

DAS Departamento de Automação e Sistemas
CTC **Centro Tecnológico**
UFSC Universidade Federal de Santa Catarina

Validation and Sensitivity Analysis of an Optimization Platform for Efficient Energy-plus Buildings

*Relatório submetido à Universidade Federal de Santa Catarina
como requisito para a aprovação da disciplina:
DAS 5511: Projeto de Fim de Curso*

Luiz Alberto SERAFIM GUARDINI

August 7, 2018

**Validation and Sensitivity Analysis of an Optimization Platform for
Efficient Energy-plus Buildings**

Luiz Alberto SERAFIM GUARDINI

Esta monografia foi julgada no contexto da disciplina

DAS 5511: Projeto de Fim de Curso

e aprovada na sua forma final pelo

Curso de Engenharia de Controle e Automação

Prof. Eduardo Camponogara

Banca Examinadora:

Dr. Frédéric Wurtz
Orientador na Empresa

Prof. Eduardo Camponogara
Orientador no Curso

Prof. Hector Bessa Silveira
Responsável pela disciplina

Dr. Paulo Renato da Costa Mendes, Avaliador

Conrado Viveiros Jordan, Debatedor

Roger Perin, Debatedor

Acknowledgements

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special gratitude I give to my parents and specially to my wife and son for keeping me company and supporting me during this journey. I would like to express my deepest gratitude towards my advisor at UFSC, Mr. Eduardo Camponogara, whose contribution in stimulating suggestions and encouragement, helped me to coordinate my project especially in writing this report. Likewise, Mr. Frederic Wurtz, CNRS enterprise advisor, and Mr. Ahmad Hably, school advisor at ENSE³ for their assistance in the project and their inspiration.

Furthermore I would also like to acknowledge with much appreciation to the staff and professors of Grenoble INP and UFSC for their guidance and for following me during my undergraduate journey. My thanks and appreciations also go to my colleagues in developing the project and people who have willingly helped me out with their abilities. I also thanks CNPq for the scholarship on Brafitec double degree program.

Abstract

Performance optimization has become a standard approach for the design and control of energy systems. This same research activity happens in the building sector. It is well known that in France, the greatest energy consuming sector is the tertiary residential sector, which makes obvious to focus the effort in new solutions, allowing an economic and environment gain.

The design of buildings is usually based on a sequential approach, which first accesses the demand reduction, then checks the energy efficiency and finally targets the use of renewable energy. When analysing all components together, as in the case of a holistic building design, it is possible to obtain a global optimum, that differs in many ways from the one obtained by segmenting the phases. In order to do that, it is necessary to optimize the design considering the building as a whole, by taking into consideration its envelope, system, occupancy and the environmental impact, all of it over the buildings life time.

There are significant difficulties on the holistic approach, for instance, calculation time due to the complexity, dynamics simulation of the building, the combinatorial explosion linked to the decision variables and parameters on the optimization, uncertainties, etc. The use of techniques such as surrogate models and parallel computing brings an aid to solve some of the issues. Since many aspects are to be considered, the optimization is multi-objective and nonlinear, however, for practical applications, the computation has to be done in reasonable time. Moreover, there is the necessity of interchangeability of different tools (for modeling, optimizing, decision making, for instance).

Project INTENSE presents a methodology and the development of a programming environment for such holistic optimization of buildings. This internship is placed at the end of the project, whereas validation for the methodology and for the developed tool has to be provided, as well as a sensitivity analysis has to be made.

Keywords: Design of Experiments, Surrogate models, Multi-objective optimization, Decision Making.

Resumo

A otimização de desempenho tornou-se uma abordagem padrão para o projeto de controle de sistemas de energia. Na França, país onde foi realizado o projeto de fim de curso, essa mesma atividade de pesquisa acontece no setor da construção Civil. Neste segmento, o maior setor consumidor de energia é o setor residencial terciário, o que torna óbvio um esforço em novas soluções, permitindo um ganho econômico e ambiental.

A concepção de prédios e edifícios baseia-se geralmente em uma abordagem sequencial, que primeiro visa a redução da demanda, depois a eficiência energética e finalmente recorre à utilização de energias renováveis. Ao analisar todos os componentes em conjunto, como no caso do projeto holístico de construção, é possível obter um ótimo global, que difere em muitos aspectos do obtido pela segmentação das fases. Para isso, é necessário otimizar o projeto considerando o edifício como um todo, levando em consideração seu revestimento e isolamento, sistema de trocas de calor, ocupação e o impacto ambiental, tudo isso durante a sua vida útil.

Existem dificuldades significativas na abordagem holística, por exemplo, o tempo de cálculo devido à complexidade, a simulação dinâmica do edifício, a explosão combinatória ligada às variáveis de decisão e parâmetros de otimização, incertezas, etc. O uso de técnicas como modelos substitutos e computação paralela auxiliam na resolução de alguns destes problemas. Como muitos aspectos devem ser considerados, a otimização é multi-objetiva e não-linear, no entanto, para aplicações práticas, a computação deve ser feita em tempo razoável. Além disso, há a necessidade de intercambiabilidade de diferentes ferramentas (para modelagem, otimização, tomada de decisão, por exemplo).

O projeto INTENSE apresenta uma metodologia e o desenvolvimento de um ambiente de programação para a referida otimização holística dos edifícios. Este estágio está situado no final do projeto, ao passo que a validação para a metodologia e para a ferramenta desenvolvida deve ser fornecida, assim como uma análise de sensibilidade.

Palavras-chave: Design de Experimentos, Modelos substitutos, Otimização multiobjetivo, Tomada de decisão.

List of Figures

Figure 1 – Illustration of the energy integration approach of the building	23
Figure 2 – Presentation of the general methodology proposed in INTENSE.	31
Figure 3 – Utilization share of major simulation programs in building optimization research.	35
Figure 4 – The MUSE standard.	36
Figure 5 – CADES user interface.	37
Figure 6 – Non-dominated solutions forming the Pareto front.	42
Figure 7 – A Taxonomy of MCDM Methods.	45
Figure 8 – Basic Three-Factor Design.	58
Figure 9 – Examples of Latin hypercube designs with $d = 2$ dimensions and $p = 20$ points.	59
Figure 10 – Latin Hypercube Design.	61
Figure 11 – Methodology adopted for the optimization using Kriging meta-model.	66
Figure 12 – (a) Rosenbrock Function; (b) Camelback function.	67
Figure 13 – Start sampling points for each phase of the algorithm.	67
Figure 14 – STD points: output after time consuming simulation.	68
Figure 15 – Kriging surfaces.	68
Figure 16 – Metamodel points: output after NSGA-II algorithm.	69
Figure 17 – Corresponding inputs for metamodel points.	69
Figure 18 – Pareto front for the toy example.	70
Figure 19 – Final Gantt graph for the internship.	73
Figure 20 – Example of inputs and outputs declaration for TRNSys and EnergyPlus.	76
Figure 21 – Interoperability with Matlab - set up.	76
Figure 22 – Interoperability with Matlab - computation.	76
Figure 23 – Case Begin: (a) Scheme of the physical system; (b) Physical model of the system.	77
Figure 24 – Modeling of case Begin into TRNSys software.	77
Figure 25 – South and East views of the building <i>Les Roches Blanches</i>	79
Figure 26 – Reference surfaces for (a) \dot{Q}_{Coll} and (b) \dot{Q}_{Aux}	82
Figure 27 – Pareto front: (a) related inputs and (b) All solutions vs Pareto formation.	82
Figure 28 – Reference surfaces with Pareto front points for (a) \dot{Q}_{Coll} , (b) \dot{Q}_{Aux} , (c) Super- position of surfaces.	83
Figure 29 – (a) Time of computation according to the number of individuals; (b) Time ratio according to the number of individuals.	84
Figure 30 – Case Begin: (a) R^2 coefficient; (b) relative RMSE.	85
Figure 31 – The Pareto front evolution in the three phases of the algorithm for different number of individuals for case Begin.	85

Figure 32 – Input points values for each phase; in blue the points from the reference Pareto, in red the solution points from the optimization.	86
Figure 33 – STD and meta-model points at the end of LHS phase.	86
Figure 34 – STD and meta-model points at the end of Global Fitting phase.	87
Figure 35 – STD and meta-model points at the end of Pareto Improvement phase.	87
Figure 36 – Pareto front evolution for case Begin with 300 individuals	88
Figure 37 – Rank of 5 best alternatives for different MCDM methods for case <i>Les Roches Blanches</i> for N = 105 individuals.	89
Figure 38 – Best TOPSIS alternatives for case Begin for N = 300 individuals for different criteria weights.	90
Figure 39 – Sensitivity analysis for different values of uncertainty.	91
Figure 40 – (a) Computation time versus the number of individuals; (b) Time ratio versus the number of individuals.	94
Figure 41 – Case <i>Les Roches Blanches</i> : (a) R^2 coefficient; (b) relative RMSE.	95
Figure 42 – The Pareto front evolution in the three phases of the algorithm for different number of individuals for case <i>Les Roches Blanches</i>	96
Figure 43 – Pareto front evolution for case <i>Les Roches Blanches</i> with 105 individuals.	96
Figure 44 – Rank of 5 best alternatives for different MCDM methods for case <i>Les Roches Blanches</i> for N = 105 individuals.	97

List of Tables

Table 1 – Decision Matrix example.	44
Table 2 – Decision matrix numerical example.	46
Table 3 – Notation experimental design matrix X	58
Table 4 – provisional planning of the 42-month project and responsibility (R)/ contribution to the tasks (orange cells).	72
Table 5 – Decision parameters for the EnergyPlus model for <i>Les Roches Blanches</i>	79
Table 6 – Rank of 5 best alternatives for different MCDM methods for case Begin for N = 300 individuals.	89
Table 7 – TOPSIS top 5 solutions for case Begin with for N = 300 individuals for different criteria weights.	89
Table 8 – Elapsed time with respect to number of inputs for the case study <i>Les Roches Blanches</i>	93
Table 9 – Rank of 5 best alternatives for different MCDM methods for case <i>Les Roches Blanches</i> for N = 105 individuals.	97

List of abbreviations and acronyms

AHP - Analytic Hierarchy Process

ANOVA - Analysis of Variance

ANR - *Agence Nationale de la Recherche* (National Research Agency)

CADES - Component Architecture for Design of Engineering Systems

DECO - Dynamic Energy, Emissions, and Cost Optimization

DoE - Design of Experiments

EIT - Energy Integration Tool

ELECTRE - *Elimination Et Choix Traduisant la REALité* (Elimination and Choice Translating Reality)

GA - Genetic Algorithms

HQE - *Haute Qualité Environnementale* (High Quality Environmental standard)

HVAC - Heat, Ventilation and Air Conditioning

INTENSE - *INTégration ENergétique des Systèmes et de l'Enveloppe des bâtiments* (Energy Integration of Systems and Buildings Envelope)

LCA - Life Cycle Analysis

LHS - Latin Hypercube Sampling

MADM - Multi-Attribute Decision Making

MCDM - Multi-Criteria Decision Making

MODM - Multi-Objective Decision Making

MOO - Multi-Objective Optimization

MOP - Multi-objective Optimization Problem

MUSE - *Modèle Unifié pour les dispositifs et Systèmes Energétiques* (Unified Model for Devices and Energy Systems)

NSGAI - Non-dominated Sorting Genetic Algorithm II

PA - Percent-Any

PT - Percent-Top

RSM - Response Surface Methodology

RMSE - Root Mean Square Error

RT2012 - *Réglementation thermique 2012* (Thermic Regulation 2012)

SIMINTHEC - *SIMulation et outils logiciels INteropérables pour la gestion des énergies THERmique et EleCTriques dans les bâtiments*

SML - Standard Meta Language

TOPSIS - Technique for Order Preference by Similarity to Ideal Solution

TRNSYS - TRaNsient SYstem Simulation

WSM - Weighted Sum Model

List of symbols

CO_2 Dióxido de Carbono

Contents

1	INTRODUCTION	19
1.1	The Project INTENSE Propositions	19
1.2	Structure	20
2	CONTEXT	21
2.1	Context, positioning and objectives of the proposal	21
2.1.1	Objectives and characteristics	21
2.2	Partners	24
2.3	State of the Art	25
2.3.1	Optimization methods	25
2.3.2	Existing energy integration tools and methods	26
2.4	Positioning of the Project	30
2.4.1	Usual approach to energy integration	30
2.4.2	Energy Integration of Buildings Systems and Envelope	31
3	METHODS	35
3.1	Software and Standards	35
3.1.1	The MUSE interchangeability standard	35
3.1.2	CADES Framework	36
3.1.3	TRNSys	37
3.1.4	EnergyPlus	38
3.2	Optimization	39
3.2.1	An optimization problem	39
3.2.2	The multi-objective optimization problem	41
3.3	Multi-Criteria Decision Making (MCDM) and Sensitivity Analysis	43
3.3.1	An overview on MCDM	43
3.3.2	Sensitivity Analysis	48
3.3.2.1	Alternative-criteria sensitivity analysis	48
3.3.2.2	Input-output sensitivity analysis	51
3.4	Meta-modeling	56
3.4.1	Design of Experiments (DoE)	57
3.4.1.1	Latin Hypercube Sampling	59
3.4.2	Response Surface Methodology	61
3.4.3	Kriging	62
3.5	The adaptive DoE	66

4	TASKS	71
5	TASKS DEVELOPMENT	75
5.1	CADES	75
5.1.1	Basic Tutorial	75
5.1.2	Interchangeability	75
5.2	Case Begin	77
5.3	Case <i>Les Roches Blanches</i>	78
5.3.1	EnergyPlus - Decision parameters and objective functions	79
6	RESULTS	81
6.1	Case Begin	81
6.1.1	Reference surfaces and reference Pareto front	81
6.1.2	Computation time analysis	83
6.1.3	The coefficient of determination and the RMSE	84
6.1.4	Pareto evolution for different number of individuals	85
6.1.5	Input points evolution	86
6.1.6	Pareto evolution for case Begin with 300 individuals	87
6.1.7	Decision making example	88
6.1.8	Alternative-Criteria Sensitivity analysis	89
6.1.9	Input-Output Sensitivity analysis	90
6.2	Case study <i>Les Roches Blanches</i> - EnergyPlus	93
6.2.1	Influence of inputs in calculation time	93
6.2.2	Reference Pareto front	93
6.2.3	Computation time analysis	94
6.2.4	The coefficient of determination and the RMSE	94
6.2.5	Pareto evolution for different number of individuals	95
6.2.6	Pareto front evolution for <i>Les Roches Blanches</i> with 105 individuals	96
6.2.7	Decision making	97
6.2.8	Sensitivity analysis	97
7	CONCLUSIONS AND FUTURE DEVELOPMENTS	99
	BIBLIOGRAPHY	101
	APPENDIX A – CADES BASIC TUTORIAL	107

1 Introduction

Over the last decades we have seen the exodus of the countryside to the cities and metropolises. Its consequence is the growth of demand in construction and energy supply in concentrated regions leading to an increase on the impact over the environment. Many actions have been taken, such as international agreements, in order to collectively take action in favor of climate and environment preservation.

In France, where this work took place, the building sector is the most energy consuming one, representing 43% of the primary energy consumption and 66% of the electrical energy consumption which makes this sector worthy improving. However, the economical appeal tied to the the building sector demand do not encourage the construction of more efficient homes. As a counter measurement, different standards have been gradually put in place to ensure new guidelines in terms of consumption and emission. This sort of regulations go against the budgetary interest, and yet, reconciling low consumption, low emission, cost control competitiveness, and occupant's comfort requires transversal skills in many areas. Thus, the search of an optimal construction parameters represents a major challenge, which is not yet completely solved.

In this context, many different actors aim to contribute to this challenge by proposing an easy-to-use tool which allows to easily determine the optimal parameters for the construction and rehabilitation of buildings. This is the research project INTENSE - INTégration ENergétique des Systèmes et de l'Enveloppe des bâtiments - or Energy Integration for Systems and Building Envelope.

1.1 The Project INTENSE Propositions

The outlines of the project INTENSE [1] is to develop a methodology and a scalable computing tool dedicated to the energy integration of buildings (envelope and energy systems) by coupling tools of dynamic simulation, multi-objective optimization (algorithms), life-cycle analysis, and decision making. It also deals with the problems related to software interchangeability.

The objective of the project is to define the methodology and elaborate an Energy Integration Tool (EIT) that leads to the best global performance defined with a multi-objective approach considering for instance, life cycle, comfort, energy consumption, financial aspects etc. From all possible combinations, the EIT will allow the end users (researchers, architects) to define a set of optimal solution regarding the system and envelope characteristics, presenting the best global efficiency taking into account the energy sources and needs. The tool will will

also offer the possibility of optimizing, with a global approach, all types of innovative systems or products related to buildings, namely in research labs or industrial.

The project will base itself into dynamic modeling, favoring important factors such as control policies, and the interaction between the occupants and the building. The method will allow simultaneous consideration of all combinations of systems and envelopes constraints. The optimization problem to be dealt with is extremely complex because of the diversity of desired solutions, from the number of iterations (systems, envelopes, exterior factors, occupants). In order to limit the calculation time, sensibility analysis will be used over the complexity of the models while the optimization methods are being adapted. The technique of micro genetic algorithms will be considered.

The EIT to be developed will endure the energy needs from the occupants (heat, cooler, electricity) using the best possible combination among the locally available sources, the possible systems and the composition of the building envelope, by interacting with the occupants and the outside environment. Knowing that it is not possible to treat all possible configurations amid the duration of the project, the tool will be scalable. Hence, the tool will be used on a case study i.e an apartment building called "Les Roches Blanches".

For the beginning of the planning of the project (2013) a design optimization tool using both global and refined approaches did not exist. The EIT is based on a very broad methodology because it can be applied to the projects in the drafting phase and over the buildings both from tertiary and residential sectors.

The usual design approach (restraint, efficacy and renewable energies) is not enough in the context of future energy-plus buildings. It will become essential to optimize the efficiency with an global and multi-objective criterion, knowing that the embodied energy corresponds to an important factor of consumption. The tools nowadays available correspond to a few bricks considering the complete energy integration aimed by INTENSE.

1.2 Structure

This report is organized as follows: Chapter 2 gives a more specific context of the project, as its objectives, state of the art, schedule, partners and explicit the context of the work done during the internship; In chapter 3 I talk about the chosen methods for the development of the tool, for instance Design of Experiments and Kriging; Chapter 4 gives specific information of the designated work to the internship; Chapter 5 talks about how the specifications on chapter 4 were accomplished; In chapter 6 I show the results achieved, and finally, on chapter 7 I conclude about the work done and describe future developments.

2 Context

Text contents of this chapter were adapted from the scientific document from project INTENSE [2], since this core document describes with precision the partners, the proposal of the project, its state of art and its positioning. The references were revised in order to guarantee their validity.

2.1 Context, positioning and objectives of the proposal

2.1.1 Objectives and characteristics

The buildings of the future (individual, collective and tertiary houses) will be energy producers and will therefore aim at optimizing the various renewable sources to meet the energy needs of the building, in relation to the activity and the occupants. The management of these different sources and storage, the comfort of the occupants in buildings, that are very well insulated thermally, will be the important aspects to consider during their design. The same applies to the environmental criteria (greenhouse gas emissions, pollutants, preservation of resources, embodied energy, life cycle analysis, etc.) and financial (total cost), which can not be neglected. This tendency to optimize globally according to performance criteria from the design phase is summed up in the notion of energy integration. In this case, global dynamic modeling is essential in order to determine the most discerning coupling of systems during the design phase, which is not limited to a simple juxtaposition. The scientific obstacles are essentially methodological. The energy integration of all systems and the building envelope is a problem of extremely complex multi-objective optimization that has never been solved with an approach as comprehensive as that envisaged in the INTENSE project:

- **Dynamic modeling** to take into account the behavior of all energy systems as well as the possible envelope¹ types of a given building. One difficulty will be to define a compromise between accuracy and calculation time knowing that the multi-objective optimization is based on the complete lifespan of the building. The building models will of course be of the multi-zone type. The number of zones considered will be the subject of a sensitivity study with respect to the performance criteria, in order to limit the calculation times. This will also be the case with the number of nodes of multi-layer models (storage tank for example). The modeling is related to the lifespan of the building. A study will also be conducted on the number of annual simulations required,

¹ The building envelope is the physical separator between the interior and exterior of a building. Components of the envelope are typically: walls, floors, roofs, fenestrations and doors. Fenestrations are any opening in the structure: windows, skylights, clerestories, etc.

and their periodicity to take into account evolution scenarios such as energy prices or renovation work. Concerning the building, its geometry is supposed to be known (defined by the architect). The envelope model will characterize the indoor environment (thermal, air and lighting) with parameters that are either optimized or fixed a priori. In the case of a real use of the EIT (Energy Integration Tool) by a consulting firm, it will determine the nature of these different parameters, as well as the possible systems for the project. The modeling related to the case studied in the INTENSE project will be quite precise (taking into account the dynamics, number of zones in the building), but nothing will prevent the user of the EIT to have models degraded because of lack of information on the building ("sketching" phase of a construction project for example). This would also make it easier and faster to use the EIT. The INTENSE project aims in fact to develop a rather general methodology, based either on very simplified models or on more complex models (consistency of the models to be respected). These will generally be considered within the EIT for research laboratories that optimize innovative systems. The parameters to be optimized can then be limited to this single system. As part of the INTENSE project, it is not planned to make an exhaustive list of all possible configurations. The systems and types of envelopes will be determined from the studied case (task 5.3 presented afterwards), in order to show the interest of the developed methodology. The EIT will be evolutionary since it will allow the integration of new systems and associated models.

- **Multi-objective global optimization** of the building: The Life Cycle Analysis, or LCA (is important in the case of buildings with very low requirements), comfort (taking into account thermal and visual comfort), energy (the main criterion is the primary energy consumed over the lifetime) and financial (overall cost estimate over the lifetime by integrating the possible aid obtained, the evolution of the building and the cost of energy). The level of comfort required for the occupants may be slightly degraded if they favor an economic gain. The number of possible combinations is relatively large, between the locally available sources, the systems and the possible envelope characteristics. It will be necessary to meet the needs of the building, taking into account both the occupancy patterns and the outdoor environment, both of which are known only during design (evolution to be predicted). The chosen optimization method is based on genetic algorithms due to the difficulty of defining a global optimum. As many configurations will be simulated over long periods, we will implement specific optimization methods such as micro genetic algorithms and/or Kriging technique to reduce computation time.
- **Decision support:** the optimization will not be based on the aggregation of several criteria but will take into account the notion of dominance (Pareto front). Depending on the nature of the decision maker, the best design will be defined from the population

of optimal solutions from the optimization phase.

- **Coupling of the different dedicated tools:** building systems and building envelope modeling, life cycle analysis (or access to databases to assess the complete life cycle of the building), multi-objective optimization and decision support .

The optimized design tool will be particularly adapted to the buildings of the future that will be energy producers by making the best use of the renewable energies available on site. The design proposed in INTENSE is much more complex than the highly sequential design considered for current buildings (reduce needs, then improve efficiency and finally use renewable energies). Figure 1 illustrates the energy integration approach over the life of the building, highlighting the performance evaluation and the evolutionary nature of the building assumptions.

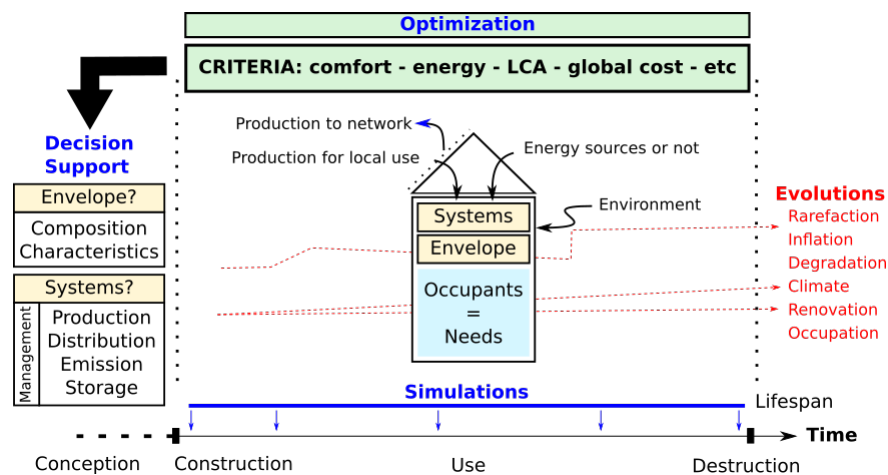


Figure 1 – Illustration of the energy integration approach of the building

The EIT will be developed in partnership with three major actors of the act of construction, and who will be potential users: the design office, the architect and the research laboratories. It will be intended for commercial buildings as well as for individual or collective housing. In construction, current "global" and quantitative studies are essentially regulatory (RT2012). They allow parametric studies with limited performance criteria (bio-climatic parameter, primary energy consumption, interior temperature comfort).

Task 5.3 of the project is the task in which the internship is inserted and it will be devoted to a case study (apartment building). The results obtained with the EIT will be compared with those of a classical design approach (type RT2012, supported by simulation tools such as Pleiades-Comfie or EnergyPlus). The validation of the EIT will consist in estimating the gain obtained compared to the classical solution. The models used will be chosen from existing libraries, ensuring that they are already validated. Their robustness will be verified on the ranges considered of variation of the parameters.

On the case of building treated in Task 5.3, the EIT will be the subject of two specific studies, as it will be shown later on chapter 3:

- a sensitivity study to improve the evaluation process. It is essential to acknowledge the uncertainties associated with the input parameters of the model.
- to take into account the impact of the occupant on overall performance.

Technological barriers are essentially linked to the "computer" coupling of different tools: modeling, optimization, decision support, databases. The contribution of G2ELAB will be important especially on the coupling between digital tools. We will thus capitalize the results obtained in part from the ANR SimintheC and PLUME [3] projects (CADES² software marketed by the start-up Vesta-System).

2.2 Partners

The diversity of tasks in terms of disciplinary fields led to the identification of seven partners with complementary skills. The contribution of each partner is thus linked to his recognized skills:

- **LOCIE** [4] - *Laboratoire d'Optimisation de la Conception et Ingénierie de l'Environnement* or Laboratory of Optimization of the Design and Engineering of the Environment in english: methodological approach to energy integration, modeling and multi-objective optimization (energy, financial cost by the global cost method, CO₂ emissions) of "classical" thermal systems coupled to buildings;
- **LGCB** [5] - *Laboratoire Génie Civil et Bâtiment* or Laboratory of Civil Engineering and Building: multi-objective optimization of the building envelope by the genetic algorithms method, modeling the thermo-ventilation behavior of buildings taking into account natural and artificial lighting, influence of occupant behavior;
- **ARMINES** [6]: life cycle analysis and environmental impact assessment with EQUER software;
- **G2ELAB** [7] - *Grenoble Génie Electrique Lab* or Grenoble Electric Engineering Lab: methods and tools for the coupling and interchangeability of models for the optimization, formulation and resolution of optimization problems of energy systems, decision support;
- **CEA-INES** [8] - *Commissariat à l'énergie atomique et aux énergies alternatives* or Alternative Energies and Atomic Energy Commission: modeling of multi-functional thermal

² Component Architecture for Design of Engineering Systems

systems dedicated to buildings with low energy needs, product development activity energy systems for housing in partnership with industrial customers;

- **ALBEDO ÉNERGIE** [9]: professional vision through the activities of sustainable design consultancies for buildings and territories around the themes Energy - Environment - Health, development of Ecoquartiers (eg. La Ravoire, Savoie), building developments with very low environmental impact;
- **FAURE (architect)**: professional vision through aspects of urban planning, architectural design (functionalities, spaces, aesthetics, sociology, etc). The daily practice of architectural design and project management makes it possible to accompany the reflection, to highlight and highlight the specific constraints of the building through the professional perspective.

2.3 State of the Art

Overall efficiency is a major objective of the energy integration approach. It is based on the implementation of efficient optimization methods to avoid the local optimum. We indicate the available approaches related to optimization and the one that seems to us the most interesting. The integration tools already available are then presented and positioned in relation to the project INTENSE.

2.3.1 Optimization methods

Optimization methods can be divided into three groups:

- **Exhaustive**: all solutions are evaluated (significant calculation time);
- **Classics**: they are based on the expression of a performance function and the evaluation of derivatives. It is not always possible to express these derivatives. In addition, the risk of this approach is to converge to a local minimum;
- **Heuristics**: Genetic algorithms (GA) are inspired by mechanisms of natural selection and genetics [10]. They reflect the principles of survival of the best adapted individuals, conservation of genetic heritage and mutation. GA eliminates some of the problems of the two previous methods. They benefit from great robustness and flexibility of use because they manipulate a binary coding of the parameters and explore the search space by means of a population of solutions (and not a single solution). The main disadvantage of GA is their slow convergence when the number of solutions to test is high. Nevertheless, convergence can be considerably accelerated by reducing the size of the population through Genetic Micro-Algorithms (μ -GA) [11], [12]. Modeling is a second important way of reducing computation time by choosing suitable models. One needs

to define a compromise between complexity and precision, by making a sensitivity study linked to the number of zones or layers defined in the models. Numerous studies related to multi-criteria optimization based on genetic algorithms have been carried out [13], [14], particularly for applications related to energy issues in the building [15] and modeling [16], [15], [17].

