm by 6h while giving the same Pareto front. When accuracy is prior to time computation reduction, using one iteration of OptiTypSS before launching NSGA-II on annual simulations with its results as initial population still allows converging faster. A summary of time convergences and performances of the methods is provided in **Fig. *12***.

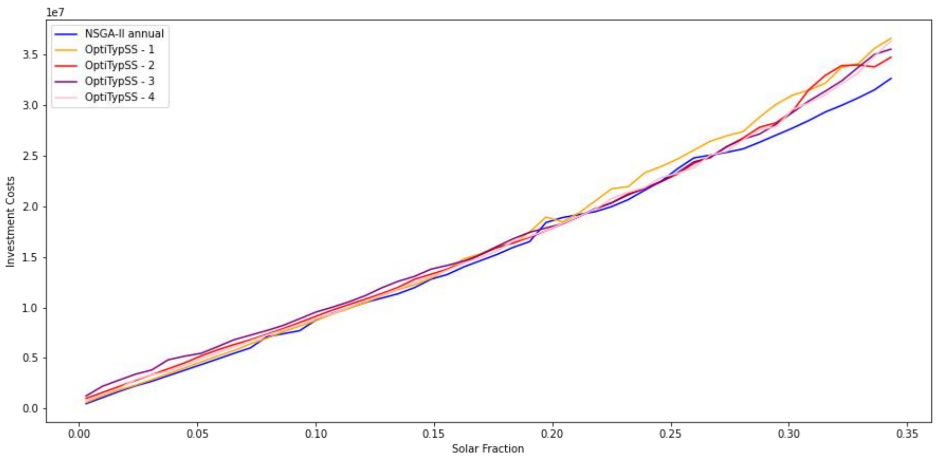

Fig. 11: Interpolations of the reference and the approximated Pareto fronts given by OptiTypSS

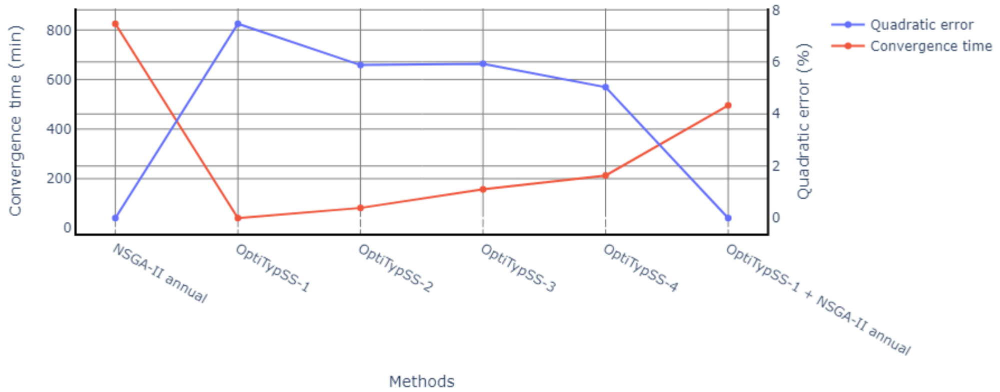

Fig. 12: Comparisons of the different algorithms performance.

4.4 Storage management and design in short sequences simulations

One important thing that was observed during the optimisation phase with OptiTypSS is that the approximated Pareto fronts were far more accurate in cases of lower investment costs than in cases of high investment costs, as shows Fig. 13. This means that for higher storage capacity and larger collectors area, the short sequences accuracy in simulation decreases. Indeed, looking at the design choices determined by the reference and OptiTypSS-4 simulations showed that for high investment costs (above 20 million euros), the storage capacity was underestimated by the optimization algorithm applied on short sequences, as presented in the Fig. 14.