The issue of multicriteria optimization is linked to the consideration of contradictory criteria requiring the definition of a compromise among the "optimal" solutions for one or more criteria. In the case of evolutionary optimization methods of the GA type, the solutions generated during the optimization can be compared thanks to the notion of dominance [18]. A solution dominates a second solution if it is equivalent or more efficient for each of the objectives, and if it is more efficient for at least one of the objectives. The set of non-dominated solutions forms the Pareto front. At the end of the optimization process, a compromise between the different objectives must be found in order to retain one of the solutions constituting the Pareto front. This allows the decision maker to make an informed choice. The compromise on the solution can be obtained using a decision support tool as we envisage it in INTENSE. It is also possible to turn a multi-objective problem into a single-objective one [18]. In this case, the risk is not to explore the whole field of research, especially the Pareto front. There are methods for transforming a multi-objective problem into a single objective, the simplest is the weighted sum approach of the various performance functions. One of the partners (LGCB) notably explored this field and established the advantages and limits of the weighting by applying it to the management algorithms of hybrid ventilation systems. Another method is objective programming. In this case, reference values (objectives and anti-objectives) are defined for each of the criteria. The overall performance criterion is then defined by the generalized distance function [19];

The Kriging technique is another possibility to reduce computation time [20], [21]. One of the partners (ARMINES) has integrated this technique into the COMFIE software by means of Matlab DLLs, but it is only operational for a building with up to two zones.

2.3.2 Existing energy integration tools and methods

The DECO tool (Dynamic Energy, Emissions, and Cost Optimization) allows to optimize, from a dynamic calculation during a reference year, all types of coupled systems: fuel cells, sensors solar, wind turbines, building envelopes. The percentages of reduction of greenhouse gas emissions and the financial cost are thus optimized. The DECO tool has a much broader scope than the one envisaged in the INTENSE project since it makes it possible to compare energy choices at the regional level. The approach does not consider a fine coupling of the systems to the building and it is therefore not possible to assess for example precisely the comfort of the occupants.

Another tool, MERIT, allows energy-related profiles to be adapted to energy needs when renewable energy is used. Different coefficients are used to characterize and optimize the level of agreement between these different temporal profiles obtained from simulations (each system is modeled). The scope of the MERIT software is very wide. Its field of application extends from the building to the city. A similar approach has been used to optimize the coupling between time profiles of electrical, thermal and refrigeration requirements, and the final and renewable energy profiles required. The operation of the tool is based on the Energy Hub method. Even if the proposed method does not allow to globally optimize the building coupled to the systems with a multi-objective approach and dynamic simulations, it allows at least to optimize on a criterion (energy, environmental or financial) the combinations of possible systems. The behavior of the systems is defined in steady state considering the temporal profile of the energy needs of the building. This is an important simplification that obviously limits the accuracy of the results. The "Energy Hub" method is nevertheless very interesting because it allows to consider all combinations of systems. In the INTENSE project, we will strive to treat all configurations in the same way by activating or deactivating the models of the systems selected or not. This can easily be achieved with the binary approach related to the definition of chromosomes (GA).

In the case of a solar hot water production plant (thus multi-source), LOCIE has also developed a multi-criteria evaluation and optimization approach integrating the financial cost (total updated cost), energy consumption and CO₂ emissions (environmental approach) [16]. The solutions obtained using different optimization criteria were thus compared. This comparison highlighted the difficulty of choosing a single optimization criterion that transcribes a global performance. LOCIE has also studied different combinations of solar thermal and geothermal heat pumps (PAC). This combination exploits the excess solar energy by injecting it into the ground to promote its thermal recharge, and avoid the problems of overheating sensors. Although the approach has highlighted the complementarities between solar and geothermal systems, the overall approach to energy, economic and environmental optimization has not been addressed.

The relevance of an energy integration approach based on multicriterion dynamic optimization has been highlighted in LOCIE in the case of a relatively simple configuration of a house equipped with a double over-night cooling system, and thermoelectric double flux [22], [23]. The results were obtained thanks to the coupling of a software of dynamic simulation of the energy systems (TRNSYS) and optimization (GenOpt). The various optimization strategies envisaged have highlighted the importance of a global optimization approach, integrating both the behavior of the building and the cooling systems as well as the control-command devices. This approach is at the heart of the concerns of the INTENSE project.

A multicriteria analysis method concerning the assessment of energy consumption of buildings and the analysis of renewable energy systems has recently been developed [24].

Polynomial regression models were first identified from dynamic simulations. They make it possible to estimate the building consumption according to six parameters related to geometric, climatic and thermal criteria (inertia, losses, glazed surface). The advantage of such an approach is a quick calculation in the pre-project phase. Then, taking into account energy needs, the modeling of energy systems makes it possible to analyze energy, economic and environmental (CO_2) performances. Finally, the use of a decision support method (Electre II, III, Prométhée) allows the decision-maker to make his choice. The three steps (polynomial models, systems and decision support) were integrated into a tool (SERAD) aiming specifically at integrating renewable energy technologies with a multi-objective analysis, while focusing on fast parametric studies. The tool does not deal with the modeling of the coupling between the systems and the building; which does not allow to evaluate in particular the comfort of the occupants in summer.

LGCB, a project partner, considered a multi-objective approach for the energy renovation of buildings [15]. The developed tool MultiOpt is based on the use of existing optimization methods and the use of performance evaluation tools. The coupling of a simulation software (TRNSys), the COMIS software for aerial exchanges, two databases (environmental impacts and economic evaluation) and an optimization tool (NSGA-II - Non-dominated Sorting Genetic Algorithm) based on genetic algorithms has been realized. The MultiOpt tool optimizes the building envelope by considering the financial approach (initial investment), environmental criteria (in particular CO_2) emissions calculated with EQUER and Bilan Carbone software), energy consumption and occupant comfort. The study highlighted the limitations of a mono-objective approach given the contradictory results between the objectives. Beyond two criteria, it is much more difficult to make a choice on the selected solutions. The optimized parameters are related to the building envelope (type and composition of opaque and glazed walls) and solar protection. MultiOpt tool does not optimize the matching between available sources, systems and needs/comfort of occupants.

The partner ARMINES is a recognized laboratory among others for its activities related to life cycle analysis (LCA). The need to take into account the energy and environmental criteria throughout the life of the building (especially during its manufacture) is even more sensitive in buildings with very low energy needs [25]. ARMINES has developed, in partnership with the software company IZUBA Energies, a software platform integrating a graphic modeler (ALCYONE), a dynamic thermal simulation tool for building envelopes (COMFIE), and a LCA tool (EQUER). By its user-friendliness, this platform allows users to compare multiple variants of a project. Various studies have been conducted to investigate the reliability of the model. Comparisons between software show, with identical input data, very little difference between the results of the COMFIE simulator and the main international references such as Energy Plus and TRNSYS [26], including in the case of high-performance buildings. The deviations from measures are of the same order as those obtained with Energy Plus. Parametric studies are also proposed by the PLEIADES interface of COMFIE. Techniques of

optimization by genetic algorithm and dynamic programming have been implemented to study rehabilitation programs on a multi-criteria basis: energy, investment cost, overall cost, various environmental indicators. Constraints can be considered [27].

Project partner G2ELAB has been working for more than 30 years in the fields of optimization in electrical engineering and mechatronics. This led him to work on the development of methods and tools for optimization such as the CADES framework [28] (<http://www.cades-solutions.com>), which is now itself valued by the start-up Vesta-System (<http://www.vesta-system.com>). This software makes it possible to compile and derive energy system models, to couple them with each other and with optimization algorithms. This has led the G2ELAB to develop, on one hand, a know-how on methods and tools for the coupling and the interchangeability of the models for the optimization, and on the other hand on the formulation and the resolution of the problems of optimization of energy systems. This know-how has been applied in recent years to building energy, particularly in ANR projects such as SIMINTHEC, PLUME and FIABILITE. The INTENSE project is an opportunity to focus this work on a building-wide scale (to expand the libraries of models and algorithms, to consolidate software architectures, to consolidate its knowledge of optimization approaches, etc.). G2ELAB has already shown the interest for the approaches developed on stations with positive energy. The Lab has attained optimizers which accomplished the simultaneous optimization at a building scale of the building envelope, the capacity of the energy systems and the supervision strategy. This was achieved by optimizing a return on investment over the lifespan of the building [29].

At the scale of the city, tools (Syncity, DESDOP) make it possible to optimize the energy choices [30], [31]. They take into account criteria of financial cost and CO_2 emissions. The approach adopted in the INTENSE project is on a much smaller scale than the city to study more closely the behavior of systems coupled to a building.

With regard to overall optimized management, the MIGRER project aimed at developing a control-command system that makes the most of the potential of the different sources of renewable energy. It has shown the interest of a central control-command element able to manage the simultaneous and variable inputs of different sources of renewable energy while ensuring the energy needs of a house [32]. This control-command system could be implemented in the EIT, and optimized to take into account the full life of the building.

These different works show the interest of using optimization and modeling tools for a better design of energy systems and building envelopes. The INTENSE project will develop an innovative methodology for integrating energy systems and the building envelope. A global model will be developed in order to consider all "systems + envelope" solutions that can be envisaged (as for the Hub approach, but with "dynamic" models).

2.4 Positioning of the Project

It is often mentioned that the energy design of buildings takes place in three stages: sobriety (reduction of needs), the energy efficiency of the systems, and the use of renewable energies. This approach is not sufficient when considering the buildings of the future because it is particularly necessary to consider the life of the building given the large share of embodied energy in the context of buildings with very low needs. New buildings will even become positive-energy from 2020, thanks to the production of energy from renewable sources. The best management of this production in order to minimize the non-renewable energy surcharge, given the needs is extremely complex because it is often intermittent. Storage then becomes necessary. It is also possible to inject energy into a network or to directly consume the energy produced. The challenge of energy integration will be on the one hand to design the buildings (envelope characteristics, choice and sizing of these systems) and on the other hand to develop optimized control-command laws to minimize non-renewable energy share. The EIT developed in INTENSE will of course answer the first point. It can also be used for optimization with a global approach of predictive control-command algorithms developed as part of research on the common management of energy systems in the building. Management is of course a crucial point of overall efficiency. We will approach energy management with a classic approach in the INTENSE project to limit the scope of investigation. In addition, developments based on life cycle analysis in INTENSE may lead to steps such as HQE³, which have remained more qualitative up to now.

Energy integration recurs to optimizing globally a multi-source and multi-use system (energy needs), considering various evaluation criteria, while respecting the regulations and standards in force. As we have already said, energy integration goes beyond the traditional approach of reducing consumption based on sobriety, efficiency and renewable energies. Although this approach is a rational approach, it does not necessarily lead to a global energy, financial, environmental optimum.

2.4.1 Usual approach to energy integration

Until now, energy integration has mainly been applied to industrial processes with the pinch method, which is a graphic approach that is not adapted to the complexity of the process of designing and optimizing the energy systems associated with the building(s). The pinch method [33] aims to promote heat exchange on a given industrial site by defining on a temperature-heat power diagram, energy requirements (cold flow) and recovery opportunities (hot flow). It can be extended to other environmental, exogetic and financial factors. Nevertheless, it is not based on a dynamic analysis of phenomena. It is therefore

³ The Haute Qualité Environnementale or HQE (High Quality Environmental standard) is a standard for green building in France, based on the principles of sustainable development first set out at the 1992 Earth Summit.

not possible to correctly take into account the complexity of the interactions between the building envelope, the environment, the systems and the occupants.

2.4.2 Energy Integration of Buildings Systems and Envelope

The approach of the energy integration of the systems and the building envelope, as we foresee it in the INTENSE project is based on the coupling of tools of dynamic simulation (envelope of the building and systems), optimization and decision making tools. The general methodology is shown in Figure 2. The growing interest in developing a method of energy integration is related to the greater diversity of systems in buildings of the future that become energy producers. These systems also become more complex as in the case of multi-functional compact systems providing at the same time heating, domestic hot water, ventilation and cooling. The reduction in consumption and the desire to make investments profitable make it necessary to take into account the total lifespan of the building (i.e., LCA, total discounted cost). As we have seen, several tools have been developed at national and international levels. On the other hand, no tool allowing to meet all the following criteria has been listed:

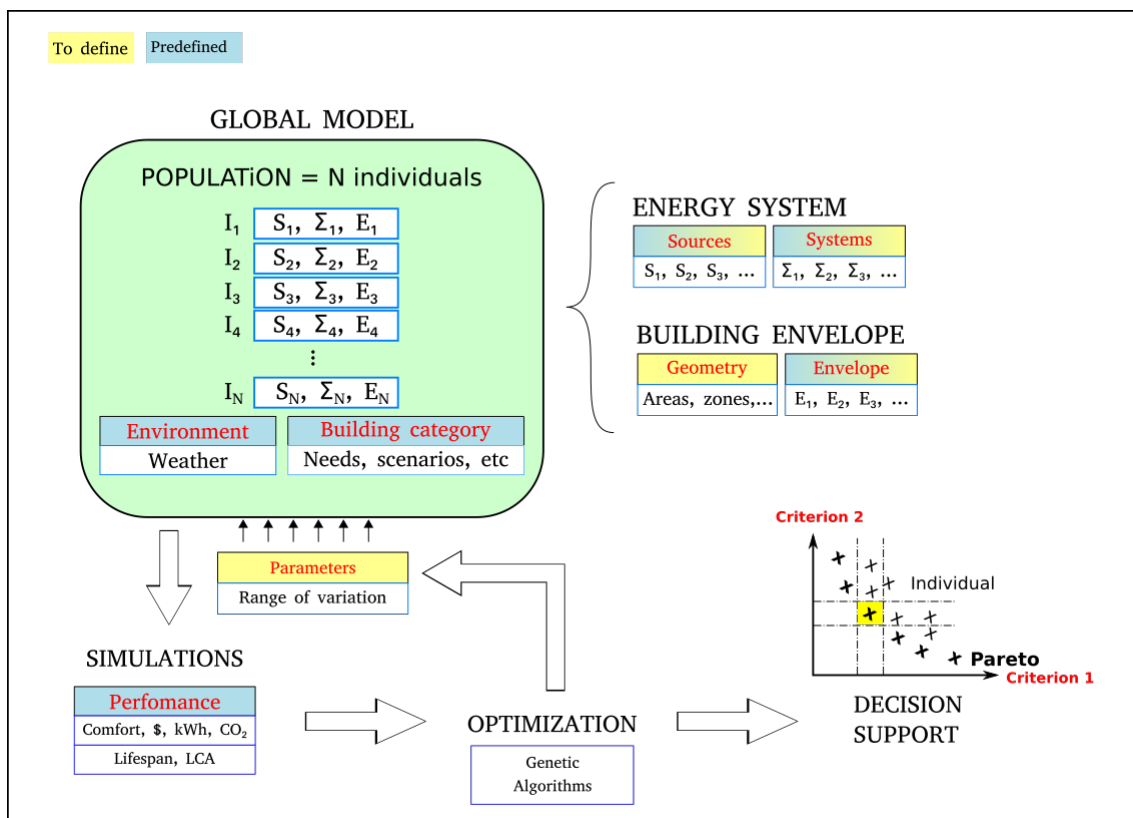


Figure 2 – Presentation of the general methodology proposed in INTENSE.

- A sufficiently fine variable-speed modeling of the systems and the building envelope to correctly evaluate the various criteria such as comfort (thermal, visual) and energy consumption;

- A global modeling of all the combinations of envelope types and systems selected in an evolutionary computer tool allowing for example to optimize new systems developed in R&D;
- A multi-criteria optimization method allowing to have a population of optimal solutions (Pareto front) which will be classified thanks to the decision-making tool according to the criteria favored by the decision-maker. The prior approach is based on genetic micro-algorithms (μ -GA) that work on small populations in order to reduce computation time [11], [34]. The application of genetic algorithms to improve energy efficiency has been the subject of a multitude of scientific works that will serve as a basis for our work [35]. Other approaches such as the Kriging technique will be studied (task 2.3) if they are more efficient (compromise between calculation time / global optimum);
- Taking into account the specificities of buildings with high energy performance. They are notably related to modeling assumptions (internal supply scenarios, airtightness, thermal bridges, etc.), to the consideration of new multifunctional compact systems and to the significant impact of gray energy over time. because of the level of insulation and energy production systems;
- Overall building energy efficiency defined by an approach that is both multi-source (renewable and non-renewable), multi-energy (thermal, electrical), multi-criteria (comfort, energy, LCA, overall cost) and multi-use (heating, cooling), electricity;
- Access to databases and tools to evaluate the complete life cycle of the building (LCA, total updated cost).

The INTENSE project is therefore adapted to the problems of buildings of the future. It will develop a more precise (modeling) and richer tool (common and evolutive platform for dynamic simulation tools, multicriterion optimization and decision support) than the available energy integration tools (further described on chapter 3). It will be a tool for evaluation, sizing and multi-criteria analysis intended for the professional world (design office, architects and industrialists), making it possible to optimize the design of the envelopes of buildings and associated systems. It will be scalable and useful for the study of specific problems in research such as the influence of the occupant, the optimization of control-command systems.

At the international level, a new task of the international energy agency entitled "Integration of energy storage with NZEB⁴: optimization and automation" is in the process of being set up. Two of the partners (LOCIE and LGCB) participate in this assembly. The project will thus develop international collaborations related to the notion of energy integration. The bearer of this new annex, Fariborz HAGHIGHAT (Concordia Research Chair - Energy & Environment, Department of Building, Civil and Environmental Engineering, Concordia

⁴ Nearly zero-energy buildings

University) will thus naturally be associated with the evolution of the INTENSE project. Collaborations are envisaged because of his expertise on multi-objective optimization of buildings, based on genetic algorithms [36].

3 Methods

As mentioned on the previous sections, the focus of the development of the software CADES (Component Architecture for Design of Engineering Systems) is not only to present a tool for optimization, but also incorporate the interchangeability among different software. In this chapter we talk about the standard used to achieve such interchangeability and also the software used during the internship to test it. After that we advance to some important definitions of optimization that are required to understand its application on the problem, and finally, we talk about meta-modeling to explain how to approach the issue of costly black-box models.

3.1 Software and Standards

This section presents the software and standards that have been used in the tasks related to the EIT CADES in this internship. The choice for TRNSys and EnergyPlus for building simulation software is justified on Figure 3, as they represent a major share of utilization on research [37].

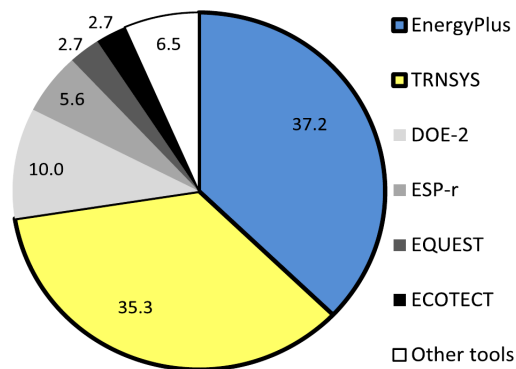


Figure 3 – Utilization share of major simulation programs in building optimization research.

Besides, the MUSE component has been chosen because of its ability of co-simulation of the dynamic models from TRNSys and EnergyPlus.

3.1.1 The MUSE interchangeability standard

The MUSE [38] component, *Modèle Unifié pour les dispositifs et Systèmes Energétiques* or Unified Model for Energy Devices and Systems, is a support for a new paradigm of models. MUSE is a standard for interchangeability, capitalization and reuse of models and algorithms. This standard offers for example the possibility of communicating models from modeling

languages such as Modelica or VHDL-AMS, with standard modeling and analysis tools such as Matlab, R, TRNSys, Pleiades/Comfie, Flux2D/3D, OpenTurns, but also with other component or service standards such as OSGi, FMI or WebServices.

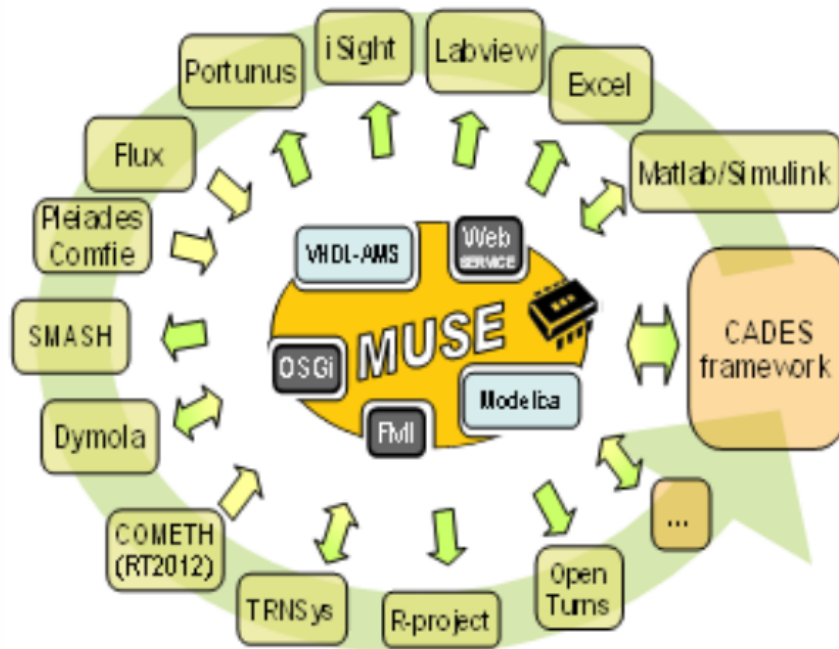


Figure 4 – The MUSE standard.

Because of its properties, it has been chosen to be the standard to achieve the interchangeability goals in the project INTENSE. By using this standard, it is possible to create a Matlab script to co-simulate a dynamic model from a TRNSys/EnergyPlus instance through CADES, situation, for instance, which was exploited to obtain an exhaustive data acquisition to form a front Pareto for the Begin case, as explained on Chapter 5.

3.1.2 CADES Framework

CADES is an optimization software developed by Vesta System. It makes it possible to manipulate Muse components as seen previously, and thus to set up interchangeability protocols. Its user interface is depicted in Figure 5.

It has several components: one to import models or create one in SML language (C++, Java and Python as well), a calculation module, an optimization module and finally a post-processing module. CADES is the main software used during this internship to run the optimizations and, analyse and observe their results. It is not the scope of this internship to present all of the software capabilities, but, since a tutorial for the new version to present some of its features had been developed during the internship, some of its components will be presented.

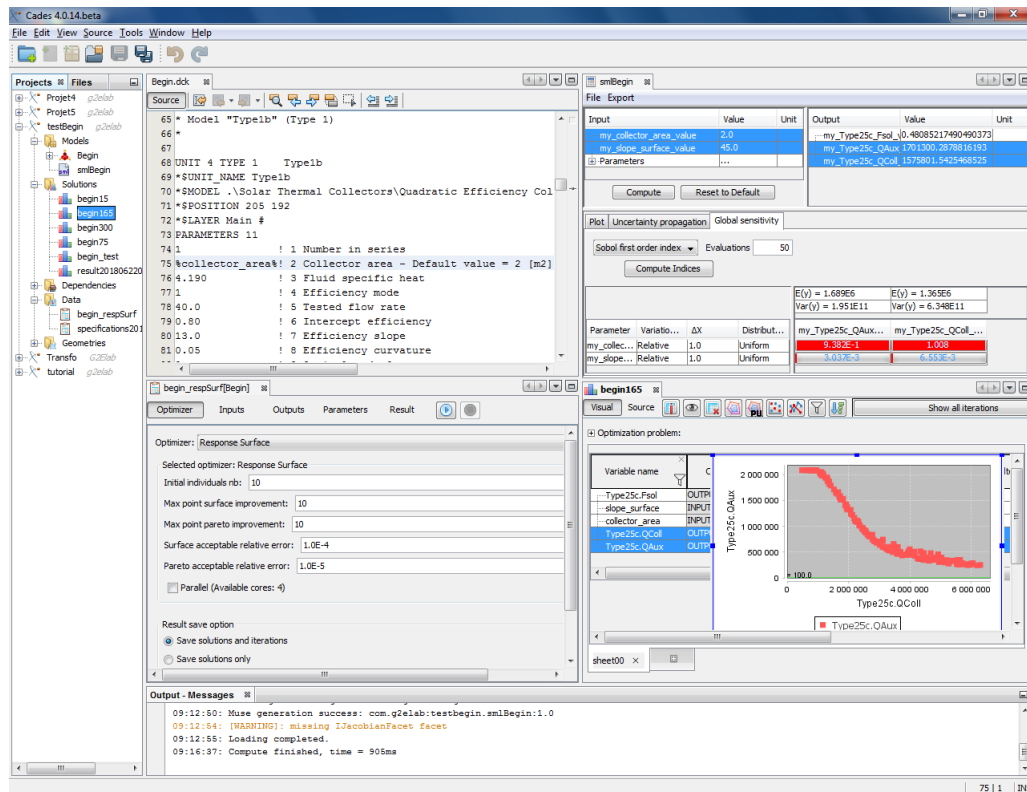


Figure 5 – CADES user interface.

3.1.3 TRNSys

TRNSys (TRaNsient SYstem Simulation) software specializes in dynamic thermal simulation applied to buildings. It is developed by the CSTB (Centre Scientifique et Technique du Bâtiment). The software integrates all the characteristics of a building and its equipment (heating systems, air conditioning) to conduct a detailed single or multi-zone study of its thermal behavior. TRNSys integrates location variables, building materials, global architecture, chosen energy concept, including the most complex such as innovative solar systems. Modular and scalable, its component database can be enriched with new energy components, systems and concepts by its user. It enables consulting firms, manufacturers and suppliers of gas and electricity to simulate the thermal performance of a building, including 3D geometry, in order to:

- reduce the time and cost of studies;
- take into account dynamic physical phenomena;
- validate the architectural and equipment choices;
- build energy efficient buildings;
- experiment innovative approaches compatible with HQE.

As described in the contextualization, the use of this software will be made in order to create the necessary models for the optimization.

3.1.4 EnergyPlus

EnergyPlus™ [39] is a whole building energy simulation program that engineers, architects, and researchers use to model both energy consumption—for heating, cooling, ventilation, lighting and plug and process loads—and water use in buildings. Some of the notable features and capabilities of EnergyPlus include:

- Integrated, simultaneous solution of thermal zone conditions and HVAC (Heat, Ventilation and Air Conditioning) system response that does not assume that the HVAC system can meet zone loads and can simulate un-conditioned and under-conditioned spaces.
- Heat balance-based solution of radiant and convective effects that produce surface temperatures thermal comfort and condensation calculations.
- Sub-hourly, user-definable time steps for interaction between thermal zones and the environment; with automatically varied time steps for interactions between thermal zones and HVAC systems. These allow EnergyPlus to model systems with fast dynamics while also trading off simulation speed for precision.
- Combined heat and mass transfer model that accounts for air movement between zones.
- Advanced fenestration models including controllable window blinds, electrochromic glazings, and layer-by-layer heat balances that calculate solar energy absorbed by window panes.
- Illuminance and glare calculations for reporting visual comfort and driving lighting controls.
- Component-based HVAC that supports both standard and novel system configurations.
- A large number of built-in HVAC and lighting control strategies and an extensible run time scripting system for user-defined control.
- Functional Mockup Interface import and export for co-simulation with other engines.
- Standard summary and detailed output reports as well as user definable reports with selectable time-resolution from annual to sub-hourly, all with energy source multipliers.

EnergyPlus is a console-based program that reads input and writes output to text files. It ships with a number of utilities including IDF-Editor for creating input files using a simple

spreadsheet-like interface, EP-Launch for managing input and output files and performing batch simulations, and EP-Compare for graphically comparing the results of two or more simulations. Several comprehensive graphical interfaces for EnergyPlus are also available. DOE does most of its work with EnergyPlus using the OpenStudio software development kit and suite of applications.

As well as TRNSys, EnergyPlus is going to be used in order to generate building models for the black-box optimization problem.

3.2 Optimization

This section presents some concepts and definitions related to optimization problems in general and then we narrow down to few definitions related to multi-objective problems, since the latter is the method mostly used during the internship.

3.2.1 An optimization problem

An optimization problem is the problem to find the best solution from all feasible solutions, meaning, the solutions that satisfy all the constraints to the referred problem.

One important notion to discuss is the concept of mathematical model, since in real life, the optimization is done upon a model. A model is a representation of our process and it has to minimally represent its desired features. We can work with two different types of models, the white-box, where we can express mathematically the static or dynamic relation between the inputs and outputs of the system, and the black-box, where we don't have the information of this relation, whereas all we know is the output for the given the inputs. Hence, for the first (white-box), we can estimate the evolution of the output given an initial position and the desired instant of time, for the latter (black-box), we need an identification in order to determine its dynamics according to given inputs. In this work, we will deal with black-box systems, and the methods of how we use Sampling Methods to reconstruct the model is given afterwards on subsection [3.4.1](#).

Usually, an optimization problem is composed of the following elements [40]:

- **decision variables:** parameters which values define a solution to the problem. They are the unknowns of a mathematical programming model and we want to determine their optimum values with an optimization method.
- **objective function(s):** a function of the decision variables to be minimized or maximized.
- **constraints:** a set of functions that defines the feasible set of solutions.

The general optimization problem is expressed in mathematical programming such as:

$$\begin{aligned}
 & \underset{x}{\text{minimize}} && f(x) \\
 & \text{subject to} && g_i(x) \leq b_i, \quad i = 1, \dots, p \\
 & && h_j(x) = 0, \quad j = 1, \dots, q. \\
 & && x \in \mathbb{S} \subset \mathbb{R}^n
 \end{aligned} \tag{3.1}$$

such that $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function or objective functions; $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$ corresponds to the inequality constraints; $h : \mathbb{R}^n \rightarrow \mathbb{R}^q$ corresponds to the equality constraints; and x is the vector of decision variables.

The subject on optimization can be extended as we wish, since we can split its concept in several types of problems. For instance, Linear Programming (LP), Integer Linear Programming (IP), Mixed Integer Linear Programming (MILP), Quadratic Programming (QP), Nonlinear Least Square, Nonlinear Programming (NLP), etc. Besides, for each of the families cited, there are several subfamilies of solution techniques and applications.

Another classification of the optimization methods is presented by Chantrelle [15], and those problems can be divided in enumerative methods, calculus-base methods and random methods.

- **Enumerative methods:** within a finite search space, or discretized infinite search space, an algorithm sequentially assesses the objective function at every point in the space. Those methods however lack real-world applicability.
- **Calculus-based methods:** based on rigorous mathematical expression of objectives, or their gradients. The main drawback of these methods is that the possibility of convergence depends on the regularity of the objective functions, which, as a result, must have an explicit expression, or permit derivatives.
- **Random (or stochastic) methods:** with these methods, no hypothesis about the regularity of the objective function is necessary, and it is the method applied to the optimization problem in this internship, since it is easier to couple them to building assessment tools.

Back to formulation (3.1), we can present some exceptions to it. First, we can have a problem without constraints, and the result is finding a maximum or a minimum for a function. It can also happen to have a problem without an objective function, which corresponds to finding the set of feasible decisions. There are also problems with multiple objectives, which corresponds to a great assortment of real problems. This last case will be discussed on the next subsection, since in this work, we will focus on the multi-objective optimization problem.

3.2.2 The multi-objective optimization problem

Traditional mono-objective optimization problems present a single best solution. However, there is a family of problems that can not represent the optimal value as a single value, since conflicting objectives can occur. Multi-Objective Optimization (MOO) provides designers with more trade-off solutions to choose from, in situations in which we aim to fulfill several (conflicting) objectives. This contrasts with traditional design, in which a single solution is obtained. The trade-off solutions that are obtained by multi-objective optimization techniques are referred to as the Pareto optimal set (in decision variable space) and their corresponding objective function values form the so-called Pareto front [41].

As an important remark, there are several methods of decomposition to transform a multi-objective problem into several mono-objective one as presented by Castillo [19]. However, not all problems can be decomposed, and the ones dealt during the internship fall into the non-decompose category. Thus, I here present some important definitions [19] that will aid the understanding of the work done during the internship.

Definition 1. *Multi-objective Optimization Problem (MOP)*

Given a vector function $\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_n(\vec{x})]$ and its feasible solution space Ω , the MOP consists in find a vector $\vec{x} \in \Omega$ that optimizes the vector function $\vec{f}(\vec{x})$.

Definition 2. *Pareto dominance*

A vector \vec{x} dominates \vec{x}' (denoted by $\vec{x} < \vec{x}'$) if $f_i(\vec{x}) \leq f_i(\vec{x}')$ for all i functions in \vec{f} and there is at least one i such that $f_i(\vec{x}) < f_i(\vec{x}')$.

Definition 3. *Pareto optimal*

A vector \vec{x}^* is Pareto optimal if does not exist a vector $\vec{x}' \in \Omega$ such that $\vec{x}' < \vec{x}^*$.

Definition 4. *Pareto optimal set*

Given a MOP, the Pareto optimal set is defined as $P^* = \{\vec{x}^* \in \Omega : \vec{x}^* \text{ is Pareto Optimal} \}$.

Definition 5. *Pareto front*

Given a MOP and its Pareto optimal set P^* , the Pareto front is defined as $PF^* = \{\vec{f}(\vec{x}) \mid \vec{x} \in P^*\}$

Definition 1 mathematically describes what has been stated so far about MOPs, as a problem with more than one objective function.

Definition 2 tells us whether a solution is better than other if it is better in one objective and equal on the other. For instance, let us consider a problem with two objective functions, and from the set of solutions, we have two solutions, \vec{x}_p and \vec{x}_q , whether \vec{x}_p is better in objective 1 but worst in objective 2, and \vec{x}_q is better in objective 2 but worst in objective

1. Then we can say that the solutions are non-dominated. It is not possible to say that one solution is better than the other.

Definition 3 states that the solutions from the Pareto optimal are the solutions that no other solution dominates.

Definitions 4 and 5 declare that the Pareto optimal set is formed by all the Pareto optimal solutions, i.e. all the non-dominated solutions. This set is known as the Pareto front and it represents the optimal solution of the MOP.

In other words, MOPs are susceptible to the "Pareto approach", which, for optimization purposes, assigns the same weight to each objective, treats them all individually and arrives at compromises between them by identifying those solutions that are non-dominated, or "Pareto Optimal". A solution is non-dominated when no alternative solution exists that will promote a particular objective without simultaneously hampering the attainment of another [15].

Figure 6 depicts the construction of the Pareto front from the overall number of solutions. We can clearly observe the formation of the Pareto front that gives the solutions to

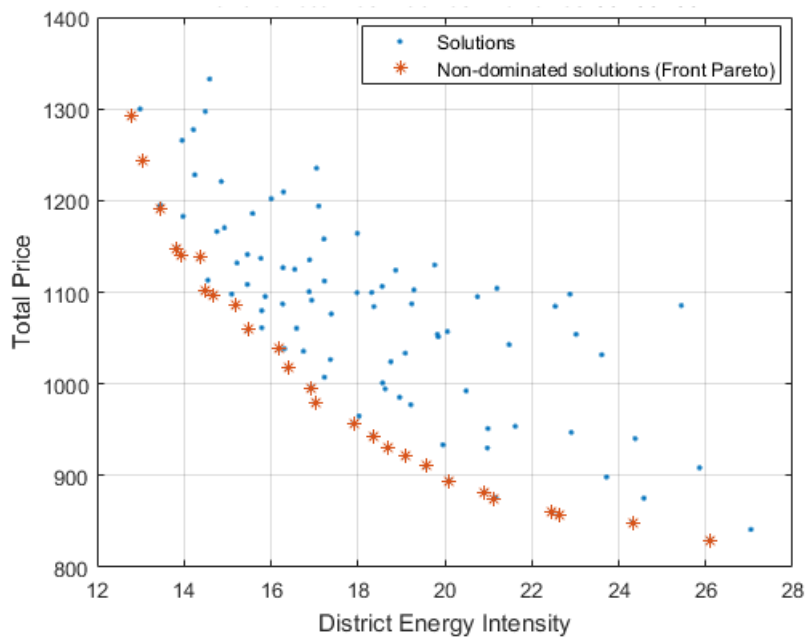


Figure 6 – Non-dominated solutions forming the Pareto front.

be used on the decision making phase.

It is also important to discuss the choice of Genetic Algorithm for the optimization. It is known that errors during the optimization process may rise from insolvable solution spaces, unfeasible combination of variables (for instance, windows area that extend the boundary of a surface), output reading errors (as in coupling of EnergyPlus), etc. A single simulation failure may crash the entire optimization process. Optimization failures caused by simulation errors

can be avoided by using evolutionary algorithms as a failed solution among a population does not impede the process. By simply rejecting the solutions having a failed simulation run, evolutionary algorithms can be surprisingly robust to high failure rates [37].

3.3 Multi-Criteria Decision Making (MCDM) and Sensitivity Analysis

The problem of how to select the best solution from the Pareto set is not trivial as it depends on a number of aspects (for example, the significance of objective functions, the demand of building investors, etc.) [37]. This process is known as multi-criteria decision making.

Moreover, once the solutions have been categorized, a study on how critical the established alternative or solution is can be made by performing a sensitivity analysis to determine whether a small change on data will modify the rank of the chosen alternative.

This subsection will present an overview on MCDM as well as on sensitivity analysis, since both steps are important on the final decision making.

3.3.1 An overview on MCDM

Multi-criteria decision making (MCDM) is one of the most well known branches of decision making. MCDM is usually divided into

- **Multi-Objective Decision Making (MODM):** studies decision problems in which the decision space is continuous, where typical example is mathematical programming problems with multiple objective functions.
- **Multi-Attribute Decision Making (MADM):** concentrates on problems with discrete decision spaces. In these problems the set of decision alternatives has been predetermined. Very often the terms MADM and MCDM are used to mean the same class of models.

We can also present the following notions present in many MCDM methods [42]:

- **Alternatives:** represent the different choices of action available to the decision maker. As in this work we analyse the output from a Pareto front, we assume the alternatives to be finite, ranging from several to hundreds. They are supposed to be screened, prioritized, and eventually ranked.
- **Multiple attributes:** the "goals" or "decision criteria". Attributes represent the different dimensions from which the alternatives can be viewed. Since different criteria represent

different dimensions of the alternatives, they may conflict with each other. For instance, cost may conflict with profit, etc. Moreover, most of the MCDM methods require that the criteria be assigned weights of importance. Usually, these weights are normalized to add up to one.

- **Decision matrix:** an MCDM problem can be easily expressed in a matrix format. A decision matrix \mathbf{A} is an $(m \times n)$ matrix in which element a_{ij} indicates the performance of alternative A_i when it is evaluated in terms of decision criterion C_j (for $i = 1, 2, 3, \dots, m$, and $j = 1, 2, 3, \dots, n$). It is also assumed that the decision maker has determined the weights of relative performance of the decision criteria (denoted as w_j for $j = 1, 2, 3, \dots, n$).

As an example, let us imagine a case depicted by the general methodology presented on Figure 2. We are concerned about the performance of some indicators, such as the comfort of the occupants, the material cost, the consumed energy, the CO_2 emission, the lifespan of the building, the LCA, etc. Each one of those indicators or attributes will represent one of more criteria (the LCA for instance can be divided in greenhouse effect, embodied energy, wastes, acidification, other atmospheric pollutants, etc), which weights are defined by the responsible for the project. The alternatives however, are the m red dots that shape the front Pareto in Figure 6. Table 1 presents a Decision Matrix example:

Table 1 – Decision Matrix example.

Alts.	Criteria			
	Comfort Occupants (w_1)	Material cost (w_2)	...	LCA (w_n)
A_1	a_{11}	a_{12}	...	a_{1n}
A_2	a_{21}	a_{21}	...	a_{2n}
\vdots	\vdots	\vdots	\ddots	\vdots
A_m	a_{m1}	a_{m1}	...	a_{mn}

A classification of MCDM is presented by Tryantaphyllou [42] (apud Chen and Hwang [1991]) and depicted in Figure 7.

There are three steps in utilizing any decision-making technique involving numerical analysis of alternatives:

1. Determine the relevant criteria and alternatives.
2. Attach numerical measures to the relative importance of the criteria and to the impacts of the alternatives on these criteria.
3. Process the numerical values to determine a ranking of each alternative.

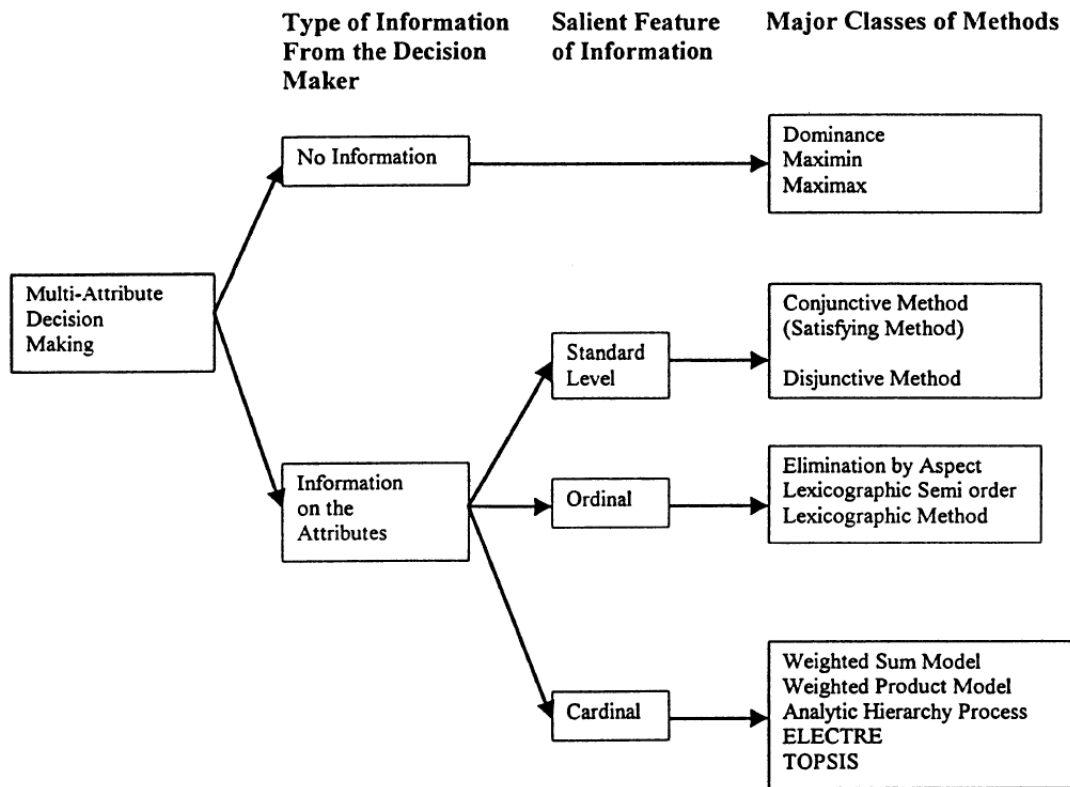


Figure 7 – A Taxonomy of MCDM Methods.

The two first steps are determined during modeling phases of the project, as they are related to its specifications. For instance, in project phases, we can determine if price of materials of a building is less or more relevant than the estimated energy spend in HVAC. It is not in the scope of this internship to directly determine those attributes, since the decision on those was defined with the aid of all partners because of the different expertises.

However, for item number (3), there are several methods, as presented in Figure 7:

The **Weighted Sum Model** (WSM) is probably the most commonly used approach, especially in single dimensional problems. If there are m alternatives and n criteria then, the best alternative is the one that satisfies (in the maximization case) the following expression:

$$A_{WSM}^* = \max_i \sum_{j=1}^n a_{ij} w_j, \text{ for } i = 1, 2, \dots, m \quad (3.2)$$

Such that A_{WSM}^* is the WSM score of the best alternative, n is the number of decision criteria, a_{ij} is the actual value of the i^{th} alternative in terms of the j^{th} criterion, and w_j is the weight of importance of the j^{th} criterion. In single-dimensional cases, where all the units are the same (e.g., dollars, seconds), the WSM can be used without difficulty. Difficulty with this method emerges when it is applied to multi- dimensional MCDM problems. Then, in combining different dimensions, and consequently different units, the additive utility assumption is violated and the result is equivalent to "adding apples and oranges" [42].

For a simple example, let us suppose that an MCDM problem involves four criteria, expressed on the exact same unit and three alternatives. Table 2 presents its decision matrix.

Table 2 – Decision matrix numerical example.

Alts.	Criteria			
	C_1 (0.20)	C_2 0.15	C_3 0.40	C_4 0.25)
A_1	25	20	15	30
A_2	10	30	20	30
A_3	30	10	30	10

Using equation (3.2), we have:

$$A_{1,WSP} = 25 \times 0.20 + 20 \times 0.15 + 15 \times 0.40 + 30 \times 0.25 = 21.50.$$

Similarly, for the other alternatives, we get:

$$A_{2,WSP} = 22.00$$

and

$$A_{3,WSP} = 20.00.$$

Then, the ranking $A_2 > A_1 > A_3$ is derived for this case.

The **Weighted Product Model** (WPM) is similar to the WSM, with the difference that addition there is multiplication in the method. Each alternative is compared with the others by multiplying a number of ratios, one for each criterion. Each ratio is raised to the power equivalent to the relative weight of the corresponding criterion. In general, in order to compare two alternatives A_K and A_L , the following product has to be calculated:

$$R\left(\frac{A_K}{A_L}\right) = \prod_{j=1}^n \left(\frac{a_{Kj}}{a_{Lj}}\right)^{w_j} \quad (3.3)$$

such that n is the number of criteria, a_{ij} is the actual value of the i^{th} alternative in terms of the j^{th} criterion, and w_j is the weight of importance of the j^{th} criterion.

If the term $R\left(\frac{A_K}{A_L}\right)$ is greater than or equal to one, then it indicates that alternative A_K is more desirable than alternative A_L (in the maximization case). The best alternative is the one that is better than or at least equal to all other alternatives.

The WPM is sometimes called dimensionless analysis because its structure eliminates any units of measure. Thus, the WPM can be used in single and multi-dimensional MCDM. An advantage of the method is that instead of the actual values it can use relative ones [42].

For an example, let us consider the same decision matrix presented on Table 2. Applying equation (3.3), we have:

$$R\left(\frac{A_1}{A_2}\right) = \left(\frac{25}{10}\right)^{0.20} \times \left(\frac{20}{30}\right)^{0.15} \times \left(\frac{15}{20}\right)^{0.40} \times \left(\frac{30}{30}\right)^{0.25} = 1.0074 > 1$$

Similarly, we get

$$R\left(\frac{A_1}{A_3}\right) = 1.0671 > 1$$

and

$$R\left(\frac{A_2}{A_3}\right) = 1.0592 > 1.$$

Therefore, the dimensionless raking for the alternatives is $A_2 > A_3 > A_1$.

The **Analytic Hierarchy Process** (AHP) decomposes a complex MCDM problem into a system of hierarchies. The final step in the AHP deals with the structure of an $m \times n$ matrix (where m is the number of alternatives and n is the number of criteria). The matrix is constructed by using the relative importance of the alternatives in terms of each criterion. The vector $(a_{i1}, a_{i2}, a_{i3}, \dots, a_{in})$ for each i is the principal eigenvector of an $n \times n$ reciprocal matrix which is determined by pairwise comparisons of the impact of the m alternatives on the i^{th} criterion.

The **ELECTRE** (for Elimination and Choice Translating Reality, English translation from the French original) has as its basic concept to deal with "outranking relations" by using pairwise comparisons among alternatives under each one of the criteria separately. The outranking relationship of the two alternatives A_i and A_j , denoted as $A_i \rightarrow A_j$ describes that even when the i^{th} alternative does not dominate the j^{th} alternative quantitatively, then the decision maker may still take the risk of regarding A_i as almost surely better than A_j . Alternatives are said to be dominated, if there is another alternative which excels them in one or more criteria and equals in the remaining criteria.

The ELECTRE method begins with pairwise comparisons of alternatives under each criterion. Using physical or monetary values, denoted as $g_i(A_j)$ and $g_i(A_k)$ of the alternatives A_j and A_k respectively, and by introducing threshold levels for the difference $g_i(A_j) - g_i(A_k)$, the decision maker may declare that he is indifferent between the alternatives under consideration, that he has a weak or a strict preference for one of the two, or that he is unable to express any of these preference relations. Therefore, a set of binary relations of alternatives, the so-called outranking relations, may be complete or incomplete. Next, the decision maker is requested to assign weights or importance factors to the criteria in order to express their relative importance.

Through the consecutive assessments of the outranking relations of the alternatives, the ELECTRE method elicits the so-called concordance index, defined as the amount of evidence to support the conclusion that alternative A_j outranks, or dominates, alternative A_k , as well as the discordance index, the counter-part of the concordance index.

Finally, the ELECTRE method yields a system of binary outranking relations between the alternatives. Because this system is not necessarily complete, the ELECTRE method is

sometimes unable to identify the most preferred alternative. It only produces a core of leading alternatives. This method has a clearer view of alternatives by eliminating less favorable ones. This method is especially convenient when there are decision problems that involve a few criteria with a large number of alternatives [42].

TOPSIS (for the Technique for Order Preference by Similarity to Ideal Solution) was developed as an alternative to the ELECTRE method and can be considered as one of its most widely accepted variants. The basic concept of this method is that the selected alternative should have the shortest distance from the ideal solution and the farthest distance from the negative-ideal solution in some geometrical sense.

The TOPSIS method assumes that each criterion has a tendency of monotonically increasing or decreasing utility. Therefore, it is easy to define the ideal and negative-ideal solutions. The Euclidean distance approach was proposed to evaluate the relative closeness of the alternatives to the ideal solution. Thus, the preference order of the alternatives can be derived by series of comparisons of these relative distances [42].

3.3.2 Sensitivity Analysis

The sensitivity analysis can be branched into **input-output sensitivity analysis**, where we analyse how a variation on the input will impact the output, and, **alternative-criteria sensitivity analysis**, which corresponds to analyse the percentage slack of variation that the values of the alternatives or criteria can have and still keep the same ranking.

There is a connection between these two types of analysis, since, when determined the criticality of the alternative, we can observe how much we can vary the input in order to keep the nominal or chosen alternative as top.

3.3.2.1 Alternative-criteria sensitivity analysis

In the alternative-criteria sensitivity analysis the decision maker can make better decisions if he can determine how critical each criterion is. In other words, how sensitive the actual ranking of the alternatives is to changes in the current weights of the decision criteria. We can perform a sensitivity analysis on the weights of the criteria. This sensitivity analysis approach determines what is the smallest change in the current weights of the criteria, which can alter the existing ranking of the alternatives. On the other hand, it is possible to determine how critical the various performance measures of the alternatives (in terms of a single decision criterion at a time) are in the ranking of the alternatives [42].

We can define the most critical criterion in two alternative ways. In the first way the interest is on whether the indication of the best (top) alternative changes or not. On the second definition the interest is on changes in the ranking of any alternative.

Also, the term *smallest change* can also be defined in two different ways. The first way is to define smallest change in absolute terms; the second way is to define smallest change in relative terms. Usually it is more meaningful to use relative changes, since the magnitude by itself does not give a reference for comparison.

To determine the most critical criterion, meaning, how critical is the change on the weights of each criterion, we can base ourselves on the following definitions [42]:

Definition 6. *Critical change*

Let δ_{kij} (for $1 \leq i < j \leq m$ and $1 \leq k \leq n$) denote the minimum change in the current weight w_k of criterion C_k such that the ranking of alternatives A_i and A_j will be reversed.

We can then define δ'_{kij} as follows:

$$\delta'_{kij} = \delta_{kij} \left(\frac{100}{w_j} \right), \text{ for any } 1 \leq i < j \leq m \text{ and } 1 \leq k \leq n. \quad (3.4)$$

such that the parameter δ'_{kij} expresses changes in relative terms.

Definition 7. *The Percent-Top (PT) Critical criterion*

The Percent-Top critical criterion is the criterion which corresponds to the smallest δ'_{kij} (for $1 \leq j \leq m$ and $1 \leq k \leq n$) value.

Definition 8. *The Percent-Any (PA) Critical criterion*

The Percent-Any critical criterion is the criterion which corresponds to the smallest δ'_{kij} (for $1 \leq i < j \leq m$ and $1 \leq k \leq n$) value.

Definition 9. *Criticality degree of criterion*

The criticality degree of criterion C_k denoted as D'_k , is the smallest percent amount by which the current value of w_k must change, such that the existing ranking of the alternatives will change. That is, the following relation is true:

$$D'_k = \min_{1 \leq i < j \leq m} \{ |\delta'_{kij}| \}, \forall n \geq k \geq 1. \quad (3.5)$$

Definition 10. *Sensitivity coefficient of criterion*

The sensitivity coefficient of criterion C_k denoted as $\text{sens}(C_k)$, is the reciprocal of its criticality degree. That is, the following relation is true:

$$\text{sens}(C_k) = \frac{1}{D'_k}, \forall n \geq k \geq 1. \quad (3.6)$$

If the criticality degree is unfeasible (i.e., impossible to change any alternative rank with any weight change), then the sensitivity coefficient is set to be equal to zero.

If we want to determine the most critical a_{ij} measure of performance when the WSM, AHP, or the WPM method is used, the following definitions are pertinent to this problem [42].

Definition 11. *Threshold value of a_{ij}*

Let τ_{ijk} (for $1 \leq i < k \leq m$ and $1 \leq j \leq n$) denote the threshold value of a_{ij} , which is the minimum change which has to occur in the current value of a_{ij} such that the current ranking between alternatives A_i and A_k will change.

Since there are m alternatives, each a_{ij} performance measure is associated with a total of $(m - 1)$ such threshold values. In a similar way as earlier regarding the definition of the δ'_{ijk} values, one can also consider threshold values expressed in relative terms. We denote these relative term threshold values as τ'_{ijk} , that is:

$$\tau'_{ijk} = \tau_{ijk} \left(\frac{100}{a_{ij}} \right), \text{ for any } 1 \leq i, k \leq m, \text{ and } 1 \leq j \leq n. \quad (3.7)$$

When we consider threshold values we will mean the ones defined in relative terms (i.e., the τ'_{ijk} values). Given the previous notion of threshold value, we define as the most sensitive alternative the one which is associated with the smallest threshold value. Also as before, one may be interested in changes in the ranking of (only) the best alternative, or in changes in the ranking of any alternative.

Definition 12. *Criticality degree of alternative A_i*

The criticality degree of alternative A_i in terms of criterion C_j , denoted as Δ'_{ij} is the smallest amount (%) by which the current value of a_{ij} must change, such that the existing ranking of alternative A_i will change. That is, the following relation is true:

$$\Delta'_{ij} = \min_{k \neq i} \left\{ \tau'_{kij} \right\}, \forall m \geq i \geq 1, \text{ and } n \geq j \geq 1. \quad (3.8)$$

Definition 13. *Most critical alternative*

Alternative A_L is the most critical alternative if it is associated with the smallest criticality degree. That is, if and only if the following relation is true:

$$\Delta'_{Lj} = \min_{m \geq i \geq 1} \left\{ \min_{n \geq j \geq 1} \left\{ \Delta'_{ij} \right\} \right\}, \text{ for some } n \geq k \geq 1. \quad (3.9)$$

Definition 14. *Sensitivity coefficient of alternative A_i*

The sensitivity coefficient of alternative A_i in terms of criterion C_j , denoted as $\text{sens}(a_{ij})$ is the reciprocal of its criticality degree. That is, the following condition is true:

$$\text{sens}(a_{ij}) = \frac{1}{\Delta'_{ij}}, \text{ for any } m \geq i \geq 1, \text{ and } n \geq j \geq 1. \quad (3.10)$$

If the criticality degree is unfeasible, then the sensitivity coefficient is set to be equal to zero.

From Definition 12 it follows that the smaller the criticality degree Δ'_{ij} is the easier is for the ranking of alternative A_i to change. Alternatively, Definition 14 indicates that ranking changes are easier, as the sensitivity coefficients $sens(a_{ij})$ are higher. Finally, when Definition 13 is combined with Definitions 12 and 14, it indicates that the most sensitive alternative is the one with the highest sensitivity coefficient.

The above definitions for alternative and criterion can be used in order to determine how critical a determined solution is. However, because each method seen in 3.3.1 have its particularities, each method requires a different analysis, which is beyond the scope of this internship. Though, it is important to take familiarity and be aware that, along with a robust analysis, we can determine if a given solution is to be considered as final.

3.3.2.2 Input-output sensitivity analysis

The input-output sensitivity analysis is the study of how the uncertainty in the output of a mathematical model or system can be apportioned to different sources of uncertainty in its inputs, in other words, it shows us how a variation δ on the input will influence or change each of the outputs of the system. It helps to determine if the model is faithful to the process it models, which are the variables that contribute most and the least to the variability of the model response, which variables, or which groups of variables, interact with which other.

It is possible to group sensitivity analysis methods into three classes: screening methods, local sensitivity analysis and global sensitivity analysis [43].

Screening methods qualitatively analyse the importance of input variables on the variability of the model response. They make it possible to establish a hierarchy within the input variables as a function of their influence on the variability of the response.

The local sensitivity analysis, like the global analysis, are quantitative analysis methods, which allow in addition to establish a hierarchy within the input variables, to give an order of magnitude of the differences within this hierarchy.