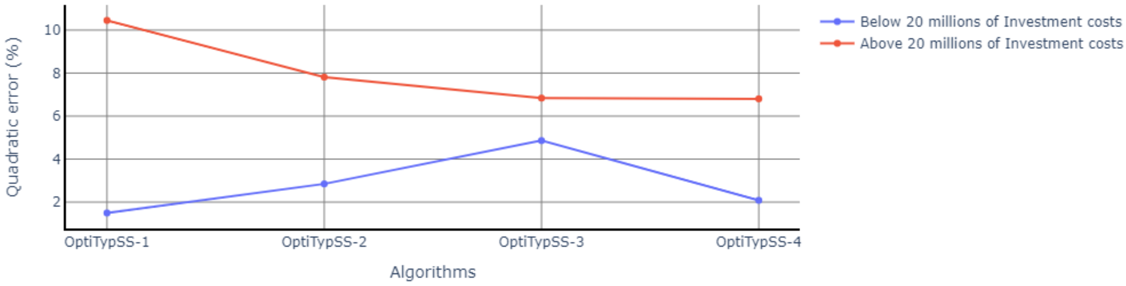

Fig. 13: Quadratic error of short sequences simulations for different ranges of investment costs.

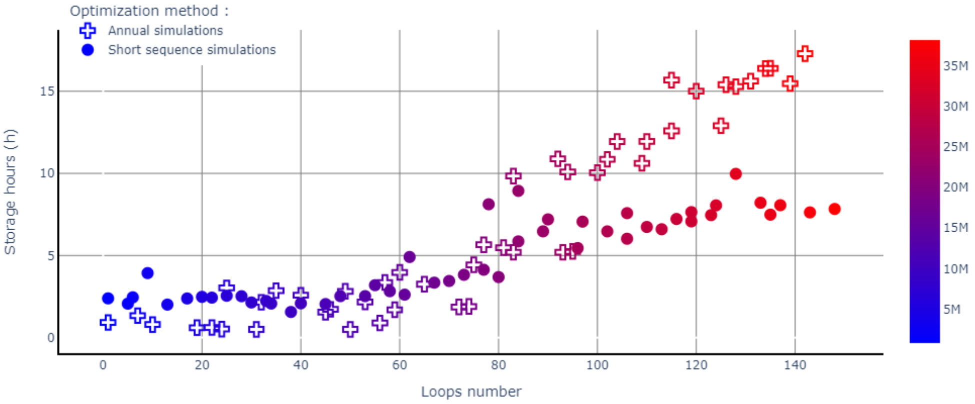

Fig. 14: Design choices difference with annual and short sequences simulations.

This outlined a bias in the short sequence algorithm : the storage management cannot be well simulated from one day to another, as the days tested consecutively do not belong to the same meteorological period. The structure of the Swipe Tools code first imposed that the storage had to be reinitialized at 0% at midnight between two days of the short sequence, which explains the limit of 8-10 hours storage recommendation in the optimization results: the code did not take into account higher capacities. A modification of the algorithm allowed to test simulations where the storage state of the precedent day of the short sequence was conserved for the beginning of the next days. Even if the days tested one after the other were not chronologically consecutive, it allowed the algorithm to take into account the possibility of using thermal energy stored even after midnight. This reduced the storage management difference between reference and clustered simulations, as shows Fig. 15. Indeed, for the fourth iteration of OptiTypSS, the global quadratic error is equal to 3.5% against 5% previously, and the quadratic error for investment costs above 20 million euros fell from 6.8% to 4.6%. It is observed in Fig. 16 that OptiTypSS still underestimates the optimal storage capacity for large collectors area (high number of loops) but far less than previously, and there is no more a storage limit of 10 hours.

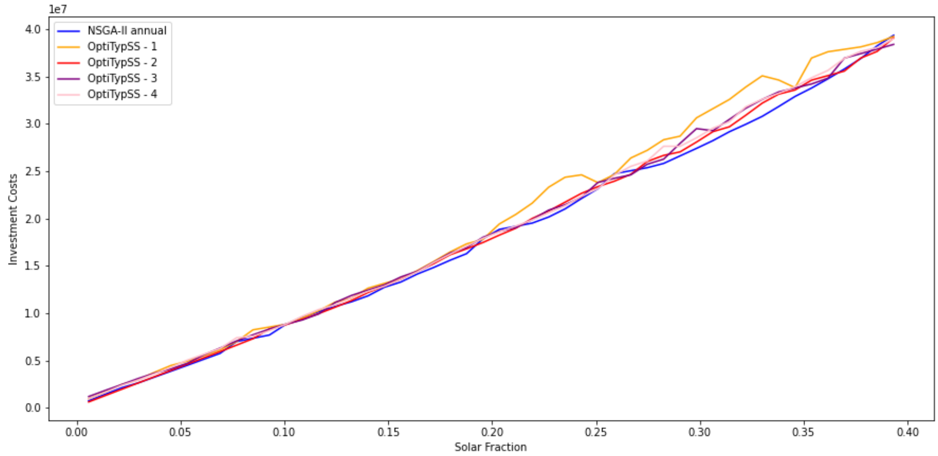

Fig. 15: Interpolations of the reference and approximated Pareto fronts given by OptiTypSS with storage state continuously modeled through the typical days.