The **local sensitivity analysis** studies how small perturbations around a value $x^0 = (x_1^0, \dots, x_p^0)$ of inputs affect the value of the output. The most traditional local analysis method is the One factor at time (OAT) approach, which consists in calculating or estimating the sensitivity indices defined by:

$$S_i = \frac{\delta y}{\delta x_i}(x_1^0, \dots, x_p^0) \quad (3.11)$$

expressing the effect on the value of the random variable Y of disturbing the values of the variables X_i around a nominal value x_i^0 .

The **global sensitivity analysis** is concerned with the variability of the output of the model in its domain of variation. It studies how the variability of the inputs is reflected on that of the output, by determining how much of the variance of the output is due to such inputs or set of inputs.

It is possible to distinguish local analysis from global analysis in this way: local analysis focuses on the value of the response, while global analysis looks at its variability.

For white-box models, we can use derivative based (local sensitivity analysis) tools to determine the sensitivity of the system. However, this analysis is a lot more limited when dealing with black-box systems. For such, we need to use what is called the global sensitivity analysis, which is often placed using Monte Carlo techniques.

To consider the sensitivity indices without hypothesis on the model, let us consider the following [43]:

$$Y = f(X_1, \dots, X_p) \quad (3.12)$$

such that X_i represents the i^{th} component of the random vector \mathbf{X} , with independent input variables.

To comprehend the importance of an input variable X_i on the variance of the output Y , we study how much the variance of Y decreases if we fix the variable X_i to a value x_i^* :

$$Var(Y | X_i = x_i^*) \quad (3.13)$$

such that $Var(X)$ corresponds to the variance of X

This quantity is the conditional variance of Y knowing $X_i = x_i^*$. The problem of this indicator is the choice of the value x_i^* of X_i , which we solve by considering the expectation of this quantity for all the possible values of x_i^* , noted:

$$E[Var(Y | X_i)] \quad (3.14)$$

such that $E(X)$ corresponds to the expected value of X .

Thus, the greater the variable X_i will be to the variance of Y , the smaller this quantity will be.

Given the Law of Total Variance, an indicator of the sensitivity of Y to X_i will be the variance of the expectation of Y conditionally at X_i :

$$Var(E[Y | X_i]) \quad (3.15)$$

The larger the variable X_i , the greater the quantity. In order to use a standardized indicator, we can define the sensitivity index [43].

Definition 15. *The sensitivity index:*

The sensitivity index expressing the sensitivity of Y to X_i is defined by:

$$S_i = \frac{Var(E[Y | X_i])}{V(Y)} \quad (3.16)$$

This index quantifies the sensitivity of the output Y to the input variable X_i , or the variance part of Y due to the variable i . It is called the first-order sensitivity index by Sobol.

Sobol [44] introduces this sensitivity index by decomposing the function f of the model in sum of functions of increasing dimensions. It is one of the most known techniques, and the one implemented on CADES. It corresponds to a Monte Carlo (or quasi-Monte Carlo) global sensitivity analysis based on the analysis of the variance.

Let us assume a model described by a function $u = f(\mathbf{x})$, such that the input $\mathbf{x} = (x_1, \dots, x_n)$ is a points inside an n -dimensional box and u is a scalar output.

Let $u^* = f(\mathbf{x}^*)$ be the required solution. The local sensitivity analysis is estimated by the partial derivative $\left(\frac{\delta u}{\delta x_k}\right)_{x=x^*}$, for x_k a local point in the domain.

The global sensitivity approach however, does not specify the input $\mathbf{x} = \mathbf{x}^*$, it considers the model $f(x)$ inside the box. Therefore, global sensitivity indices should be regarded as a tool for studying the mathematical model rather than its specified solution.

Definition 16. *Expansion into summands of different dimensions*

Considering a group of indices i_1, \dots, i_s , where $1 \leq i_1 \leq \dots \leq i_s \leq n$ and $s = 1, \dots, n$. Also considering an integrable function $f(\mathbf{x})$ defined in I^n , such that I corresponds to the unit interval $[0, 1]$. We can represent a function $f(x_1, \dots, x_n)$ by:

$$f(\mathbf{x}) = f_0 + \sum_{s=1}^n \sum_{i_1 < \dots < i_s} f_{i_1 \dots i_s}(x_{i_1}, \dots, x_{i_s}) \quad (3.17)$$

Such that f_0 is constant and represents the mean value of the model ($E(Y) = \int_{H^n} f(\mathbf{x}) d\mathbf{x}$).

Equation (3.17) means that:

$$f(\mathbf{x}) = f_0 + \sum_i f_i(x_i) + \sum_{i < j} f_{ij}(x_i, x_j) + \dots + f_{12 \dots n}(x_1, x_2, \dots, x_n) \quad (3.18)$$

With a total number of summands of 2^n .

Definition 17. *The ANOVA (Analysis of Variance) representation of $f(\mathbf{x})$*

Equation 3.17 is the ANOVA-representation of $f(\mathbf{x})$ if

$$\int_0^1 f_{i_1, \dots, i_s}(x_{i_1}, \dots, x_{i_s}) dx_k = 0, \text{ for } k = i_1, \dots, i_s \quad (3.19)$$

Definition 17 states that in order to have an ANOVA-representation of $f(x)$, the integrals of the summands f_{i_1, \dots, i_s} with respect to any of their own variables are zero. This is a condition because the term itself can not contribute to its own variance. It also means that all inputs are orthogonal to each other (independent).

Besides, it comes from (3.19) that the members in (3.17) are orthogonal and can be expressed as integrals of $f(x)$.

Let us assume that $f(\mathbf{x})$ is square integrable. Then, squaring (3.17) and integrating over I^n , we get:

$$\int f^2(\mathbf{x})dx - f_0^2 = \sum_{s=1}^n \sum_{i_1 < \dots < i_s} \int f_{i_1 \dots i_s}^2 dx_{i_1}, \dots, dx_{i_s} \quad (3.20)$$

If x is a random point uniformly distributed in I^n , then $f(\mathbf{x})$ and $f_{i_1, \dots, i_s}(x_{i_1, \dots, i_s})$ are random variables with the corresponding variances V and V_{i_1, \dots, i_s} as follows:

$$V = \int f^2(x)dx - f_0^2 = \sum_{s=1}^n \sum_{i_1 < \dots < i_s} V_{i_1, \dots, i_s} \quad (3.21)$$

and

$$V_{i_1, \dots, i_s} = \int f_{i_1 \dots i_s}^2 dx_{i_1}, \dots, dx_{i_s} \quad (3.22)$$

Definition 18. *Global sensitivity indices:*

$$S_{i_1, \dots, i_s} = \frac{V_{i_1, \dots, i_s}}{V} \quad (3.23)$$

All the S_{i_1, \dots, i_s} are non-negative and their sum is given by:

$$\sum_{s=1}^n \sum_{i_1 < \dots < i_s} S_{i_1, \dots, i_s} = 1 \quad (3.24)$$

For a piecewise continuous function $f(\mathbf{x})$, the equality $S_{i_1, \dots, i_s} = 0$ represents that $f_{i_1, \dots, i_s}(x_{i_1, \dots, i_s}) \equiv 0$. Thus, the functional structure of $f(\mathbf{x})$ can be investigated by estimating numbers S_{i_1, \dots, i_s} [44].

These indices can be obtained thanks to empirical estimators from N-type Monte Carlo samples:

$$\begin{cases} \hat{f}_0 = \frac{1}{N} \sum_{j=1}^N f(X_m) \\ \hat{V} = \frac{1}{N} \sum_{j=1}^N f^2(X_m) - \hat{f}_0^2 \end{cases} \quad (3.25)$$

Where X_m is a vector of points of the hypercube H^n drawn randomly. For a first order index we have:

$$S_i = \frac{V_i}{V} \quad (3.26)$$

with

$$V_i = V(E(Y | X_i)) = \underbrace{E([E(Y | X_i)]^2)}_{U_i} - [E(E(Y | X_i))]^2 = U_i - [E(Y)]^2 \quad (3.27)$$

Sobol (apud [43]) proposes to estimate the quantity U_i , the expected value of the square of the conditional expected value on X_i , as a classical expectation but taking into

account the conditioning at X_i , by varying in all the calls of the function all the variables except the variable X_i . This then requires two samples of size N , $(X_m^{(1)})$ and $(X_m^{(2)})$.

$$\hat{U}_i = \frac{1}{N} \sum_{m=1}^N f\left(X_{m,1}^{(1)}, \dots, X_{m,(i-1)}^{(1)}, X_{m,i}^{(1)}, X_{m,(i+1)}^{(1)}, \dots, X_{m,s}^{(1)}\right) \times f\left(X_{m,1}^{(2)}, \dots, X_{m,(i-1)}^{(2)}, X_{m,i}^{(1)}, X_{m,(i+1)}^{(2)}, \dots, X_{m,s}^{(2)}\right) \quad (3.28)$$

Which results in the following estimator:

$$\begin{cases} \hat{V}_i = \hat{U}_i - \hat{f}_0^2 = \frac{1}{N} \sum_{m=1}^N f\left(\mathbf{X}_{m,\sim 1}^{(1)}, \mathbf{X}_{m,i}^{(1)}\right) f\left(\mathbf{X}_{m,\sim 1}^{(2)}, \mathbf{X}_{m,i}^{(1)}\right) \\ \hat{S}_i = \frac{\hat{V}_i}{\hat{V}} = \frac{\hat{U}_i - \hat{f}_0^2}{\hat{V}} \end{cases} \quad (3.29)$$

Such that $\mathbf{X}_{\sim i}$ corresponds to the vector \mathbf{X} deprived of its i^{th} component.

Likewise, we can estimate the second order indices [43]:

$$S_{ij} = \frac{V_{ij}}{V} \quad (3.30)$$

such that:

$$V_{ij} = V(E[Y | X_i, X_j]) - V_i - V_j = U_{ij} - E[Y]^2 - V_i - V_j \quad (3.31)$$

we can estimate the quantity $U_{ij} = E[E[Y | X_i, X_j]^2]$ the same way, by varying all variables but X_i and X_j :

$$\hat{U}_i = \frac{1}{N} \sum_{m=1}^N f\left(X_{m,1}^{(1)}, \dots, X_{m,(i-1)}^{(1)}, X_{m,i}^{(1)}, X_{m,(i+1)}^{(1)}, \dots, X_{m,(j-1)}^{(1)}, X_{m,j}^{(1)}, X_{m,(j+1)}^{(1)}, \dots, X_{m,s}^{(1)}\right) \times f\left(X_{m,1}^{(2)}, \dots, X_{m,(i-1)}^{(2)}, X_{m,i}^{(1)}, X_{m,(i+1)}^{(2)}, \dots, X_{m,(j-1)}^{(2)}, X_{m,j}^{(1)}, X_{m,(j+1)}^{(2)}, \dots, X_{m,s}^{(2)}\right) \quad (3.32)$$

And the index S_{ij} can be estimated by:

$$\hat{S}_{ij} = \frac{\hat{U}_{ij} - \hat{f}_0^2 - \hat{V}_i - \hat{V}_j}{\hat{V}} \quad (3.33)$$

And so on for sensitivity indices of higher order.

On the other hand, total sensitivity indices can be estimated directly:

$$S_{T_i} = 1 - \frac{V(E[Y | \mathbf{X}_{\sim i}])}{V(Y)} = 1 - \frac{\mathbf{V}_{\sim i}}{V} \quad (3.34)$$

where $\mathbf{V}_{\sim i}$ is the variance of the expectation of Y conditionally to all variables except X_i . $\mathbf{V}_{\sim i}$ is then estimated as V_i , except that instead of varying all the variables except X_i , we only vary X_i [43].

So, to estimate $\mathbf{V}_{\sim i} = E[E[Y | \mathbf{X}_{\sim i}]^2] - E[E[Y | \mathbf{X}_{\sim i}]]^2 = U_{\sim i} - E[Y]^2$, we estimate $\mathbf{U}_{\sim i}$ for:

$$\hat{U}_i = \frac{1}{N} \sum_{m=1}^N f\left(X_{m,1}^{(1)}, \dots, X_{m,(i-1)}^{(1)}, X_{m,i}^{(1)}, X_{m,(i+1)}^{(1)}, \dots, X_{m,(j-1)}^{(1)}, X_{m,j}^{(1)}, X_{m,(j+1)}^{(1)}, \dots, X_{m,s}^{(1)}\right) \times f\left(X_{m,1}^{(1)}, \dots, X_{m,(i-1)}^{(1)}, X_{m,i}^{(2)}, X_{m,(i+1)}^{(1)}, \dots, X_{m,(j-1)}^{(1)}, X_{m,j}^{(1)}, X_{m,(j+1)}^{(2)}, \dots, X_{m,s}^{(2)}\right) \quad (3.35)$$

and finally,

$$\hat{S}_{T_i} = 1 - \frac{\hat{\mathbf{U}}_{\sim i} - \hat{f}_0^2}{\hat{V}} \quad (3.36)$$

As a remark, using a Monte Carlo sample size of N , the actual number of simulations of the input variables needed to estimate the sensitivity indices is $2N$, since this estimate requires two sets of simulations. The number of calls to the function of the model is then $N(m + 1)$, where m is the number of estimated indices. For a model with s input variables, the estimation of all sensitivity indices requires $N(2s)$ calls to the function. On the other hand, only the first-order indices and the total indices require only $N(2s + 1)$ calls.

In short, the analysis is made by a column-wise search to find the contribution of each input, and by removing the contribution done by the undesired inputs found from the inter-relation when satisfying Definitions 16 to 18.

One drawback of this model is that it is slow when used with complex black-box models. A more extensive discussion will be made when presenting the results for the case Begin on Chapter 6.

3.4 Meta-modeling

Meta-model (also known as surrogate model or response surface approximation) can be defined as a mathematical relation or algorithm representing input and output relations. It is referred to as a technique that makes use of the sampled data (observed by running the computer code) to build surrogate models, which are sufficient to predict the output of an expensive computer code at untried points in the design space.

As the model itself, meta-model is yet another abstraction, highlighting properties of the model itself but in a simplified or, in the case of black-box models, a reconstruction of the desired characteristics of the model. Meta-modeling typically involves studying the output and input relationships and then fitting right meta-models to represent that behaviour. For that, it involves (a) choosing an experimental design for generating data, (b) choosing a model to represent the data, and then (c) fitting the model to the observed data [45]. Thus, how to choose sample points, how to build surrogate models, and how to evaluate the accuracy of surrogate models are key issues for surrogate modeling.

For optimization problems, surrogate models can be regarded as approximation models for the cost function(s) and state function(s), which are built from sampled data obtained by randomly probing the design space, as the technique evolves from classical Design of Experiments (DoE) theory, in which polynomial functions are used as response surfaces, or metamodels [46]. Once the surrogate models are built, an optimization algorithm such as Genetic Algorithms (GA) can be used to search the new design (based on the surrogate models) that is most likely to be the optimum. Since the prediction with a surrogate model is

generally much more efficient than that with a numerical analysis code, the computational cost associated with the search based on the surrogate models is generally negligible [47].

Han [47] presents several frameworks to use surrogate models in optimization. However, as this internship is part of a larger project, the framework that was already defined to be used is afterwards depicted and explained in Figure 11. Section 3.5 presents the options made for each step of the optimization algorithm. Hence, the next sections and subsections will explain the basis of the tools used for the latter. As an outlook, the Latin Hypercube Sampling is used for the experimental design, and response surface methodology and Kriging are used for model choice and model fitting, however, in different steps of the final algorithm.

3.4.1 Design of Experiments (DoE)

The Design of Experiments is an effective tool for maximizing the amount of information gained from a study while minimizing the amount of data to be collected [48]. It consists of a statistical technique for quickly optimizing performance of a system with known input variables. It investigates the cause and effect of many different factors in a single study, instead of running many separate studies (varying one input at a time).

In experiments design, the inputs are called factors, the outputs are called responses, and the values of X are called levels. The polynomial equation that is frequently used to model the response variable (Y) as a function of the input factors (X) is [48]:

$$Y = \beta_0 + \sum_{i=1}^p \beta_i X_i + \sum_{i=1}^p \sum_{j=1}^p \beta_{ij} X_i X_j + \sum_{i=1}^p \sum_{j=1}^p \sum_{k=1}^p \beta_{ijk} X_i X_j X_k + \dots, \quad i \neq j \neq k \quad (3.37)$$

where β_0 represents the overall mean response, β_i represents the main effects for each factor ($i = 1, 2, \dots, p$), β_{ij} the two-way interaction between the i^{th} and j^{th} factors, β_{ijk} the three-way interaction between the i^{th} , j^{th} and k^{th} factors, and so on.

The final complexity of equation (3.37) is related to the desired resolution of the regression analysis. To explain that, we need to understand **Confounding**. The latter occurs when only the summation of several effects can be estimated, not the effect separately. When there are confounded results, it is required to run additional experiments to untangle the results, and thus, identify which factor combinations are active and which are not. The effects in the regression analysis equation are not confounded if the sum of their ways is less than the resolution level of the design. That means that, in order to have all of the two-way interactions unconfounded from each other, the resolution level needs to be at least V (equation (3.37) must have at least 6 terms, including β_0). However, it is usual to assume that most of the three-way and higher order interactions are negligible. The three-way and higher order interactions are not estimated separately, but their combination are used to estimate the precision of the estimates and to compute the confidence interval on the effect [48].

We can then say that an experimental design represents a sequence of experiments to be performed, expressed in terms of factors (design variables) set at specified levels (predefined values). An experimental design is represented by a matrix \mathbf{X} , where the rows denote experiment runs, and the columns denote particular factor settings, as represented on table 3.

Table 3 – Notation experimental design matrix \mathbf{X} .

Run	Factor 1	Factor 2	Factor 3	...	Factor m
1	-1	-1	-1	...	-1
2	+1	-1	-1	...	+1
3	-1	+1	-1	...	+1
\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
n	+1	+1	+1	...	+1

The most basic experimental design is a full factorial design. The number of design points dictated by a full factorial design is the product of the number of levels for each factor. The most common are 2^k (for evaluating main effects and interactions) and 3^k designs (for evaluating main and quadratic effects and interactions) for k factors at 2 and 3 levels, respectively. By 2^k , we mean that we apply for each factor two levels, +1 and -1, whereas, by 3^k we apply for each factor three levels, +1, 0 and -1. Let us consider a system with 3 inputs, (x_1, x_2, x_3) . For this case, a 2^3 full factorial design is shown in Figure 8(a).

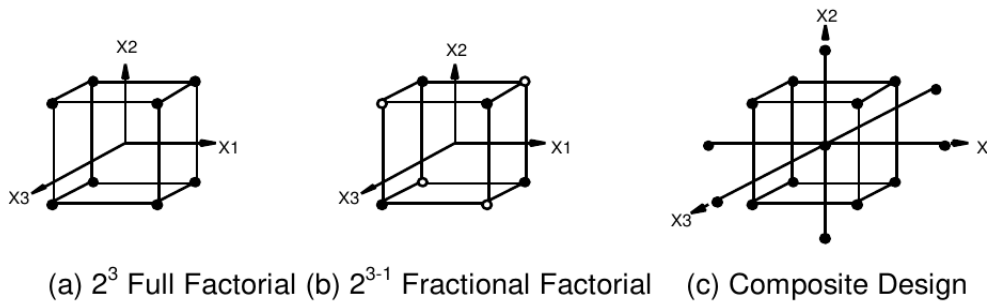


Figure 8 – Basic Three-Factor Design.

The size of a full factorial experiment increases exponentially with the number of factors; this leads to an unmanageable number of experiments. **Fractional factorial** designs are used when experiments are costly and many factors are required. A fractional factorial design is a fraction of a full factorial design; the most common are $2^{(k-p)}$ designs in which the fraction is $\frac{1}{2}(p)$. A half fraction of the 2^3 full factorial design is shown in Figure 8(b). The reduction of the number of design points in a fractional factorial design is not without a price. The 2^3 full factorial design shown in Figure 8(a) allows estimation of all main effects (x_1, x_2, x_3) , all two factor interactions $(x_{1x2}, x_{1x3}, x_{2x3})$, as well as the three factor interaction (x_{1x2x3}) . For the 2^{3-1} fractional factorial indicated by the solid dots in Figure 8(b), the main effects are aliased (or biased) with the two factor interactions. Aliased effects cannot be estimated independently unless they are known (or assumed) not to exist [45].

The parameters of the polynomials in equation (3.37) are usually determined by least squares regression analysis by fitting the response surface approximations to existing data. These approximations are normally used for prediction within response surface methodology (RSM 3.4.2) [45].

3.4.1.1 Latin Hypercube Sampling

The Latin Hypercube Sampling (LHS) was the choice on an experimental design for generating input data for the black-box model. As a contextualization, three main approaches to inserting the ‘uncertainties of system and environment’ in building simulation models are in use: stochastic differential equations, first/second order reliability methods, and Monte-Carlo sampling-based techniques. While the first two decidedly have potential, Monte-Carlo currently prevails due to the complex, non-linear and transient character of most building simulation analyses. The LHS is a well known variance-reduction technique in Monte-Carlo method, whose superiority over random sampling has been corroborated often already [49].

Let us consider an experimental design with p points in d dimensions is written as a $p \times d$ matrix in the form $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_p]^T$, where each column represents a variable, and each row $\mathbf{x}_i = [x_i^1 \ x_i^2 \ \dots \ x_i^d]$ represents a sample. A Latin hypercube design is constructed in such a way that each of the d dimensions is divided into p equal levels (sometimes called bins) and that there is only one point (or sample) at each level. As originally proposed, a random procedure is used to determine the point locations. Figure 9 [50] shows three examples of Latin hypercube designs with $d = 2$ and $p = 20$. The extreme case illustrated in Figure 9(a) is a Latin hypercube with very poor space filling qualities. Randomization alone could improve the experimental design to the point exemplified by Figure 9(b). On the other hand, optimization of point placement would lead to the better choice shown in Figure 9(c), where points are more uniformly distributed over the domain.

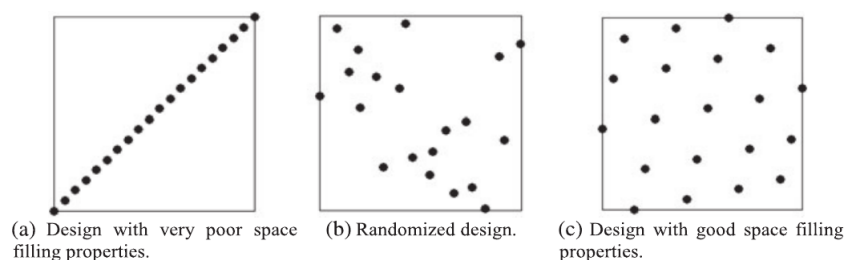


Figure 9 – Examples of Latin hypercube designs with $d = 2$ dimensions and $p = 20$ points.

The desired sampling is the one that presents the best point distribution and the smallest covariance between the points, meaning that when performing the DoE, it will give us the best meta-model, since different points will not give us equivalent results.

In addition to the methodological improvements such as orthogonal LHS and symmetric LHS that attempt to systematically generate Latin hypercube samples of better quality,

there have been a large number of studies that utilize the optimization theory to improve the performance of LHS. The approach is to define secondary criteria (objective functions), in addition to being Latin hypercube, and formulate and solve an optimization problem to achieve (near) optimal Latin hypercube samples. This can be very effective, as in any variation of LHS, there may exist a huge number of configurations (sample points arrangements) that satisfy the associated LHS criteria, but do poorly in terms of other criteria (e.g., space filling). The LHS algorithms typically randomly pick one of the many possible configurations, while optimization helps navigate through the myriad of choices and identify one that is (near) optimal in terms of secondary criteria [51].

According to Viana [50], for computer experiments, an attractive sampling technique would have to be flexible enough to (i) provide data for modeling techniques based on very different statistical assumptions and (ii) be capable of covering small to large design spaces (no constraints in terms of data density and location). The fact that Latin hypercube designs can be optimized without having to look at the statistical assumptions of the model, and have as many points as one can afford, makes LHS a strategy very attractive from a practical point of view. Another good reason for using the Latin hypercube is its flexibility. For example, if few dimensions have to be dropped out, the resulting design is still a Latin hypercube design (maybe sub-optimal, but a Latin hypercube nevertheless). That happens because Latin hypercube samples are non-collapsing.

Figure 10 illustrates the case of a Latin hypercube design with $d = 3$ dimensions and $p = 15$ points [50]. Any of the two-dimensional projections is still a Latin hypercube design with the same $p = 15$ points (although, for this particular case, the $(x_1 - x_2)$ projection is the best in terms of space filling). Thus, if one cannot afford another set of data properly designed for the smaller domain, the existing data can be reused without reduction in number of sampled points.

Although by virtue of their definition, Latin hypercube designs have good uniformity with respect to each dimension individually and desirable properties, such as space filling, column-wise orthogonality, or conversely, designs with statistically dependent variables come at the cost of very expensive optimization, these properties bear the draw back of not being invariant under transformations. This is particularly important for applications where sampling is performed in one coordinate system, but modeling is performed in another.

Another disadvantage is that optimization problem for improving LHS belongs to the class of combinatorial optimization, with a total search space of $(n!)^p$ configurations for an exhaustive search [52]. The computational efficiency of an optimization-assisted LHS algorithm depends on the size of the search space and the efficiency of the optimization algorithm used. In practice, therefore, the sampling procedure can become computationally demanding for larger values of n and p . A variety of optimization algorithms have been used in the literature to solve such combinatorial problems, including simulated annealing,

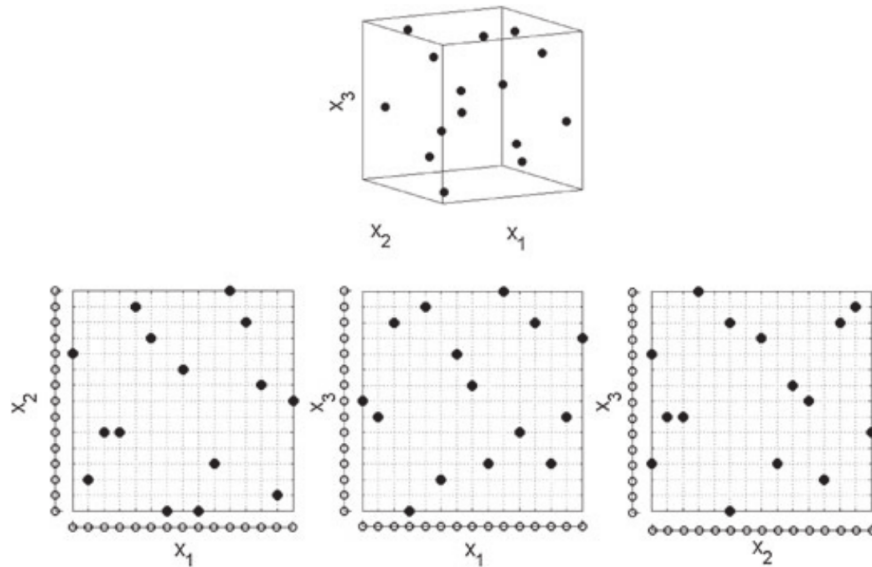


Figure 10 – Latin Hypercube Design.

genetic algorithms, and the branch-and-bound-algorithm. A table with those algorithms and its references is presented in [51].

3.4.2 Response Surface Methodology

Although there can be several definitions for Response Surface Methodology (RSM) [45], here we consider it as a sequential procedure that combines techniques of statistical design and numerical optimization in order to optimize processes and product designs [53]. These statistical approximations, or metamodels, are used to replace the actual expensive computer analyses, facilitating multidisciplinary, multi-objective optimization and concept exploration [45]. The basic assumption in RSM is that a response Y is a function of a set of design variables X_1, X_2, \dots, X_p , and the objective is to determine the levels of the input variables that maximize the response. The general RSM approach includes all or some of the following steps [45]:

- **screening:** when the number of factors is large or when experimentation is expensive, screening experiments are used to reduce the set of factors to those that are most influential to the response(s) being investigated;
- **first order experimentation:** when the starting point is far from the optimum point or when knowledge about the space being investigated is sought, first order models and an approach such as steepest ascent are used to "rapidly and economically move to the vicinity of the optimum";
- **second order experimentation:** after the best solution using first order methods is obtained, a second order model is fit in the region of the first order solution to evaluate

curvature effects and to attempt to improve the solution.

Thus, once the screening to determine the important factors is done, it is time to perform a response surface experiment. Usually, a first order model (3.38) is fitted during the initial stages of the procedure and iterated until the lack of fit is detected, indicating the presence of curvature in the system, and thus, the vicinity of the optimum condition [53].

$$Y = \beta_0 + \sum_{i=1}^p \beta_i X_i + \epsilon \quad (3.38)$$

where β_0 represents the overall mean response, β_i represents the main effects for each factor ($i = 1, 2, \dots, p$), and ϵ represents both errors of approximation and measurements (random errors).

Once the region of the optimum has been found, a second-order model (3.39) is fitted, and an analysis is performed in order to locate the optimum. The polynomial equation that is frequently used to model the response surface is a quadratic model with cross-product terms [48]:

$$Y = \beta_0 + \sum_{i=1}^p \beta_i X_i + \sum_{i=1}^p \sum_{j=1}^p \beta_{ij} X_i X_j + \sum_{i=1}^p \beta_{ii} X_i^2, \quad i \neq j \quad (3.39)$$

where β_0 represents the overall mean response, β_i represents the main effects for each factor ($i = 1, 2, \dots, p$), β_{ij} the two-way interaction between the i^{th} and j^{th} factors, β_{ii} the quadratic effect of the i^{th} factor.

In order to fit the fully second-degree polynomial in Equation (3.39), more than two levels for X variables are needed, usually three or five, that is, +1, 0 and -1 levels. That way we can model curvature, but a total of 3^k (by k representing the number of levels) experiments are needed to take observation at all the possible combinations. In one hand, we have a better estimation of the behavior of our unknown system, on the other hand, it increases the computing time.

3.4.3 Kriging

Since many computer analysis codes are deterministic and therefore not subject to measurement error, the usual measures of uncertainty derived from least-squares residuals have no obvious meaning. Consequently, some statisticians have suggested modeling responses as a combination of a polynomial model plus departures of the form [45]:

$$y(u) = f(u) + Z(u) \quad (3.40)$$

where $y(u)$ is the unknown function of interest (the output of a black-box system for instance), $f(u)$ is a known polynomial function of u , and $Z(u)$ is the realization of a normally distributed Gaussian random process with mean zero, variance σ^2 , and non-zero covariance. While $f(u)$

"globally" approximates the design space, $Z(u)$ creates localized deviations so that the Kriging model interpolates the n_s data points.

As to specify some terms, we can define the variance and the covariance [54].

Definition 19. *Variance*

$$\text{Var}(X) = E[(X - \mu)^2]$$

Such that, X represents a random variable, μ represents the mean of X ($\mu = E[X]$), and $E(X)$ represents the expected value of X (the long-run average value of repetitions of the experiment it represents).

Definition 20. *Covariance*

$$\text{Cov}(X, Y) = E[(X - E[X])(Y - E[Y])]$$

Such that X and Y are random variables and $E(\cdot)$ represents the expected value of the variable.

Definition 19 states that the variance is the expectation of the squared deviation of a random variable from its mean. In other words, it measures how far a set of samples or points are spread out from their average value.

On the other hand, Definition 20 states that the covariance is a measure of the joint variability of two random variables. It is defined as the expected product of their deviations from their individual expected values. In other words, the covariance of two random variables is the measures of the linear dependence between the random variables, meaning that if they are independent, the covariance between them is equal to zero. As a remark, the sign of the covariance shows the tendency in the linear relationship between the variables, and regarding its magnitude, as it is not normalized, its interpretation depends on the magnitudes of the variables X and Y .

Then, Kriging can be defined as an optimal interpolation based on regression against observed z values of surrounding data points, weighted according to spatial covariance values. The advantages of using Kriging are:

- Helps to compensate for the effects of data clustering, assigning individual points within a cluster less weight than isolated data points (or, treating clusters more like single points)
- Gives estimate of estimation error (Kriging variance), along with estimate of the variable itself (but error map is basically a scaled version of a map of distance to nearest data point, so not that unique)
- Availability of estimation error provides basis for stochastic simulation of possible realizations of a random field with a trend component.

Furthermore, without metamodels, the number of simulation runs in an optimisation is the number of optimizer iterations, the number of evaluations per iteration multiplied with the number of Monte Carlo or Latin hypercube samples. With the use of Kriging metamodels the number of runs is reduced to the seeding runs and extra runs for online adaptation of the metamodel. Using this approach, the total number of runs is reduced to a small percentage (5% to 20%) of those otherwise required by the original algorithm [55].

As for its mathematical definition, Goovaerts [56] states that all Kriging estimators are but variants of the basic linear regression estimator $Z^*(\mathbf{u})$ defined as:

$$Z^*(\mathbf{u}) - m(\mathbf{u}) = \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_{\alpha} [Z(\mathbf{u}_{\alpha}) - m(\mathbf{u}_{\alpha})] \quad (3.41)$$

with \mathbf{u} and \mathbf{u}_{α} corresponding to location vectors for estimation point and one of the neighboring data points (indexed by α); $n(\mathbf{u})$ the number of data points in local neighborhood used for estimation of $Z^*(\mathbf{u})$; $m(\mathbf{u})$ and $m(\mathbf{u}_{\alpha})$ are the expected values (means) of $Z(\mathbf{u})$ and $Z(\mathbf{u}_{\alpha})$; $\lambda_{\alpha}(\mathbf{u})$ is the Kriging weight assigned to datum $Z(\mathbf{u}_{\alpha})$ for estimation location \mathbf{u} (same datum will receive different weight for different estimation location).

$Z(\mathbf{u})$ is treated as a random field with its trend component $m(\mathbf{u})$, which generates the residual component:

$$R(\mathbf{u}) = Z(\mathbf{u}) - m(\mathbf{u}) \quad (3.42)$$

Kriging estimates residual at \mathbf{u} as weighted sum of residuals at surrounding data points. The Kriging weights, λ_{α} , are derived from the covariance function or semivariogram, which should characterize residual component. The distinctions between trend and residual are somewhat arbitrary and vary with scale. So, the objective of the method is to determine the weights λ_{α} that minimize the variance of the estimator:

$$\begin{aligned} \text{minimize } & \sigma_E^2(\mathbf{u}) = \text{Var}\{Z^*(\mathbf{u}) - Z(\mathbf{u})\} \\ \text{subject to } & E\{Z^*(\mathbf{u}) - Z(\mathbf{u})\} = 0 \end{aligned} \quad (3.43)$$

Then, the random field (RF) $Z(\mathbf{u})$ is decomposed into residual and trend components:

$$Z(\mathbf{u}) = R(\mathbf{u}) + m(\mathbf{u}) \quad (3.44)$$

with the residual component treated as a random field with stationary mean of zero and a stationary covariance. The residual covariance function is generally derived from the input semivariogram model, which should represent the residual component of the variable.

There are three main variants for the Kriging method: simple, ordinary and universal (also called Kriging with a trend) [57].

For the **Simple Kriging**, it is assumed that the trend component is constant and its mean is known: $m(\mathbf{u}) = m$, such that:

$$Z_{SK}^*(\mathbf{u}) = m + \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_{\alpha}^{SK} [Z(\mathbf{u}_{\alpha}) - m] \quad (3.45)$$

The estimate is automatically unbiased, since $E[Z(\mathbf{u}_\alpha) - m] = 0$, and consequently $E[Z_{SK}^*(\mathbf{u})] = m = E[Z(\mathbf{u})]$. Hence, the estimator error $Z_{SK}^*(\mathbf{u}) - Z(\mathbf{u})$ is a linear combination of random variables representing residuals at the data points \mathbf{u}_α . To minimize the error variance we take the derivative of the error variance, obtained using rules for variance of a linear combination of random variables [57]. At the end, we obtain the following relation:

$$\mathbf{K}\lambda^{SK}(\mathbf{u}) = \mathbf{k} \quad (3.46)$$

where \mathbf{K} is the matrix of covariance between data points, \mathbf{k} is the vector of covariance between the data points and the estimation points, and $\lambda^{SK}(\mathbf{u})$ is the vector of simple Kriging weights for the surrounding data points.

Once we have the Kriging weights, we can compute both the Kriging estimate and the Kriging variance, which is given by

$$\sigma_{SK}^2(\mathbf{u}) = C(0) - \lambda_{SK}^T(\mathbf{u})\mathbf{k} = C(0) - \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha^{SK}(\mathbf{u})C(\mathbf{u}_\alpha - \mathbf{u}) \quad (3.47)$$

For the **Ordinary Kriging**, rather than assuming that the mean is constant over the entire domain, we assume that it is constant in the local neighborhood of each estimation point, that is, $m(\mathbf{u}_\alpha) = m(\mathbf{u})$ for each nearby data value $Z(\mathbf{u}_\alpha)$ that we are using to estimate $Z(\mathbf{u})$ [57]. Another aspect is that the Kriging weight sum is set to one, resulting on the estimator:

$$Z_{OK}^*(\mathbf{u}) = \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha^{OK}(\mathbf{u})Z(\mathbf{u}_\alpha) \text{ with } \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha^{OK}(\mathbf{u}) = 1 \quad (3.48)$$

It is then necessary to add a Lagrange parameter in order to minimize the error variance subject to the unit-sum constraint on the weights, resulting in the following optimization problem:

$$\begin{aligned} \text{minimize } & L = \sigma_E^2(\mathbf{u}) + 2\mu_{OK} \left[1 - \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha(\mathbf{u}) \right] \\ \text{subject to } & \frac{1}{2} \frac{\delta L}{\delta \mu} = 1 - \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha(\mathbf{u}) = 0 \end{aligned} \quad (3.49)$$

Once the Kriging weights (and Lagrange parameter) are obtained, the ordinary Kriging error variance is given by:

$$\sigma_{OK}^2(\mathbf{u}) = C(0) - \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha^{OK}(\mathbf{u})C(\mathbf{u}_\alpha - \mathbf{u}) - \mu_{OK}(\mathbf{u}) \quad (3.50)$$

Finally, the **Universal Kriging** (or Kriging with a trend) is similar to the ordinary one, except that instead of fitting the local mean in the neighborhood of the estimation point, one can fit a linear or a higher-order trend in the coordinates of the data points [57]. For instance, for a planar set of coordinates (x, y) , the local linear trend model is given by:

$$m(\mathbf{u}) = m(x, y) = a_0 + a_1x + a_2y \quad (3.51)$$

The solution for this model is similar to the one presented for the ordinary Kriging, but to the addition of the respective Lagrange terms (2 for given example) and the addition of extra columns and rows in the \mathbf{K} matrix. The universal Kriging is a general case for the ordinary Kriging case previously seen, since the latter can be considered as universal Kriging with a zero-order trend model.

3.5 The adaptive DoE

The methodology used on CADES is the one called adaptive DoE and proposed by Fraisse et al [58]. It allows to both fitting the Kriging model in the global space and near the optimal individuals (Pareto front). The schematic for the methodology is given in figure 11.

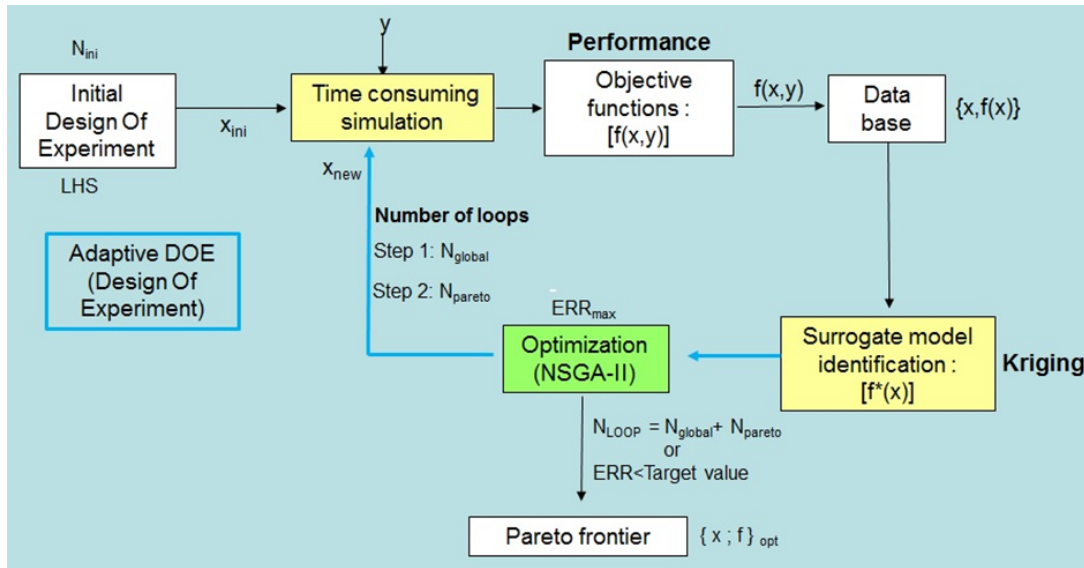


Figure 11 – Methodology adopted for the optimization using Kriging meta-model.

In order to illustrate the progression of the algorithm we have created a toy example with the optimization test functions Rosenbrock and Camelback functions, which surfaces are presented in figure 12:

$$\begin{cases} \text{Rosenbrock} = (1 - x_1)^2 + 100(x_2 - x_1^2)^2 \\ \text{Camelback} = 2x_1^2 - 1.05x_1^4 + x_1^6 + x_1x_2 + x_2^2 \end{cases} \quad (3.52)$$

The algorithm starts by sampling the initial data points using the Latin Hypercube Sampling technique (LHS). Figure 13 depicts the position of those points for the inputs relative to the reference Pareto front and for each phase of the algorithm. For the LHS phase, it corresponds to X_{ini} and for Global Fitting (Step I) and Pareto Improvement (Step II) phases, they correspond to X_{new} .

The set of points is then evaluated using the reference time consuming model (which represents a co-simulation with EnergyPlus/TRNSys when using interoperability) to the

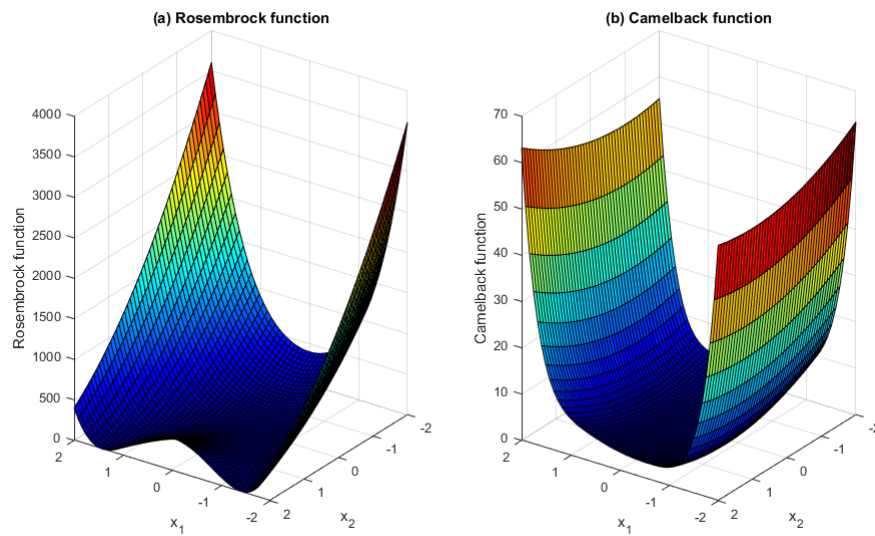


Figure 12 – (a) Rosenbrock Function; (b) Camelback function.

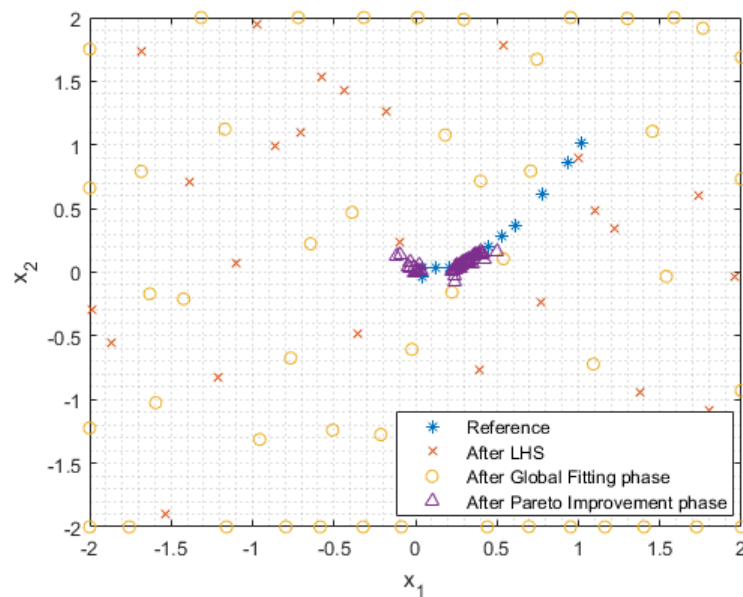


Figure 13 – Start sampling points for each phase of the algorithm.

selected points. Figure 14 shows the placement of those points on the reference surface for each of the phases.

We can notice that on LHS phase we have the recognition of the black-box model. On Global Fitting phase, the points are located also on the verge of the surface, which gives a better knowledge of the surface. Finally, on Pareto Improvement phase, the points converge close to the solution. In fact, the aim of Step 1 is to explore the whole (global) design space. Thus, in Step 2, the identified Kriging model is then trained to run the multi-criteria optimization process using also NSGA-II algorithm to find the optimal solutions (Pareto front).

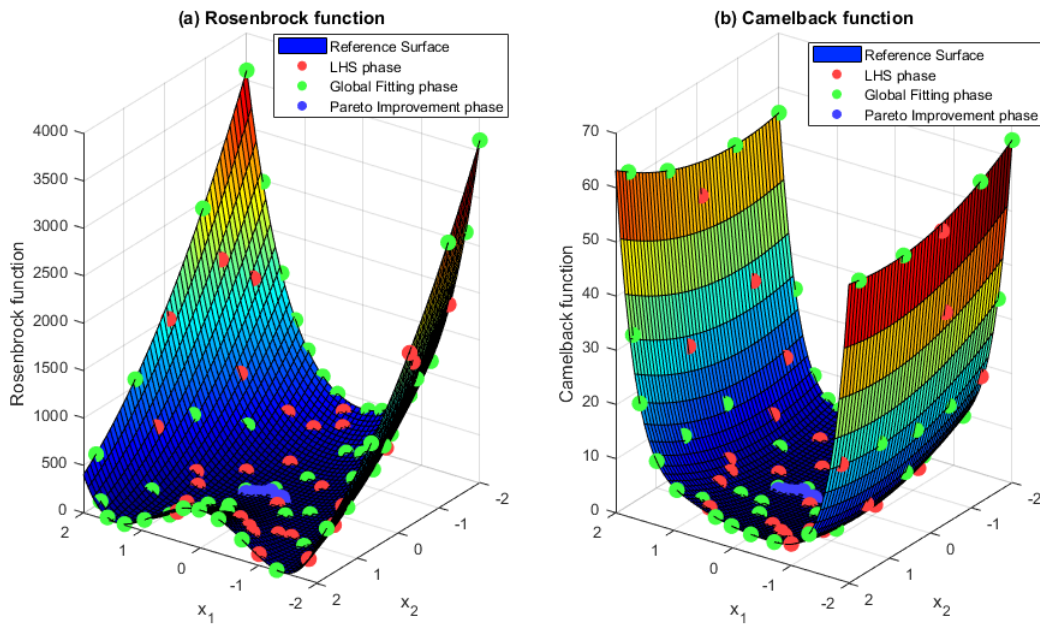


Figure 14 – STD points: output after time consuming simulation.

After the evaluation, the Kriging surrogate model is identified for each of the objective functions. The evolution of the Kriging surface is illustrated in figure 15.

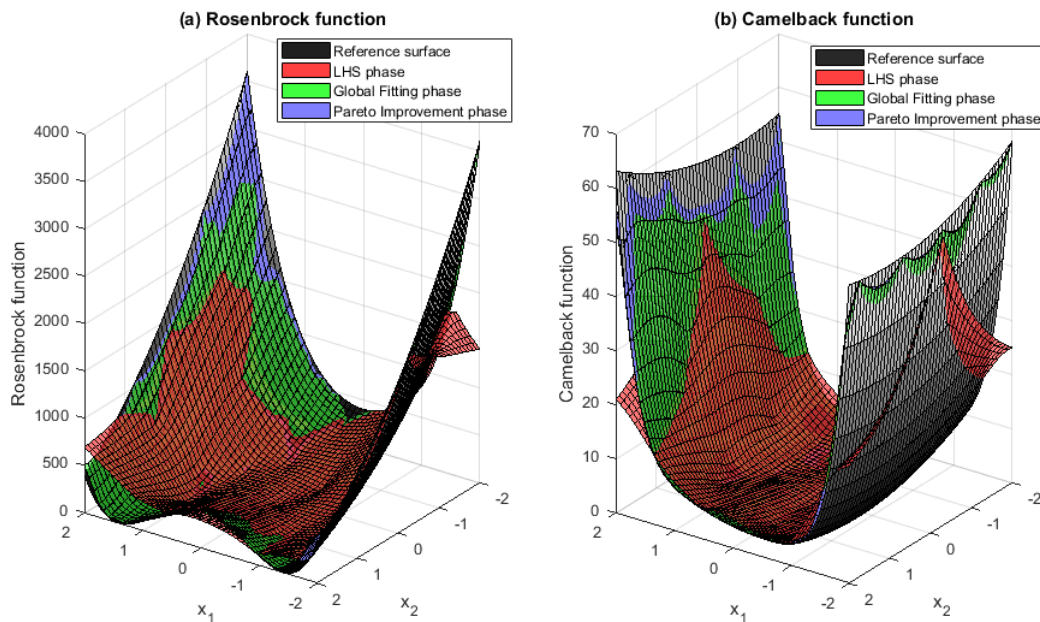


Figure 15 – Kriging surfaces.

It is possible to observe that after each phase, the Kriging surface gets closer to the reference one.

On Step 1, after the NSGA-II optimization, the individual with the maximal error is

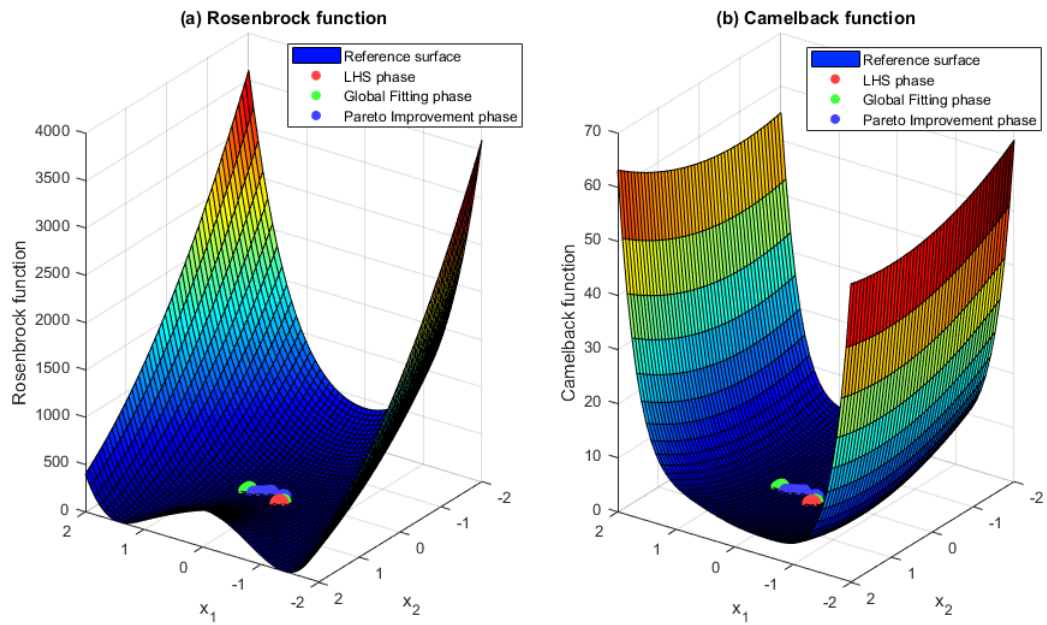


Figure 16 – Metamodel points: output after NSGA-II algorithm.

selected for a new time consuming simulation N_{global} times or until the acceptable error is reached. On Step 2, the selection of the individual is made from the criterion of maximal covariance, so, the individual that presents it is then selected for a new time consuming simulation. For each phase, after the NSGA-II optimization, we get the points presented in Figure 16.

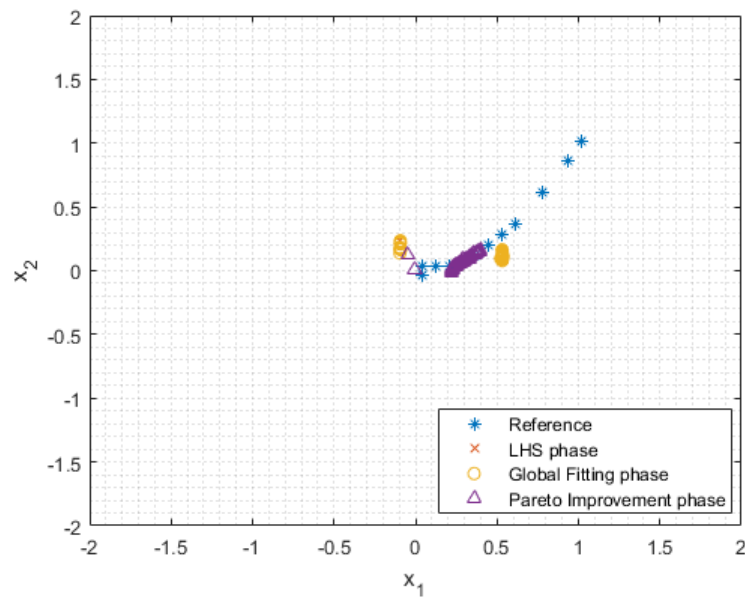


Figure 17 – Corresponding inputs for metamodel points.

The optimization process ends with the maximum number of loops not greater than

the sum of number of individuals in each steps ($N = N_{LHS} + N_{global} + N_{pareto}$) or if the acceptable error target is achieved [58].

The respective inputs for each phase after the NSGA-II optimization is portrayed in Figure 17.

We can observe the approximation to the reference points relative to the Pareto front as the phases progress. This reflects the Pareto front evolution for each phase, as it is shown in Figure 18.

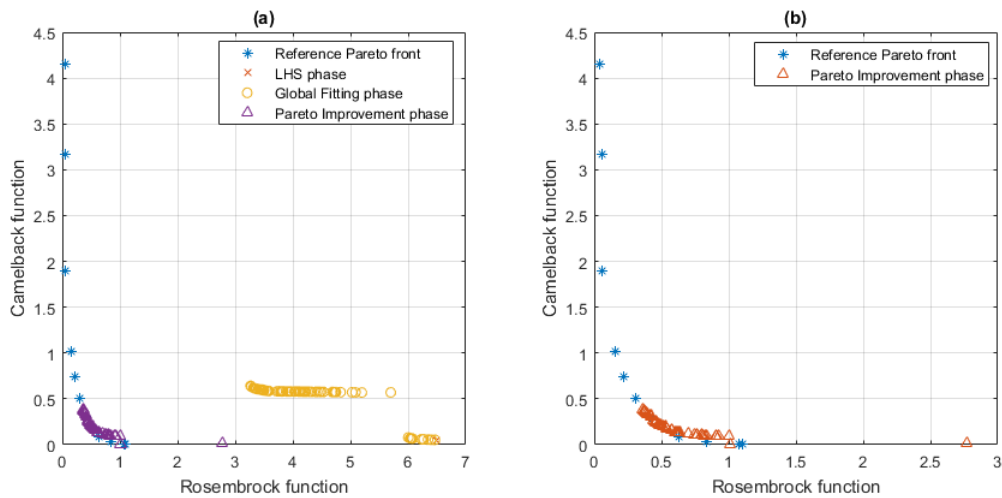


Figure 18 – Pareto front for the toy example.

As expected, the input points from the first phases are not close enough to the ones forming the reference Pareto front as shown in Figure 18(a). However, after the final phase, shown in Figure 18(b), we have a Pareto front formation that gets close to the reference one.

4 Tasks

The schedule for the INTENSE project is depicted on table 4. In the general context presented so far, the work to be performed during this internship is related to task 5 (Development of the EIT), with more emphasis on sub-task 5.3 - case study, validation and dissemination:

- Definition of the collective building (envelope and selected systems) which will serve as a basis for studying the LCA / carbon footprint approach in task 4.1;
- Using the tool developed in task 5.2;
- Sensitivity study related to the number of years of simulation, and the choice of these years according to the predefined evolution scenarios (energy cost, climate, renovation works ...), with the aim if necessary to reduce the times of calculation;
- The results obtained with the EIT will be compared with those of a conventional design approach (type RT2012, with the support of simulation tools such as Pleiades-Comfie or EnergyPlus). This will validate both the proposed methodology, verify the consistency of results and estimate the gain achieved through the EIT. The validation of the methodology will also consist in evaluating the contribution of the EIT for the decision-maker who wishes to go to high-performance buildings with a global vision;
- Sensitivity analysis of certain parameters having an uncertain character. In order to improve the evaluation process, it is essential to appreciate the uncertainties associated with the input parameters of the model. In this sensitivity analysis some parameters will be studied in more detail such as the implementation, the weather, the cost of energy. This will assess the uncertainty of the results and identify the most influential parameters to which particular attention should be paid;
- Taking into account the role of the occupant on overall performance;
- Concerning the dissemination of the EIT, a seminar will be organized at the end of the project to present the results obtained on the case study. It will be open to a potentially user public: design offices with architects, researchers and industrialists. It will be an opportunity to develop new partnerships between the industrial world, research and construction.