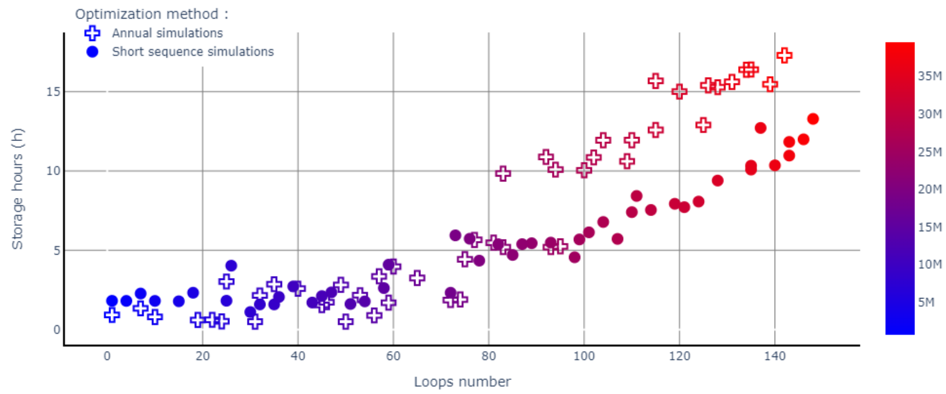

Fig. 16: Design choices comparison between annual NSGA-II and OptiTypSS with storage state continuously modeled through the typical days.

For specific plants with high storage capacity and with load demands that imply storing regularly energy through the night, the inaccuracy of storage state simulation with short sequence will have a significant impact on the quality of the results. In these cases, using the annual simulation with NSGA-II beginning with an initial population given by the first iteration of OptiTypSS could be a great compromise between time computation and results accuracy.

In the meantime, OptiTypSS is sufficiently accurate and can be used for design optimization for general types of plants. The great computation time reduction allows thinking of more complex optimization design problems, particularly by increasing the decision variables number. Indeed, more decision variables could be added such as the choice of the solar collectors technology, the design of the initial loop , or even the captors spacing and orientation.

5. Conclusion

This paper studies a new optimization method for design of solar thermal plants producing industrial heat. Instead of simplifying the model of the plant for an annual performance simulation, it investigates the use of meteorological data clustering with an intern algorithm selecting typical days in a year, named TypSS for Typical Short Sequence Algorithm. Originally developed by Sayegh (2020) for dynamic building performance simulations, it was adapted and validated in this paper for solar thermal plants‘ performance simulations. It determines the approximation of the annual solar fraction on hourly steps of a solar thermal plant with great accuracy, while reducing the computation time by 25 to 30 times depending on the short sequence length.

The NSGA-II (Non-dominated Sorted Genetic Algorithm II) well-spread optimization algorithm, presented by Deb et al. (2002), is used to obtain a Pareto front of optimal design solutions minimizing the initial investment costs while maximizing the predicted annual solar fraction. Applying this new method of performance calculation for solar plants with NSGA-II allows reducing greatly time calculation, even if it needs to regenerate more precise short sequences around the Pareto front space in order to gain in accuracy. However, one problem emerges for high storage capacities: the short sequences are not able to simulate accurately the storage state through one night, as two consecutive days of the typical sequence does not belong to the same meteorological period. It is therefore advised to use the annual simulations when the optimization design solutions looked for should make a significant use of night storage, while for plants with lower storage capacities OptiTypSS would be sufficient.

The algorithm developed allows obtaining faster many design optimization results for various problems: it can be very useful to test optimal design solutions variation with different initial parameters. Indeed, it was proved by Kamerling et al. (2021) that solar field output temperature optimization in control strategy could improve significantly the plant efficiency. It is planned to investigate whether taking into account the control strategy in design optimization would lead to greater results or if there is no need of making this choice before building the structure. This fast optimization tool will also permit running more complex design optimization problems with many more decision variables, which will be investigated in the future.