Although the main objective of the internship is disclaimed on sub-task 5.3, aspects of other tasks will be executed, since the EIT is still in development, so, bugs and improvements can still be found on the way to its accomplishment.

The work to be performed and the one presented in sub-task 5.3 and other eventual sub-tasks were divided as follows:

- Literature review on subjects related to multi-objective optimization, genetic algorithms, sampling methods (Latin Hyper Cube more specifically), Design of Experiments, surrogate models and Multi-Criteria Decision Making (MCDM) in order to understand the undergoing process of sampling and optimization.
- Introduction to the use of CADES v3 software (software to be developed / validated during the project) and creation of a tutorial for its new version CADES v4 beta.
- Testing and validation of software results and performance, starting with simple models and advancing to the case study.
- Sensitivity analysis of the results.
- Creation of a training tutorial to be presented in Simurex (SIMULATION et Retour d'EXpérience).
- Writing the report and preparing the presentation for defense.

The final Gantt graph for the tasks to be developed is presented on Figure 19.

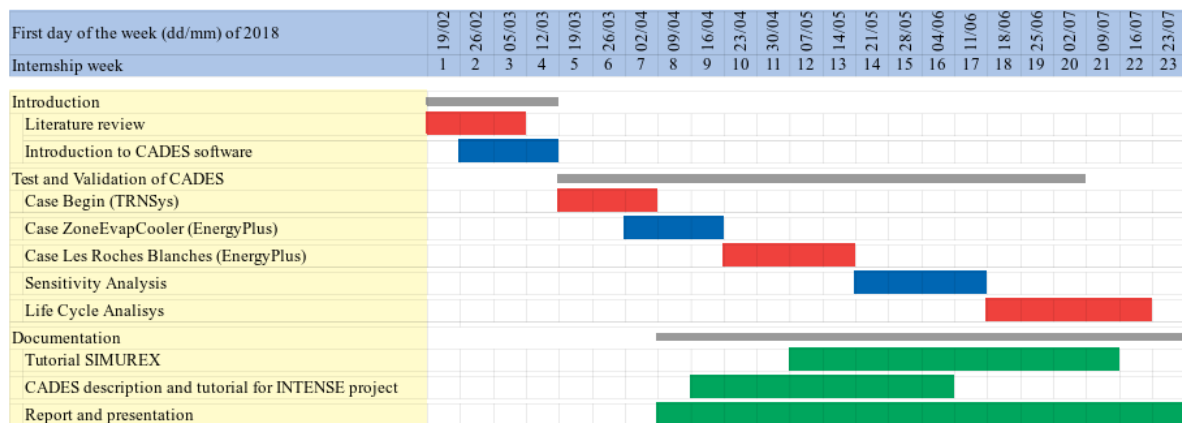


Figure 19 – Final Gantt graph for the internship.

5 Tasks development

This chapter presents the development of the tasks done during the internship. It starts with a presentation of the tutorial developed to present the new version of the software CADES v4 and a short explanation on interchangeability. It is also included in this chapter, the cases of study *Begin* and *Les Roches Blanches*.

5.1 CADES

5.1.1 Basic Tutorial

The development of a tutorial, set as a milestone for the project (L8), has been developed. It contains basic functionalities, optimization examples for test functions, explanation on the interchangeability between CADES and TRNSys and EnergyPlus, the application of the proposed algorithm (the adaptive DoE) called on the software as Response Surface method, how to view the solutions and finally, a section on sensitivity analysis. The tutorial is fully presented on [Appendix A](#).

5.1.2 Interchangeability

The interoperability between CADES and the modeling and simulation software is made by creating a new model in CADES. [Figure 20](#) presents a simplified scheme demonstrating such interchangeability, as further information is present in the tutorial on [Appendix A](#).

Matlab interoperability, however, is made differently. We first need to declare the GroupID, ArtifactID and Version of the model (gav), found directly under the model properties. We also need to import the classes libraries and methods relative to MatlabMuseCaller. We then create a variable from the method declared and finally load the respective MUSE to the declared gav. After that we can get information, for instance, on the Inputs, Outputs and Jacobians on the model. [Figure 21](#) presents the Matlab code for the steps explained above.

Once the initial set up is done, we can proceed to the calculations. [Figure 22](#) presents the code to set a value to a Facet, with the aid of the command `caller.setStaticFacetVariableValue()`, and then call the co-simulation via MUSE.

After the computation, we can retrieve the output values and store them by using the command `caller.getStaticFacetVariableValue()`. As a remark, this interchangeability is not exclusive to TRNSys or EnergyPlus models, but to any model created and declared on CADES.

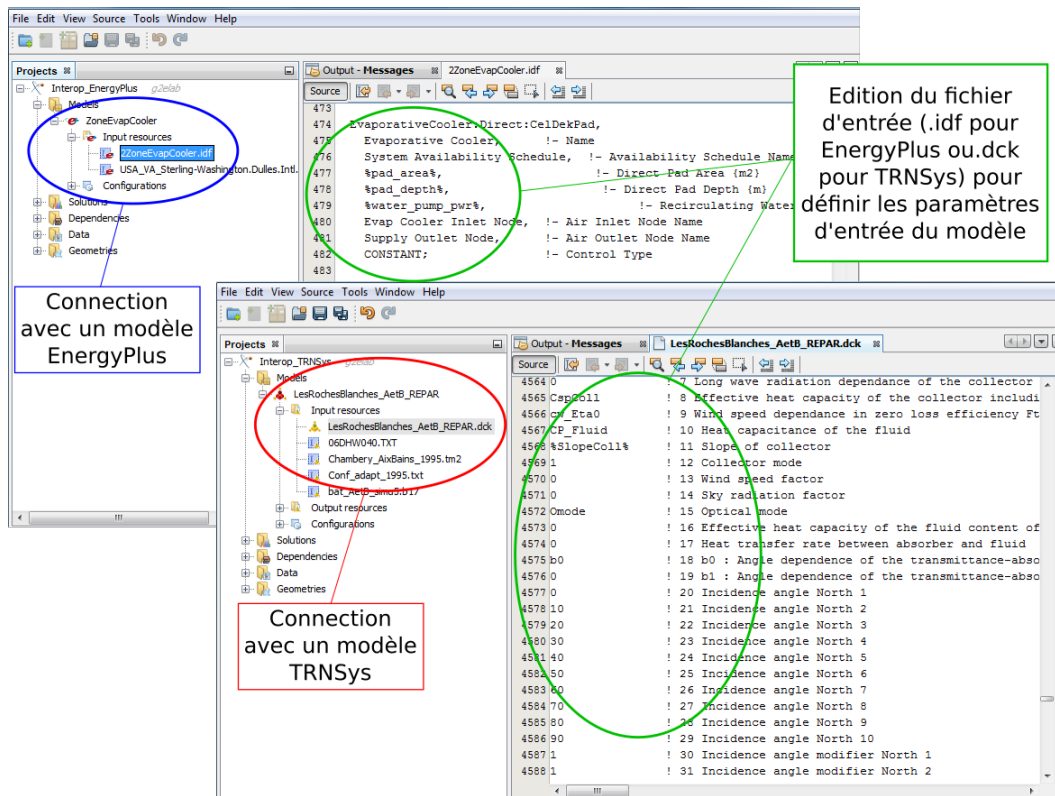


Figure 20 – Example of inputs and outputs declaration for TRNSys and EnergyPlus.

```

9 - gav = 'org.gu.g2elab:tutorial.Begin:1.0';
10 - import org.gu.vesta.cades.scripting.matlab.plugin.MatlabMuseCaller;
11 - caller = MatlabMuseCaller();
12 - caller.loadMuse (gav);
13 - sf_inputs = caller.getStaticFacetInputNames;
14 - sf_outputs = caller.getStaticFacetOutputNames;
15 - jf_inputs = caller.getJacobianFacetInputNames;

```

Figure 21 – Interoperability with Matlab - set up.

```

26 - tic;|
27 - for i=1:nbPoints
28 -     caller.setStaticFacetVariableValue('collector_area',collectorArea(i));
29 -     disp(i/nbPoints*100)
30 -     for j=1:nbPoints
31 -         caller.setStaticFacetVariableValue('slope_of_surface',collectorSlope(j));
32 -         successStaticCompute = caller.computeStaticFacet;
33 -         if (~successStaticCompute), disp('Error in static computing'), break; end
34 -         Type25c.QAux(i,j) = caller.getStaticFacetVariableValue('Type25c.QAux');
35 -         Type25c.QColl(i,j) = caller.getStaticFacetVariableValue('Type25c.QColl');
36 -         Type25c.Fsol(i,j) = caller.getStaticFacetVariableValue('Type25c.Fsol');
37 -     end
38 - end

```

Figure 22 – Interoperability with Matlab - computation.

5.2 Case Begin

A simpler example than the objective case study has been used for the interchangeability test and validation for a TRNSys model. The system to be analyzed is so called Begin, and it is present as an example in TRNSys software. Figure 23 depicts the system and the model, whereas figure 24 shows it modeled into TRNSys user interface.

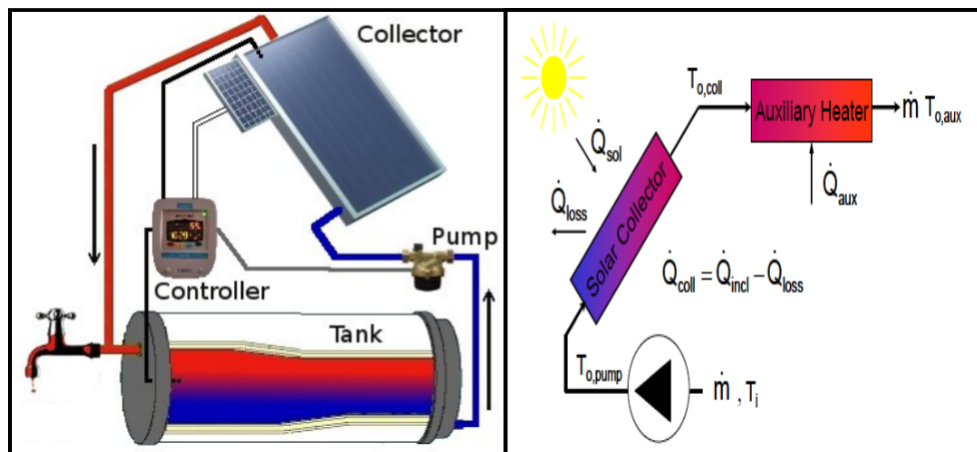


Figure 23 – Case Begin: (a) Scheme of the physical system; (b) Physical model of the system.

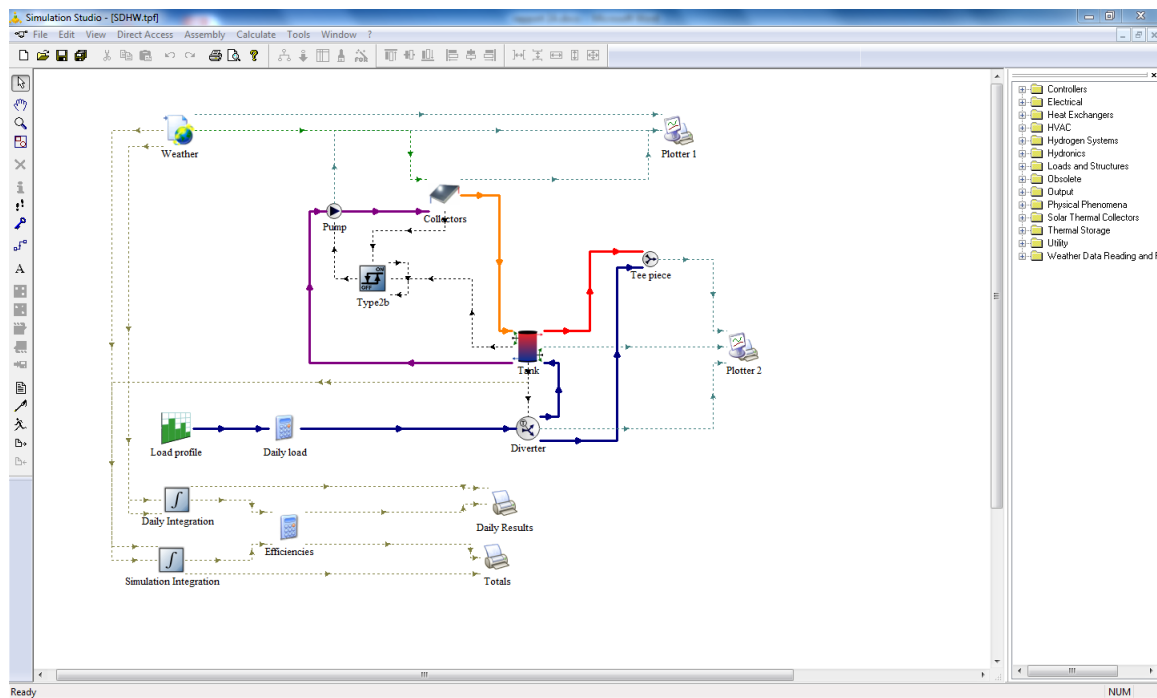


Figure 24 – Modeling of case Begin into TRNSys software.

For this case, we have as inputs

- the solar panel surface area, varying between 1 and 10 m^2 ;
- the solar panel inclination, varying between 0 and 90 degrees.

As outputs and objectives:

- maximize the energy collected by the solar panel (\dot{Q}_{Coll});
- minimize the energy consumed by the auxiliary system (\dot{Q}_{Aux}).

As a remark, the scope of this internship does not include modeling the cases, since, as mentioned before, the optimization is performed over a black-box model. The pictures presented have the intent of clarifying how a model is created and obtained for the Optimization, using the desirable interchangeable software. So, for the TRNSys model, in the end, we get a **.dck** file, which is a text file containing data on parameters (its configuration to determine inputs and outputs will be explicit in the next chapter), and a **.tm2** file, containing the weather data for the specific location.

So, the aim of this simplified case is to build a front Pareto from an exhaustive data acquisition. Once this reference is build, vary the the number of individual for the LHS phase, for the Kriging surface and for the Pareto improvement, as shown in subsection 3.5. From the acquired data, compare the evolution of the front Pareto according to the different number set for the latter parameters. The goal of this analysis is helping to understand the main case study, Les Roches Blanches, because, due to the complexity of its model, it's not viable to obtain a reference front Pareto as made for the case Begin.

5.3 Case *Les Roches Blanches*

The specifications for the building have been previously developed by ARMINES, Albedo Énergie and ENTPE [59] in previous tasks of the INTENSE project. The building studied is depicted in figure 25. It has four levels and a living area of about 1200 m^2 . The building is composed of 17 apartments, a landing, a garage/basement and attic. Each of these areas has been simulated as a thermal zone. The model has 20 thermal zones in total. It is located in Chambéry (Savoie, France) at an altitude of 235 m.

The buildings walls, intermediate floor, partitions and floor are made of concrete. Walls have gypsum plaster board and exterior insulation. Floor on crawl space has Polystyrene insulation; however the Flat roof and Floor under attic have polyurethane insulation and glass wool insulation respectively. The windows in rooms are PVC double glazed windows with overall thermal performance coefficient $U_w = 1.3 W/(m^2.K)$. The number of occupants in an



Figure 25 – South and East views of the building *Les Roches Blanches*.

apartment is kept equal to the number of room in the apartment. The occupancy for different hours during weekdays and weekend are varied from 25% - 50% - 100%. The internal heat gains of housing (related to cooking, electrical appliances, etc.) are taken into account on hourly basis and vary between 1.5 W/m^2 to 7.68 W/m^2 . The cooling and natural ventilation have been taken into account from 15 July to 15 August. The infiltration was considered 0.15 vol.h^{-1} . The mechanical ventilation varies with size of the apartment. Dynamic thermal simulations were run in EnergyPlus 8.8.0 software. This software models the thermal comfort, ventilation, infiltration, internal heat gains, and the heating and cooling loads of the building. The weather file (Meteonorm) used is a typical year weather of the city of Chambéry (Savoie, France) [58].

5.3.1 EnergyPlus - Decision parameters and objective functions

In order to optimize the building, decision parameters and objective functions have been defined. The decision parameters used to test the tool and their range of variation are presented in Table 5.

Table 5 – Decision parameters for the EnergyPlus model for *Les Roches Blanches*.

Decision parameters	Range of variation	Area (m^2)
The thickness of insulation in the exterior walls	0.1 to 0.4 m	906
The thickness of the slab of the intermediate floor	0.1 to 0.25 m	869
The thickness of insulation in the floor on the outside	0.1 to 0.4 m	216
The thickness of insulation in the flat roof	0.1 to 0.4 m	243
The U_w value of the windows	0.9 to $2.5 \text{ W/(m}^2 \cdot \text{K)}$	207
The building orientation	0 to 270°	-

The functions used to evaluate the decision parameters are the sum of heating and cooling loads supplied by the system to the building and the cost related to the changes. The heating and cooling loads are directly calculated by EnergyPlus and are related to the comfort

of the occupants. They represent the energy demand of the building. The cost equations were developed on Task 3 by the LGCB partner. They include the price of the material and its installation. Costs are expressed in €/m², excluding taxes:

- Insulation thickness variation cost of the wall = $520.1x + 73.603$;
- Insulation thickness variation cost of the floor on the outside = $53 + 50(x - 0.1)$;
- Insulation thickness variation cost of the roof terrace = $40.585e^{4.4882x}$;
- Cost of the intermediate floor = $663.25y + 57.49$;
- Variation cost of the U_w of the windows = $110 \left(1 - \frac{1}{1 + e^{-30(z-1.15)}} \right) + 45$.

where x is the insulation thickness in meters, y is the concrete slab thickness in meters and z is U_w , in W/m^2K .

The objective is to find the minimum between the sum of heating and cooling loads, and the minimum cost of the operation. As a recall, because it is a multi-objective black-box optimization, and due to the size and complexity of the problem, we have no knowledge of the optimal solution. For an estimation of time, each calculation (for a given input, to get the corresponding outputs) takes around 3 minutes. For example, for an exhaustive calculation with 50 points for each of the 6 inputs would take $50^6 * 3$ minutes. Even if we use a more powerful machine that reduces calculation time and are capable of parallel-thread, we still have the combinatorial explosion because of the number of inputs. It is possible however to make comparisons with the results from other optimization methods.

6 Results

The computations and optimizations were made in a computer with an Intel(R) Core(TM) i5-6600 CPU @ 3.30 Ghz, 8.00Gb of memory RAM, running Windows 7 Professional 64 bits as operating system.

It is important to remark that even if the modeling on both EnergyPlus and TRNSys are equivalent, there might be a difference in the optimization results, since EnergyPlus is a software oriented to the building envelope, whereas TRNSys is oriented to system modeling.

6.1 Case Begin

As mentioned before, the case Begin is considered a "simple" case, since it doesn't have many parameters in its modeling. It is a case with two inputs (Collector Area and Collector Slope) and two outputs (\dot{Q}_{Coll} and \dot{Q}_{Aux}), hence, we can recreate an output surface for each output and graphically observe the behavior of the algorithm on each output.

6.1.1 Reference surfaces and reference Pareto front

So, in order to create those surfaces representing the outputs, we divided each input range in n points equally spaced between the minimum and maximum of each of those inputs, as determined on Chapter 5. Once it was done, an exhaustive computation was done to determine the outputs in order to determine a reference for comparison with the results from the adaptive DoE. Besides, a Pareto front was calculated.

Two exhaustive calculations were made, one for $n = 200$ that took around 6,25 hours, and a second one for $n = 100$, which took 1,5 hours to run. Regarding the proximity of the curves, we can analyse the relative root mean square error between the two data sets, which is of 0.0805%. It is still possible to determine the R^2 (coefficient of determination) of 0.999994673¹ between the two Pareto fronts. From those results we can say that no extensive calculation need to be done, and a reference with $n = 100$ is already good enough for the validation of the algorithm. Figure 26 presents the reference surfaces for $n = 100$.

Figure 27(b) depicts in blue all solutions from this exhaustive computation, and in red, the non-dominated solutions forming the front Pareto. Figure 27(a) presents the respective input points for the points forming the front Pareto.

¹ The coefficient of determination, R^2 , is the proportion of the variance in the dependent variable that is predictable from the independent variable(s), and its value approaching to 1 indicates accuracy of one curve with respect to the reference.

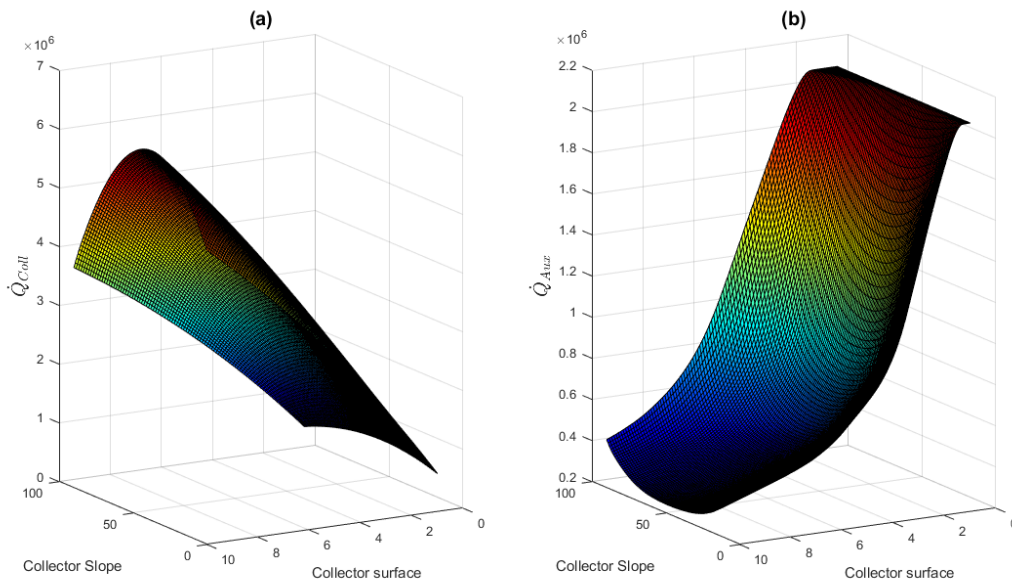


Figure 26 – Reference surfaces for (a) \dot{Q}_{Coll} and (b) \dot{Q}_{Aux} .

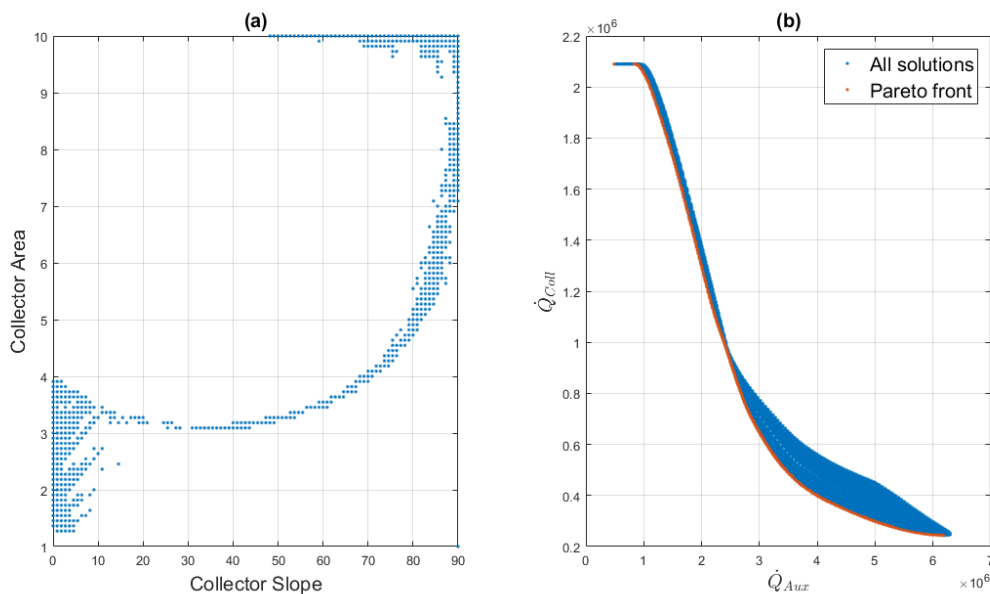


Figure 27 – Pareto front: (a) related inputs and (b) All solutions vs Pareto formation.

We can also observe how those points are distributed throughout the reference surfaces, as shown in Figure 28. We can clearly observe on Figure 28(c) that the intersection of the surfaces do not form any sort of solution. Intuitively we might think that, as in the case of constraints, the intersection would represent an active set², but this is not the case here. Another aspect to remark is that the best solutions for this multi-objective optimization (the

² The active set at x is made up of those constraints $g_i(x)$ that are active at the current point. The active set is particularly important in optimization theory as it determines which constraints will influence the final result of optimization.

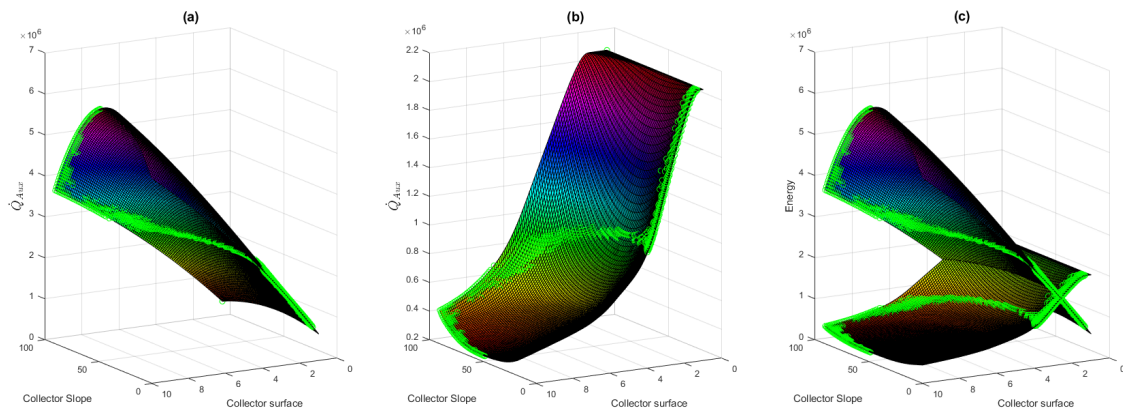


Figure 28 – Reference surfaces with Pareto front points for (a) \dot{Q}_{Coll} , (b) \dot{Q}_{Aux} , (c) Superposition of surfaces.

ones forming the Pareto front) are not on the ones where the derivative of the surfaces are zero (max or min points). This happens because of the criteria of dominance (Definition 2) presented before. Those plotted points represent the best solutions for the problem, and the MCDM tool is used upon them to determine an adequate solution.

6.1.2 Computation time analysis

Once the reference surfaces for the outputs and the reference Pareto front could be calculated, we can proceed with the validation of the proposed Adaptive DoE algorithm. For that, seven different experiments were done, with $N = 30, 75, 120, 165, 210, 255$ and 300 individuals evenly distributed on each phase (for instance, for $N = 30$, we have 10 individuals for the LHS phase, 10 individuals for the Global Fitting phase and 10 individuals for the Pareto Improvement phase).

The first analysis done is related to the computational time and it is depicted in Figure 29, where the time ration is computed as the ratio between the exhaustive computation elapsed time by the optimization elapsed time.

For this case, according to Figure 29(a) we can not find a linear correspondence between the optimization elapsed time and the number of individuals, although we can observe that with a larger number of individuals, this relation becomes more linear. Moreover, we can observe on Figure 29(b) that the time ratio has an asymptotic behavior, meaning that increasing the number of individuals from a certain point will decrease the influence in the ration. Besides, this figure depicts how much faster the optimization turns with respect to the reference. For instance, for 30 individuals, we have an optimization response over 300 times faster than the exhaustive, and for 300 individuals, 7.6 times faster.

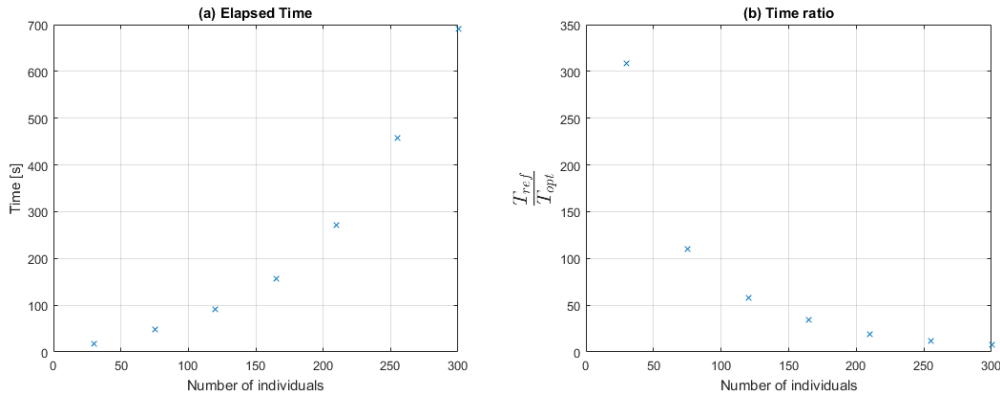


Figure 29 – (a) Time of computation according to the number of individuals; (b) Time ratio according to the number of individuals.

6.1.3 The coefficient of determination and the RMSE

The coefficient of determination, R^2 , is the proportion of the variance in the dependent variable that is predictable from the independent variable(s), and its value approaching to 1 indicates accuracy of one curve with respect to the reference. This coefficient is here determined between the individuals of the reference model Pareto front and the Pareto front individuals obtained from the meta-model. It is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (6.1)$$

Such that n is the number of measurements, y_i is the value of the i^{th} measurement, \hat{y} the value of its corresponding prediction (in our case, the meta-model Pareto front), and finally, \bar{y} is the mean value of the measurements.

This coefficient as a condition to determine if increasing the number of individual improves the output of the algorithm with respect to the reference. Along with the R^2 we have used a relative Root Mean Square Error (RMSE), also with the intent of analysing the progress according to the number of individuals.

As we can observe from Figure 30, on (a), increasing the number of individuals increases the value of R^2 until a value close to 1, as desired; on (b) decreases the relative RMSE value, but, because of the identification and metamodeling, the value is not close to zero. However, we can observe that the slope in both cases is also diminished whereas we increase the number of individuals.

In real cases simulations (outside of the scope of validation) we will not have a reference to compare our results with. That said, these coefficients are compelling for validating the methodology, but not to determine the trade-off between computational time and precision.

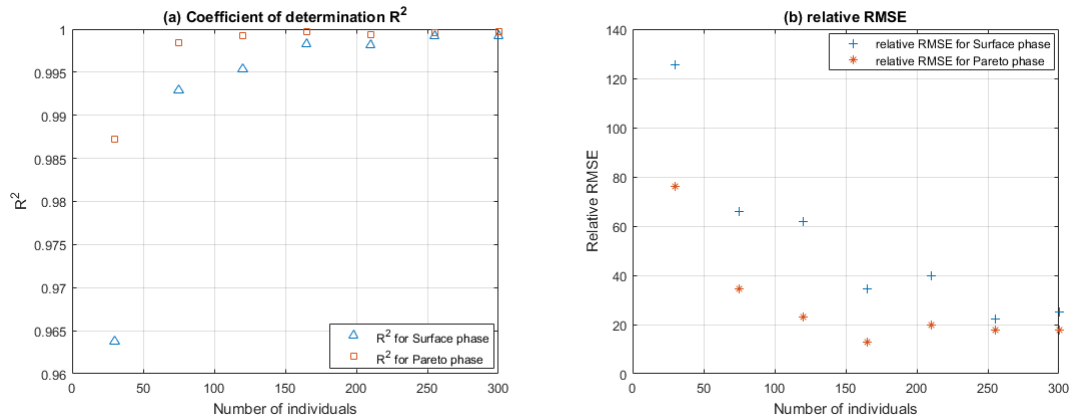


Figure 30 – Case Begin: (a) R^2 coefficient; (b) relative RMSE.

6.1.4 Pareto evolution for different number of individuals

Another valid analysis is made from observing the Pareto front evolution according to the number of individuals. Figure 31 presents for each of the three phases, the evolution of the Pareto front according to the number of individuals. To enhance visibility and understanding, we have plotted four out of the 7 cases, for $N=30$, $N=120$, $N=210$ and $N=300$.

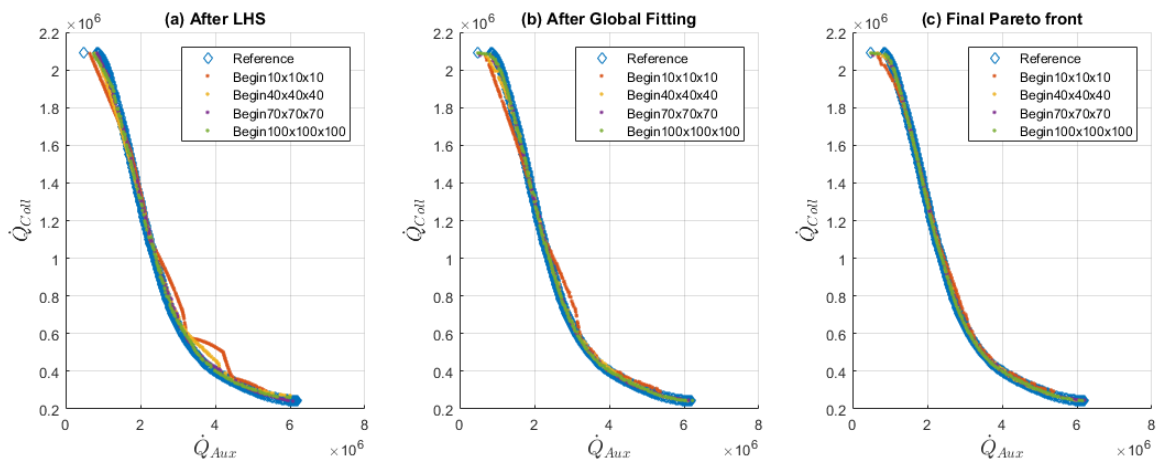


Figure 31 – The Pareto front evolution in the three phases of the algorithm for different number of individuals for case Begin.

We can observe throughout the experiments that the Pareto front from the optimization approaches the reference Pareto front when increasing the number of individuals. Although this is desirable, it is important to remember the trade off between the precision and the computation time of the optimization.

For this simple case, it is harder to detect the improvement of the Pareto front, since the set of points are well superposed. However, we can observe some zones, such as on the top and close to the centre, where we evolution is visible. This is more explicit for a more complex case, as the next case, *Les Roches Blanches*.

6.1.5 Input points evolution

We can observe the evolution of the algorithm by analysing the input points forming the Pareto front. Figure 32 shows the evolution of the input points related to the Pareto front for the case Begin with 300 individuals.

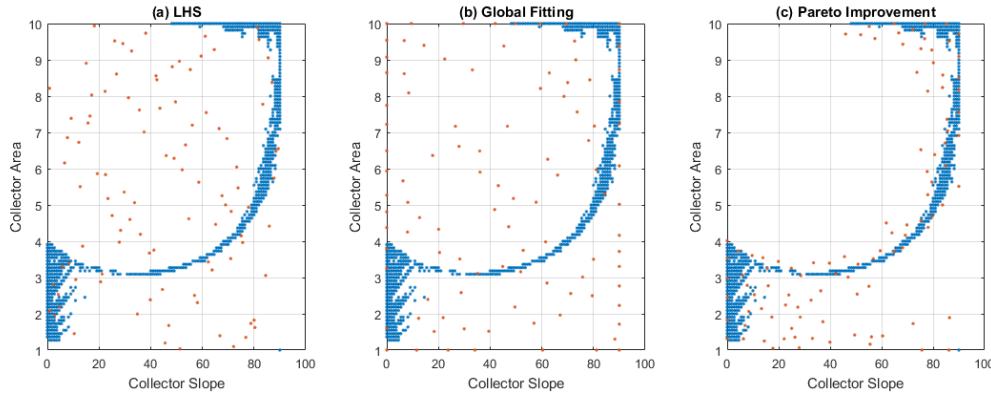


Figure 32 – Input points values for each phase; in blue the points from the reference Pareto, in red the solution points from the optimization.

We can observe that the points from the optimization solution approaches the ones from the reference. We can rely on a 3D plot in order to better observe the behavior of those points. Figures 33 to 35 presents this evolution. The STD points correspond to the points from the database, whereas the meta-model points represent the points after the NSGA-II optimization from the adaptive DoE (Figure 11).

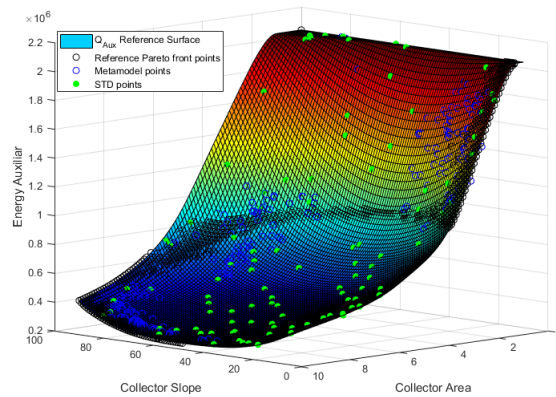


Figure 33 – STD and meta-model points at the end of LHS phase.

At the end of the LHS phase, there might not be too much correspondence between the Pareto front reference points and the STD/meta-model points, since this phase only rebuilds the model from the given inputs.

For the Global Fitting phase, which corresponds to the Step I from the adaptive DoE algorithm, we can observe that the points spread throughout the surface and we can also

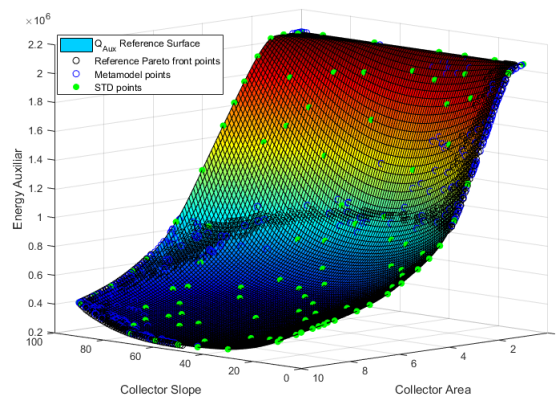


Figure 34 – STD and meta-model points at the end of Global Fitting phase.

observe the points of the verge of the surface. This is a desirable behavior, since the aim of this phase is to explore the the global design space.

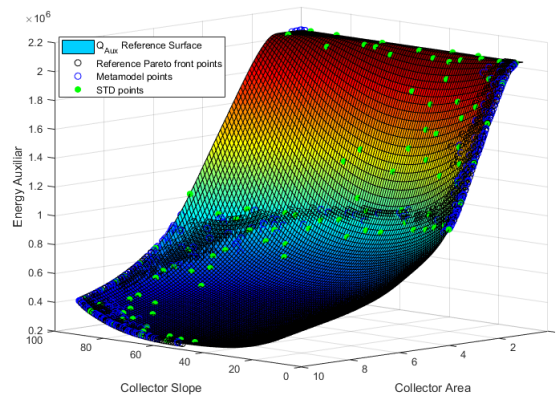


Figure 35 – STD and meta-model points at the end of Pareto Improvement phase.

Finally, at the end of the Pareto improvement phase, we can observe the convergence of the points to the Pareto front reference inputs. Since we have a gap from the optimization result to the reference, because of errors related to meta-modeling for instance, the result is not exact. The discussion on how the individuals are selected and how to determine if the increasing on the number of individuals is justified is given on the next subsections.

The same analysis can be made for the second output surface, which will give similar results regarding the objective of each phase.

6.1.6 Pareto evolution for case Begin with 300 individuals

Finally, for the instance with higher number of individuals, we can observe the Pareto front evolution on each phase.

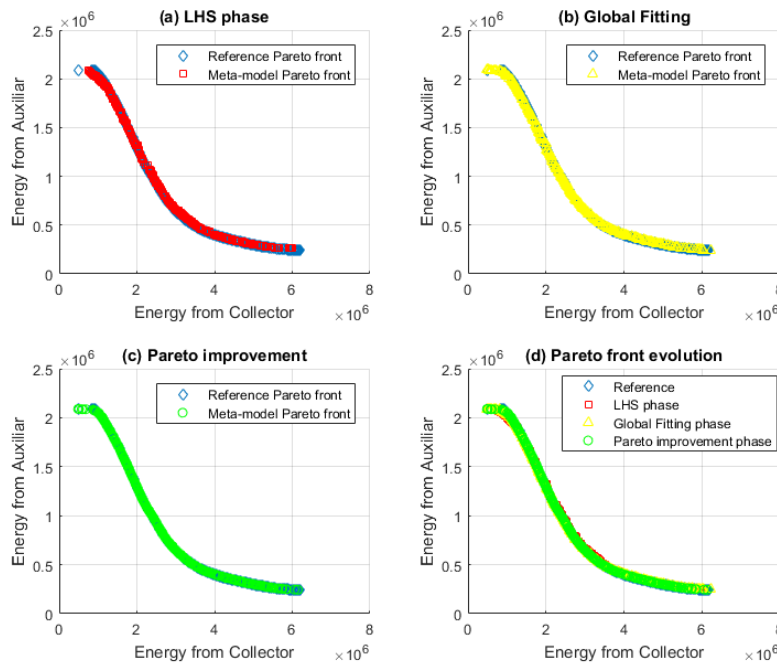


Figure 36 – Pareto front evolution for case Begin with 300 individuals

Since this is a simpler case, we have the Pareto fronts close one to another on all phases (we will see on the other cases that this is not often the case). Even though, we can observe that after each phase the Pareto front from the optimization gets even closer from the reference. Result that can be validated by the previous R^2 and RMSE analysis, where the coefficient from the Global Fitting phase is smaller than from the Pareto improvement phase (0,9992 against 0.9997) and the error greater (24.94% against 17.81%).

6.1.7 Decision making example

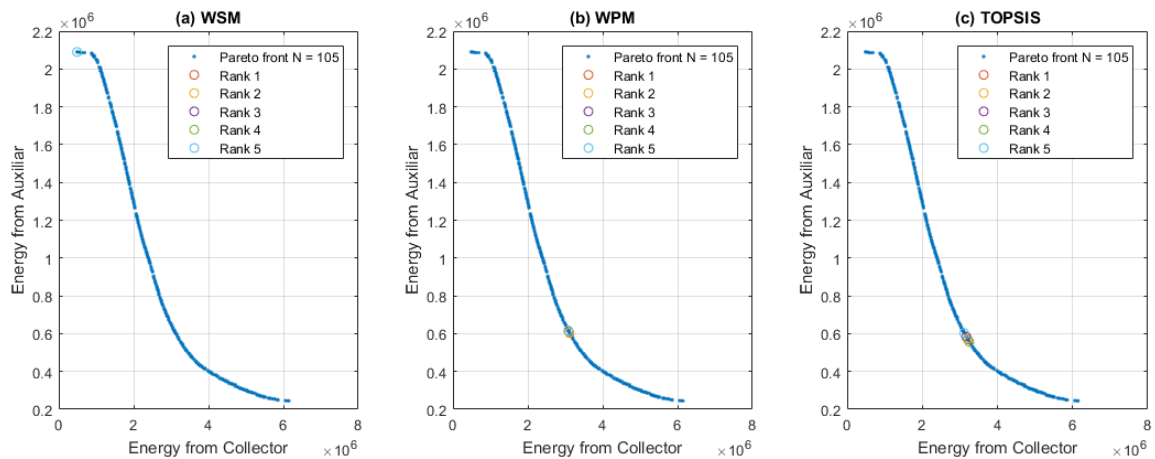
Let us consider that in modeling and design phases we have established that all criteria have the same weight. As we have two objective functions for this case, the System Auxiliary Energy and the Photo-voltaic Collector Energy, we have a split weight of 0.5 for each criteria. That way, Table 6 presents the results for different methods of MCDM in order to determine the best alternative among the 423 ones from the Pareto front.

It is easy to remark that each of the methods used have a different ranking of best alternatives. However, as the Pareto front formation is not sorted, and to have a better representation of the result, Figure 37 depicts the location of the results from Table 9 on the Pareto front for the referred case.

As explained before, WSM is better used for single objectives, since it is not dimensionless, and this characteristic can be identified here since its result diverges from the other methods. Besides, we can observe that WPM and TOPSIS methods bring closer results than

Table 6 – Rank of 5 best alternatives for different MCDM methods for case Begin for N = 300 individuals.

Solution Rank	Method		
	WSM	WPM	TOPSIS
1	126	397	319
2	186	335	363
3	197	148	129
4	199	298	372
5	225	167	397

Figure 37 – Rank of 5 best alternatives for different MCDM methods for case *Les Roches Blanches* for N = 105 individuals.

the ones presented by WSM, and the decision on the method is up to the decision maker.

6.1.8 Alternative-Criteria Sensitivity analysis

The alternative-criteria sensitivity analysis is not on the scope of this internship, however, it is interesting to discuss the ranking of the chosen alternatives. So, in order to illustrate the importance of the weight of each criteria, the weights were changed from 0.1 to 0.9 (here we have considered that the weights sum is equal to one). Table 7 presents the results of the five best alternatives (out of 423) for each of the weights.

Table 7 – TOPSIS top 5 solutions for case Begin with for N = 300 individuals for different criteria weights.

Solution Rank	Weights for $\dot{Q}_{Aux}/\dot{Q}_{Coll}$								
	0.1/0.9	0.2/0.8	0.3/0.7	0.4/0.6	0.5/0.5	0.6/0.4	0.7/0.3	0.8/0.2	0.9/0.1
1	126	126	219	325	319	315	76	229	55
2	186	186	299	330	363	337	96	184	19
3	197	197	130	332	129	117	147	93	69
4	199	199	38	132	372	153	187	94	264
5	225	225	270	374	397	59	191	113	265

We can observe that the changing the criteria weights brings a change on the top criteria. As a complement to Table 7, we can observe in Figure 38 the position of the top solution on the Pareto front for $N = 300$ individuals.

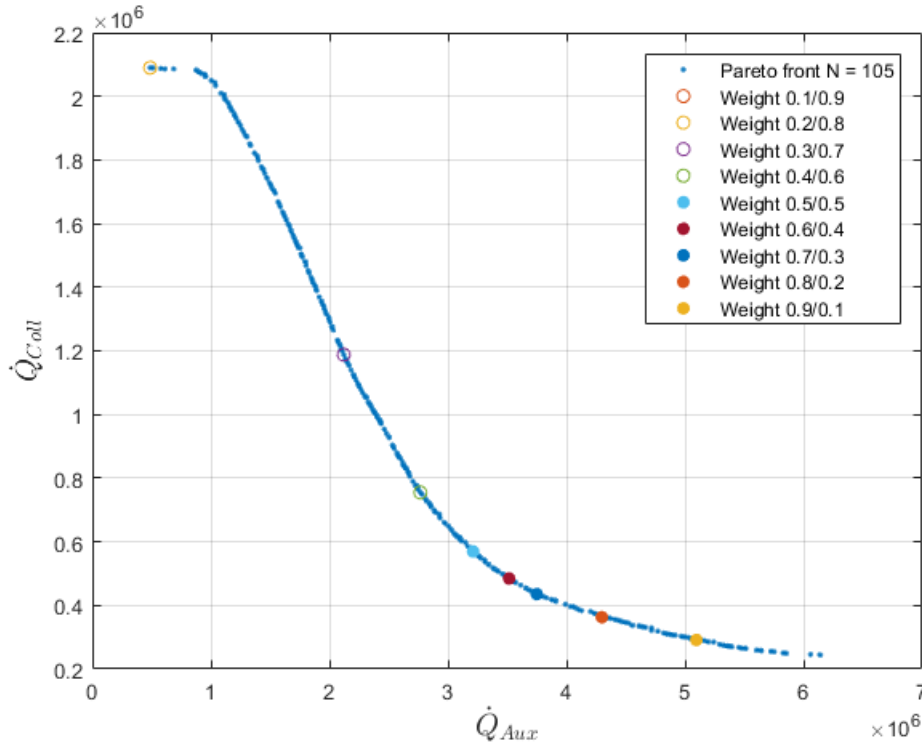


Figure 38 – Best TOPSIS alternatives for case Begin for $N = 300$ individuals for different criteria weights.

We can clearly observe the influence of a given criterion weight to each of the objectives, as the top solution wanders throughout the Pareto front solutions.

To determine the criticality of a determined rank presented in Table 6, we can use the formulations proposed by [42]. However, this is not in the interest of the internship as this tool is not implemented on CADES at this time.

6.1.9 Input-Output Sensitivity analysis

As a disclosure, in general, a study on the inputs uncertainties of the system is done during the project. That way, we can obtain a sensitivity analysis according to the given study. It is not our case however, since this case study is an example from TRNSys.

In order to observe the global sensitivity given by Sobol method, we gave to each one of the two inputs, an equal relative uniform variation of 1%, 10% and 20%. Figure 39 depicts the results of this analysis according to the number of evaluations. As a recall, the evaluations represent the number of samples of each coordinate from the unitary hypercube \mathbb{H}^n , and increasing its number, exponentially increases the number of iterations.

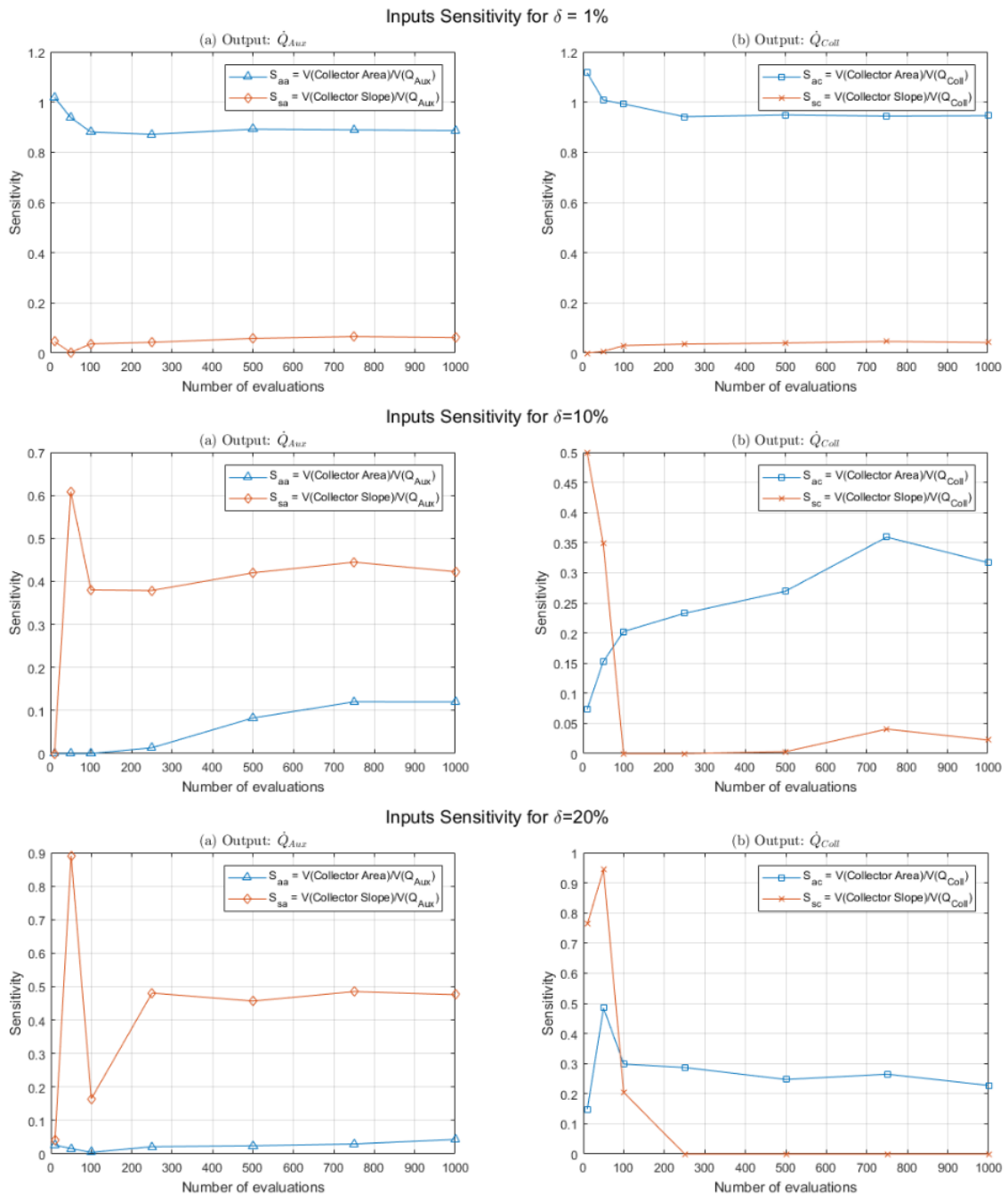


Figure 39 – Sensitivity analysis for different values of uncertainty.

The graphs are plotted with the two outputs separately, since this sort of sensitivity analysis is made with respect to a single output at a time. Another remark is that the Sobol method stabilizes after a certain number of evaluations. It is not necessary to reach an asymptote, but to observe the order of magnitude of the inputs and also, the guarantee that they will keep their ranks.

From the results, we can observe that, in this case study, when we increase the uncertainty on the inputs, there is a shift in their rank for the specific output \dot{Q}_{Aux} , as the collector slope becomes more critical. For the \dot{Q}_{Coll} output we keep the same rank, however,

we can observe that when increasing the uncertainty, the order of magnitude of the sensitivity of the two inputs approach to each other. We can conclude from these, that the notion of uncertainty is important in a project, and the decisions made can vary according to the results of the simulation.

As stated before, the Sobol method for global sensitivity has a drawback when dealing with complex black-box models. For instance, for this case study, for 1000 evaluations, the sensitivity analysis took two and a half hours, and over eight hours for 2500 evaluations.

As a base for comparison, the sensitivity analysis using the same method, for the white-box model Camelback/Rosenbrock test functions (the one used as toy example in subsection 3.5), is computed in less than two seconds, considering the same number of inputs, outputs and 2500 evaluations. The use of meta-modeling has been considered, but, since the Kriging surfaces are generated by the third-party OpenTURNS [60] platform, few adjustments are needed in order to make it work.

6.2 Case study *Les Roches Blanches* - EnergyPlus

As stated before, the case study *Les Roches Blanches* is a case way more complex than the case *Begin*. To start, it contains a larger number of modeling parameters. Besides, instead of two inputs, this case study has six, as previously presented on Table 5: **efit** corresponds to the external floor insulation thickness, **ewit** the external wall insulation thickness, **ifct** the inter-floor concrete thickness, **trit** the terrace roof insulation thickness, **uw** the thermal performance of windows, and **naxis** the orientation of the building referred to the geographic north. For the outputs, we have the total price of the chosen envelope materials and the district energy consumed for a certain time period.

6.2.1 Influence of inputs in calculation time

A first analysis was made regarding the calculation time according to the number of inputs. Each instance co-simulates the model according to the given inputs, and since they are combined to form the LHS (each input corresponds to one of the hyper-planes of the Hypercube), it is interesting to observe this behavior. Table 8 presents the results for the optimizations done for this case.

Table 8 – Elapsed time with respect to number of inputs for the case study *Les Roches Blanches*.

LHS	Number of individuals		Kriging Estimate Error	Elapsed Time [hours]	Bounded Inputs
	Global Fitting	Pareto Improvement			
10	10	10	0.010	1:30	efit, ewit
10	10	10	0.011	1:35	efit, ewit, ifct
10	10	10	0.010	1:41	efit, ewit, ifct, trit
10	10	10	0.033	1:30	efit, ewit, ifct, trit, uw
10	10	10	0.011	1:30	efit, ewit, ifct, trit, uw, naxis

From those results, it is possible to first conclude that increasing the number of inputs will not impact the calculation time. As the intent is to have an optimization with all the related inputs, this first observation makes clear that the best choice is having all the inputs for a complete problem optimizations. So, from this point on, all the result presented have a full rank of inputs.

6.2.2 Reference Pareto front

As previously discussed, an exhaustive data collection from the case study *Les Roches Blanches* would take more time than available for the project. However, to elaborate a reference Pareto front, the NSGAI algorithm was used by setting an acceptable error of 10^{-9} between generations. A total of 1321 iterations were necessary, taking around 72 hours. As a

remark, the larger number of samples or individuals, the larger the computational time, so this reference is a trade off between computational time and required precision.

6.2.3 Computation time analysis

Once the reference surfaces for the outputs and the reference Pareto front could be calculated, we can proceed with the validation of the proposed Adaptive DoE algorithm. For that, seven different experiments were included in this rapport, for $N = 15, 30, 45, 60, 75, 90$ and 105 individuals evenly distributed on each phase (for instance, for $N = 15$, we have 5 individuals for the LHS phase, 5 individuals for the Global Fitting phase and 5 individuals for the Pareto Improvement phase).