6. Acknowledgments

This work has received funding from the Ministère de l'Enseignement Supérieur, de la Recherche et de l'Innovation (MESRI), along with support from the European Union's Horizon 2020 re-search and innovation programme under grant agreement No 884213 (project FRIENDSHIP).

7. References

Cabello, J.M., Cejudo, J.M., Luque, M., Ruiz, F., Deb, K., Tewari, R., 2009. Optimization of the sizing of a solar thermal electricity plant: Mathematical programming versus genetic algorithms. In: 2009 IEEE Congress on Evolutionary Computation. Presented at the 2009 IEEE Congress on Evolutionary Computation (CEC), IEEE, Trondheim, Norway, pp. 1193–1200. DOI: 10.1109/CEC.2009.4983081

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Computat. 6, 182–197. DOI: 10.1109/4235.996017

Kalogirou, S.A., 2004. Optimization of solar systems using artificial neural-networks and genetic algorithms. Applied Energy 77, 383–405. DOI: 10.1016/S0306-2619(03)00153-3

Kamerling, S., Vuillerme, V., Rodat, S., 2021. Solar Field Output Temperature Optimization Using a MILP Algorithm and a 0D Model in the Case of a Hybrid Concentrated Solar Thermal Power Plant for SHIP Applications. Energies 14, 3731. DOI: 10.3390/en14133731

Kaufman, L., Rousseeuw, P.J., 2005. Finding groups in data: an introduction to cluster analysis, Wiley series in probability and mathematical statistics. Wiley, Hoboken, N.J. ISBN 0-47 1-73578-7

Klemeš, J.J. (Ed.), 2011. Sustainability in the process industry: integration and optimization, Green manufacturing & systems engineering. McGraw-Hill, New York. ISBN: 9780071605540

Kumar, L., Hasanuzzaman, M., Rahim, N.A., 2019. Global advancement of solar thermal energy technologies for industrial process heat and its future prospects: A review. Energy Conversion and Management 195, 885–908. DOI: 10.1016/j.enconman.2019.05.081

Mahmoudimehr, J., 2018. Optimal design of hybrid photovoltaic-hydroelectric standalone energy system for north and south of Iran. Renewable Energy 14. DOI: 10.1016/j.renene.2017.08.054

Mahmoudimehr, J., Sebghati, P., 2019. A novel multi-objective Dynamic Programming optimization method: Performance management of a solar thermal power plant as a case study. Energy 168, 796–814. DOI: 10.1016/j.energy.2018.11.079

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., n.d. Scikit-learn: Machine Learning in Python. MACHINE LEARNING IN PYTHON 6. https://hal.archives-ouvertes.fr/hal-00650905v2

PVGIS, 2019. EU SCIENCE HUB. Photovoltaic Geographical Information System [WWW Document]. URL https://ec.europa.eu/jrc/en/pvgis (accessed 6.16.21).

Sayegh, H., 2020. Holistic optimization of buildings based on the evaluation of annual performances from short simulation sequences 181. https://tel.archives-ouvertes.fr/tel-03219964v2

Sayegh, H., Leconte, A., Fraisse, G., Wurtz, E., Rouchier, S., 2022. Computational time reduction using detailed building models with Typical Short Sequences. Energy 244, 123109. DOI: 10.1016/j.energy.2022.123109

Silva, R., Berenguel, M., Pérez, M., Fernández-Garcia, A., 2014. Thermo-economic design optimization of parabolic trough solar plants for industrial process heat applications with memetic algorithms. Applied Energy 113, 603–614. DOI: 10.1016/j.apenergy.2013.08.017

Thiel, G.P., Stark, A.K., 2021. To decarbonize industry, we must decarbonize heat. Joule 5, 531–550. DOI: 10.1016/j.joule.2020.12.007

Tilahun, F.B., Bhandari, R., Mamo, M., 2019. Design optimization and control approach for a solar-augmented industrial heating. Energy 179, 186–198. DOI: 10.1016/j.energy.2019.04.142

Wallerand, A.S., Kermani, M., Voillat, R., Kantor, I., Maréchal, F., 2018. Optimal design of solar-assisted industrial processes considering heat pumping: Case study of a dairy. Renewable Energy 128, 565–585. DOI: 10.1016/j.renene.2017.07.027