The first analysis done is related to the computational time and it is depicted in Figure 40. In this case we can observe in Figure 40(a) that the time grows almost linearly according

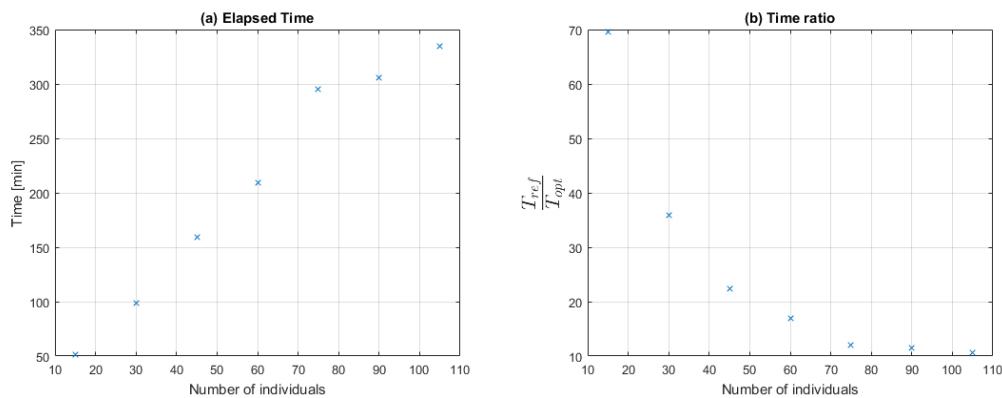


Figure 40 – (a) Computation time versus the number of individuals; (b) Time ratio versus the number of individuals.

to the number of individuals. The difference between responses if compared to case Begin can be given because of the complexity of the problem, since more time is spent in the time consuming model (co-simulation). Figure 40(b) gives us the ration between the time took for calculating the reference and the computation time for the optimization. We can observe that for the case with 15 individuals, we arrive to a result 70 times faster, and in the case of 105 individuals, 10 times faster, showing a great improvement. Besides, we can observe that for a large number of individuals, increasing it will not impact the ratio, since, according to the graph, we have an asymptotic behavior.

6.2.4 The coefficient of determination and the RMSE

Same as for the case Begin, we have used those coefficients to verify the improvement of the response with respect to an increase on the number of individuals.

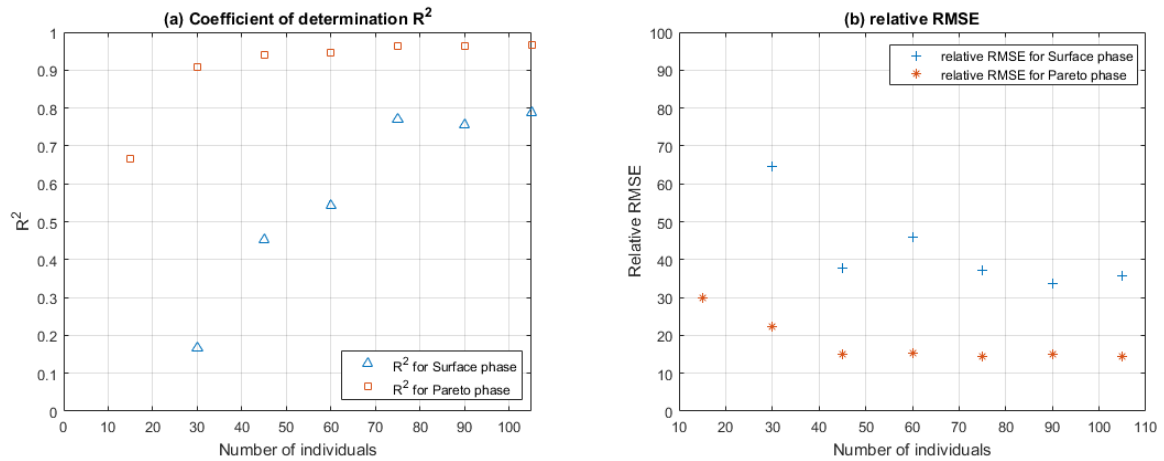


Figure 41 – Case *Les Roches Blanches*: (a) R^2 coefficient; (b) relative RMSE.

As we can observe from Figure 41, the increase of the number of individuals increases the value of R^2 and decreases the relative RMSE value. However, we can see that the slope in both cases is also diminished whereas we increase the number of individuals.

Another remark here is the achieved precision if using R^2 if compared to the simpler case Begin. We can notice that on the simpler case the coefficient R^2 reaches a value really close to 1 (0.9997) once the slope stabilizes. However, this value is around 0.965 for the complex case, showing that for this sort of cases, it is harder to obtain a better proportion of the variance in the dependent variable that is predictable from the independent variable.

6.2.5 Pareto evolution for different number of individuals

Another valid analysis is made from the observation of the Pareto front evolution according to the increase of the number of individuals. Figure 42 presents for each of the three phases, the evolution of the Pareto front according to the number of individuals. To enhance visibility and understanding, the same number of individuals from the previous sections was used in order to display the results.

We can observe throughout the experiments that the Pareto front from the optimization approaches the reference Pareto front when increasing the number of individuals. Although this is desirable, it is important to remember the trade off between the precision and the computation time of the optimization.

Another remark is regarding the evolution of the Pareto fronts. It is a lot more visible its improvement for a higher number of individuals.

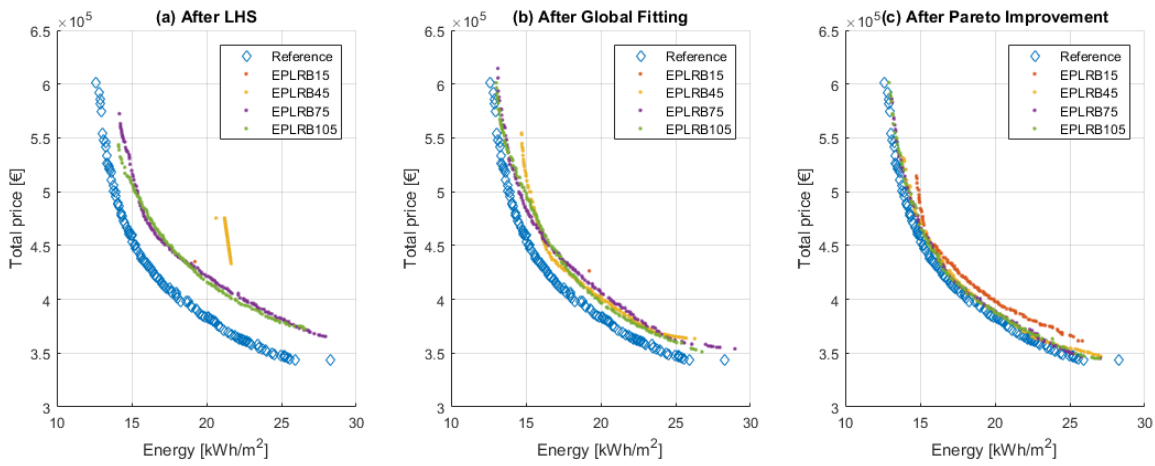


Figure 42 – The Pareto front evolution in the three phases of the algorithm for different number of individuals for case *Les Roches Blanches*.

6.2.6 Pareto front evolution for Les Roches Blanches with 105 individuals

Finally, for the instance with 105 individuals, we can observe the Pareto front evolution on each phase.

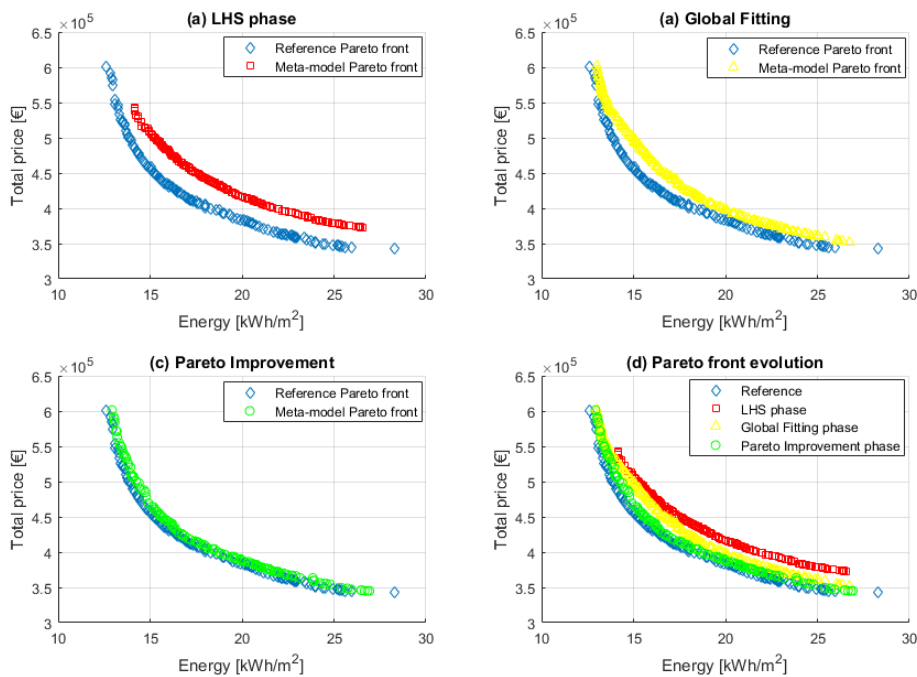


Figure 43 – Pareto front evolution for case *Les Roches Blanches* with 105 individuals.

In this case study we can clearly observe the approximation of the Pareto front in each phase of the algorithm (contrary to case *Begin*). We can also notice that the Pareto fronts are not purely convex, reflecting the complexity of the problem that can be attributed to an increased number of decision parameters.

6.2.7 Decision making

Having the same consideration of weights as for the case Begin, as we also have two objective functions for this case, the District Energy and Total price, we have a split weight of 0.5 for each criteria as well. Table 9 presents the results for different methods of MCDM in order to determine the best alternative among the 138 ones from the Pareto front.

Table 9 – Rank of 5 best alternatives for different MCDM methods for case *Les Roches Blanches* for N = 105 individuals.

Solution Rank	Method		
	WSM	WPM	TOPSIS
1	134	44	129
2	138	136	35
3	30	8	137
4	120	72	109
5	91	29	50

We can again remark that each of the methods used have a different ranking of best alternatives. However, as the Pareto front formation is not sorted, to have a better representation of the result, Figure 44 depicts the location of the results from Table 9 on the Pareto front for the respective case.

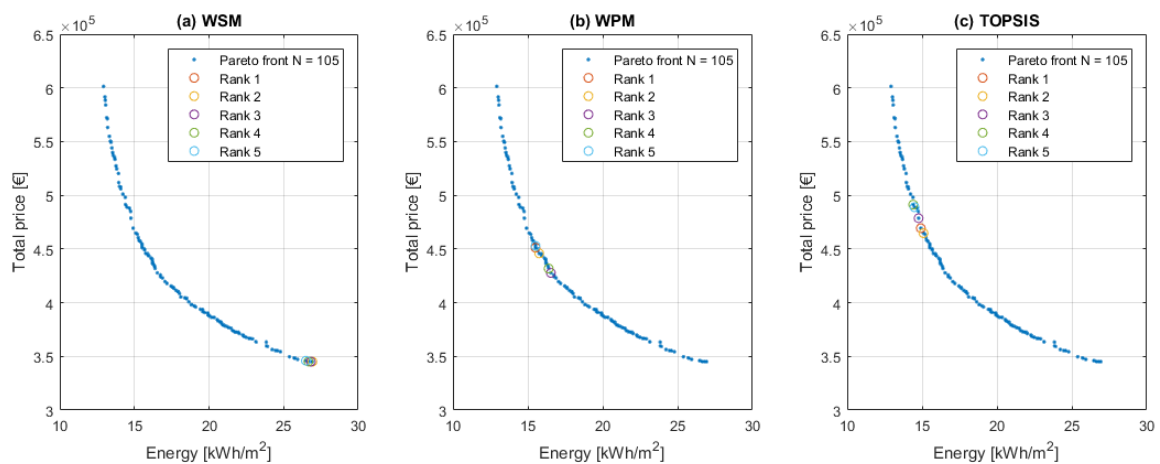


Figure 44 – Rank of 5 best alternatives for different MCDM methods for case *Les Roches Blanches* for N = 105 individuals.

We can make the same observation regarding the validity of the results from the WSM. Besides, we can again observe that WPM and TOPSIS methods bring closer results than the ones presented by WSM, and the decision on the method is up to the decision maker.

6.2.8 Sensitivity analysis

As stated before, Sobol method is costly for black-box complex models. After few attempts, it was decided that it is impractical to make such analysis for this case study, since a

considerable number of evaluations is necessary and the computation is pricey.

Few propositions were presented, such as making the sensitivity analysis using the meta-model instead of the real model, since it has been already validated that the meta-model have a good representation of the original model. However, the Kriging meta-modeling is made using OpenTURNS [\[60\]](#) library, and a software development has to be done in order to gather the meta-model information, task not done up to the moment.

7 Conclusions and Future developments

Several aspects of project INTENSE have been explored during the internship. For instance, its methodology, the application of its concepts, the proposed methodology of the adaptive DoE for the cases study and the interchangeability between CADES and other software.

Concerning the validation, for all the cases study presented here, as well as other cases that have been used as test functions (i.e. test function from tutorial), the results obtained are satisfactory. An extensive analysis had been done in order to observe the evolution of points for each phase of the algorithm and its applicability could be validated. The final phase of the algorithm delivers a result close to the reference if the right number of individuals is selected. We can observe an improvement of the response when increasing the number of individuals. The computation time required to obtain an acceptable answer have been reduced by at least ten times by the use of metamodeling.

A point to remark is that in real life application there is no reference to compare and validate the results. A post-processing module had been put in place in CADES in order to randomly pick-up an user determined number of points in order to validate a given Pareto front according to the user set up for the optimization. But, as we could notice on the cases study is that once the Pareto front forms a convex form at the end of the optimization process we have a slight decrease of the RMSE (or slight increase of R^2 coefficient. It is though, up to the end-user, to determine the trade off between the required precision and computation time.

As future developments, we can list the following tasks:

- The validation for a more complex case on TRNSys for the case *Les Roches Blanches*, as the partner finishes the modeling and shares it.
- An analysis on computation time using parallel processing.
- A robust analysis for the black box models.
- A sensitivity analysis and tutorial for CADES sensitivity analysis tool.
- Add a third output axis to include the Life Cycle Analysis (LCA) as a new objective of the optimization.

Finally, we can say that CADES is a tool with many important capabilities, and it can be used in several fields (not only on building optimization), but in several areas of engineering.

Bibliography

- 1 INTENSE. <http://www.agence-nationale-recherche.fr/Projet-ANR-13-VBDU-0003>: [s.n.]. Cited on page 19.
- 2 FRAISSE, G. *INTégration ENergétique des Systèmes et de l'Enveloppe des bâtiments : développement d'une méthodologie et d'un outil de conception optimisant la performance globale*. Projet ANR INTENSE - Programme Villes et Bâtiments Durables. Cited on page 21.
- 3 PLUME. <https://www.projet-plume.org/en>: [s.n.]. Cited on page 24.
- 4 LOCIE. <http://int.polytech.univ-smb.fr>: [s.n.]. Cited on page 24.
- 5 LGCB. <http://www.entpe.fr>: [s.n.]. Cited on page 24.
- 6 ARMINES. <https://www.armines.net/fr>: [s.n.]. Cited on page 24.
- 7 G2ELAB. <http://www.g2elab.grenoble-inp.fr>: [s.n.]. Cited on page 24.
- 8 CEA. <http://www.cea.fr/>: [s.n.]. Cited on page 24.
- 9 ENERGIE, A. <http://www.albedo-energie.eu>: [s.n.]. Cited on page 25.
- 10 GOLDBERG, D. et al. *Algorithmes génétiques : Exploration, optimisation et apprentissage automatique*. [S.l.]: Addison-Wesley, 1996. Cited on page 25.
- 11 KRISHNAKUMAR, K. Micro-genetic algorithms for stationary and non-stationary function optimization. In: RODRIGUEZ, G. (Ed.). *Intelligent Control and Adaptive Systems*. SPIE, 1990. Disponible em: <https://doi.org/10.1117/12.969927>. Cited 2 times on page(s) 25 and 32.
- 12 DIAS, T.; MILANEZ, L. F. Optimal location of heat sources on a vertical wall with natural convection through genetic algorithms. *International Journal of Heat and Mass Transfer*, Elsevier BV, v. 49, n. 13-14, p. 2090–2096, jul 2006. Disponible em: <https://doi.org/10.1016/j.ijheatmasstransfer.2005.11.031>. Cited on page 25.
- 13 ZITZLER, E. et al. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Transactions on Evolutionary Computation*, Institute of Electrical and Electronics Engineers (IEEE), v. 7, n. 2, p. 117–132, apr 2003. Disponible em: <https://doi.org/10.1109/tevc.2003.810758>. Cited on page 26.
- 14 HAMMACHE, A.; BENALI, M.; AUBÉ, F. Multi-objective self-adaptive algorithm for highly constrained problems: Novel method and applications. *Applied Energy*, Elsevier BV, v. 87, n. 8, p. 2467–2478, aug 2010. Disponible em: <https://doi.org/10.1016/j.apenergy.2009.11.026>. Cited on page 26.
- 15 CHANTRELLE, F. P. et al. Development of a multicriteria tool for optimizing the renovation of buildings. *Applied Energy*, Elsevier BV, v. 88, n. 4, p. 1386–1394, apr 2011. Disponible em: <https://doi.org/10.1016/j.apenergy.2010.10.002>. Cited 4 times on page(s) 26, 28, 40, and 42.

- 16 ACHARD, G. F. C. V. et G. Application des algorithmes génétiques au paramétrage d'un modèle simplifié du système bâtiment-chauffage. p. 6, 1999. Cited 2 times on page(s) 26 and 27.
- 17 FRAISSE, G. et al. Identification of equivalent thermal rc network models based on step response and genetic algorithms. 01 2011. Cited on page 26.
- 18 SANTIAGO, A. et al. A survey of decomposition methods for multi-objective optimization. In: *Studies in Computational Intelligence*. Springer International Publishing, 2014. p. 453–465. Disponible em: <https://doi.org/10.1007/978-3-319-05170-3_31>. Cited on page 26.
- 19 CASTILLO, O. et al. Méthodologie de coordination technico-économique-environnementale par optimisation multi-objectif. application à la cogénération de la mine de cuivre quebrada blanca. p. 7, 2003. Cited 2 times on page(s) 26 and 41.
- 20 KLEIJNEN, J. P. Kriging metamodeling in simulation: A review. *European Journal of Operational Research*, Elsevier BV, v. 192, n. 3, p. 707–716, feb 2009. Disponible em: <<https://doi.org/10.1016/j.ejor.2007.10.013>>. Cited on page 26.
- 21 GENGEMBRE, E. *Contribution à la construction de méta-modèles pour l'optimisation sous contraintes de critères énergétiques : application à la conception des bâtiments*. Tese (Doutorado) — Université de Toulouse, 12 2011. Cited on page 26.
- 22 FRAISSE, G. et al. Comparative study of various optimization criteria for SDHWS and a suggestion for a new global evaluation. *Solar Energy*, Elsevier BV, v. 83, n. 2, p. 232–245, feb 2009. Disponible em: <<https://doi.org/10.1016/j.solener.2008.07.021>>. Cited on page 27.
- 23 FRAISSE, G. et al. Intégration énergétique d'un système de refroidissement par sur-ventilation nocturne et double flux thermoélectrique. SFT, 2010. Cited on page 27.
- 24 CATALINA, T. *Estimation of residential building energy consumptions and analysis of renewable energy systems using a multi-criteria decision methodology*. Tese (Doutorado) — INSA de Lyon, 7 2009. Cited on page 27.
- 25 THIERS, S.; PEUPORTIER, B. Energy and environmental assessment of two high energy performance residential buildings. *Building and Environment*, Elsevier BV, v. 51, p. 276–284, may 2012. Disponible em: <<https://doi.org/10.1016/j.buildenv.2011.11.018>>. Cited on page 28.
- 26 PEUPORTIER, B. *Bancs d'essais de logiciels de simulation thermique*. 2005. Journée SFT-IBPSA « Outils de simulation thermo-aéraulique du bâtiment ». La Rochelle. Cited on page 28.
- 27 RIVALLAIN, M.; BAVEREL, O.; PEUPORTIER, B. *Genetic multi-criteria optimization for existing buildings holistic retrofit*. 2012. International Symposium Life Cycle Assessment and Construction. Nantes. Cited on page 29.
- 28 DELINCHANT, B. et al. An optimizer using the software component paradigm for the optimization of engineering systems. *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, Emerald, v. 26, n. 2, p. 368–379, apr 2007. Disponible em: <<https://doi.org/10.1108/03321640710727728>>. Cited on page 29.

- 29 WURTZ, F. et al. Dimensionnement optimal et simultané de l'enveloppe, des systèmes et de la stratégie de gestion en phase d'esquisse : application aux gares à énergie positive. In: *XXXe Rencontres AUGC-IBPSA*. Marseille, France: [s.n.], 2012. Disponible em: <<https://hal.archives-ouvertes.fr/hal-00717120>>. Cited on page 29.
- 30 KEIRSTEAD, J.; SAMSATLI, N.; SHAH, N. *Energy Efficient Cities: Assessment Tools and Benchmarking Practices*. [S.l.]: World Bank Publications, 2010. ISBN 0821381040. Cited on page 29.
- 31 WEBER, C.; SHAH, N. Optimisation based design of a district energy system for an eco-town in the united kingdom. *Energy*, Elsevier BV, v. 36, n. 2, p. 1292–1308, feb 2011. Disponible em: <<https://doi.org/10.1016/j.energy.2010.11.014>>. Cited on page 29.
- 32 HAZYUK, I.; GHIAUS, C.; PENHOUE, D. Optimal temperature control of intermittently heated buildings using model predictive control: Part i – building modeling. *Building and Environment*, Elsevier BV, v. 51, p. 379–387, may 2012. Disponible em: <<https://doi.org/10.1016/j.buildenv.2011.11.009>>. Cited on page 29.
- 33 STAINE, F.; FAVRAT, D. *Intégration Énergétique de procédés industriels par la méthode du pincement*. Tese (Doutorado) — Collection Ravel Industrie, 1994. Cited on page 30.
- 34 SUDHAKAR, T.; BALAJI, C.; VENKATESHAN, S. Optimal configuration of discrete heat sources in a vertical duct under conjugate mixed convection using artificial neural networks. *International Journal of Thermal Sciences*, Elsevier BV, v. 48, n. 5, p. 881–890, may 2009. Disponible em: <<https://doi.org/10.1016/j.ijthermalsci.2008.06.013>>. Cited on page 32.
- 35 BAÑOS, R. et al. Optimization methods applied to renewable and sustainable energy: A review. *Renewable and Sustainable Energy Reviews*, Elsevier BV, v. 15, n. 4, p. 1753–1766, may 2011. Disponible em: <<https://doi.org/10.1016/j.rser.2010.12.008>>. Cited on page 32.
- 36 MAGNIER, L.; HAGHIGHAT, F. Multiobjective optimization of building design using TRNSYS simulations, genetic algorithm, and artificial neural network. *Building and Environment*, Elsevier BV, v. 45, n. 3, p. 739–746, mar 2010. Disponible em: <<https://doi.org/10.1016/j.buildenv.2009.08.016>>. Cited on page 33.
- 37 NGUYEN, A.-T.; REITER, S.; RIGO, P. A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy*, Elsevier BV, v. 113, p. 1043–1058, jan 2014. Disponible em: <<https://doi.org/10.1016/j.apenergy.2013.08.061>>. Cited 2 times on page(s) 35 and 43.
- 38 MUSE. <http://muse-component.org/>: [s.n.]. Cited on page 35.
- 39 ENERGYPLUS. <https://energyplus.net/>: [s.n.]. Cited on page 38.
- 40 CAMPONOGARA, E. *Métodos de otimização: Teoria e prática*. 2006. Cited on page 39.
- 41 ARIAS-MONTANO, A.; COELLO, C. A. C.; MEZURA-MONTES, E. Multi-objective airfoil shape optimization using a multiple-surrogate approach. In: *2012 IEEE Congress on Evolutionary Computation*. IEEE, 2012. Disponible em: <<https://doi.org/10.1109/cec.2012.6256491>>. Cited on page 41.
- 42 TRIANTAPHYLLOU, E. *Multi-criteria Decision Making Methods: A Comparative Study*. 1. ed. [S.l.]: Springer US, 2000. (Applied Optimization 44). ISBN 978-1-4419-4838-0, 978-1-4757-3157-6. Cited 8 times on page(s) 43, 44, 45, 46, 48, 49, 50, and 90.

- 43 JACQUES, J. *Contributions to sensitivity analysis and generalized discriminant analysis*. Tese (Theses) — Université Joseph-Fourier - Grenoble I, 12 2005. Disponível em: <<https://tel.archives-ouvertes.fr/tel-00011169>>. Cited 4 times on page(s) 51, 52, 54, and 55.
- 44 SOBOL', I. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and Computers in Simulation*, Elsevier BV, v. 55, n. 1-3, p. 271–280, 02 2001. Disponível em: <[https://doi.org/10.1016/s0378-4754\(00\)00270-6](https://doi.org/10.1016/s0378-4754(00)00270-6)>. Cited 2 times on page(s) 53 and 54.
- 45 SIMPSON, T. et al. Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, Springer Nature, v. 17, n. 2, p. 129–150, jul 2001. Disponível em: <<https://doi.org/10.1007/pl00007198>>. Cited 5 times on page(s) 56, 58, 59, 61, and 62.
- 46 WANG, G. G.; SHAN, S. Review of metamodeling techniques in support of engineering design optimization. *Journal of Mechanical Design*, ASME International, v. 129, n. 4, p. 370, 2007. Disponível em: <<https://doi.org/10.1115/1.2429697>>. Cited on page 56.
- 47 HAN, Z.-H.; ZHANG, k.-s. Surrogate-based optimization. 03 2012. Disponível em: <https://www.researchgate.net/publication/221927230_Surrogate-Based_Optimization>. Cited on page 57.
- 48 UY, M.; TELFORD, J. K. Optimization by design of experiment techniques. In: *2009 IEEE Aerospace conference*. IEEE, 2009. Disponível em: <<https://doi.org/10.1109/aero.2009.4839625>>. Cited 2 times on page(s) 57 and 62.
- 49 JANSSEN, H. Monte-carlo based uncertainty analysis: Sampling efficiency and sampling convergence. *Reliability Engineering & System Safety*, Elsevier BV, v. 109, p. 123–132, jan 2013. Disponível em: <<https://doi.org/10.1016/j.ress.2012.08.003>>. Cited on page 59.
- 50 VIANA, F. A. C. A tutorial on latin hypercube design of experiments. *Quality and Reliability Engineering International*, Wiley-Blackwell, v. 32, n. 5, p. 1975–1985, nov 2015. Disponível em: <<https://doi.org/10.1002/qre.1924>>. Cited 2 times on page(s) 59 and 60.
- 51 SHEIKHOESLAMI, R.; RAZAVI, S. Progressive latin hypercube sampling: An efficient approach for robust sampling-based analysis of environmental models. *Environmental Modelling & Software*, Elsevier BV, v. 93, p. 109–126, jul 2017. Disponível em: <<https://doi.org/10.1016/j.envsoft.2017.03.010>>. Cited 2 times on page(s) 60 and 61.
- 52 VIANA, F. A. C.; VENTER, G.; BALABANOV, V. An algorithm for fast optimal latin hypercube design of experiments. *International Journal for Numerical Methods in Engineering*, Wiley-Blackwell, p. n/a–n/a, 2009. Disponível em: <<https://doi.org/10.1002/nme.2750>>. Cited on page 60.
- 53 HYNDMAN, R. J. et al. Nonparametric and semiparametric response surface methodology: a review of designs, models and optimization techniques. 2013. Disponível em: <<https://robjhyndman.com/papers/ISR-RSRSM.pdf>>. Cited 2 times on page(s) 61 and 62.
- 54 LEON-GARCIA, A. *Probability, Statistics, and Random Processes For Electrical Engineering (3rd Edition)*. Pearson, 2008. ISBN 0131471228. Disponível em: <<https://www.amazon.com/Probability-Statistics-Processes-Electrical-Engineering/dp/0131471228?SubscriptionId=0JYN1NVW651KCA56C102&tag=techkie-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0131471228>>. Cited on page 63.

- 55 HOPFE, C. et al. Robust multi-criteria design optimization in building design. 09 2012. Disponível em: <https://www.researchgate.net/publication/235412482_Robust_multi-criteria_design_optimization_in_building_design>. Cited on page 64.
- 56 GOOVAERTS, P. Geostatistics for natural resource evaluation. v. 42, 01 1997. Disponível em: <https://www.researchgate.net/publication/224839861_Geostatistics_for_Natural_Resource_Evaluation>. Cited on page 64.
- 57 BOHLING, G. *Kriging*. 2005. Kansas Geological Survey. Cited 2 times on page(s) 64 and 65.
- 58 FRAISSE, G. et al. Towards holistic building optimization using a computing environment that enable interoperability between numerical tools. 2018. ECOS 2018. Cited 3 times on page(s) 66, 70, and 79.
- 59 GARNIER, C.; SPITZ, C.; STAHAPOULOS, N. *Étude du cas de base des logements collectifs à Jacob Bellecombette - Simulation thermique dynamique - Hypothèse de base*. Projet ANR INTENSE. Cited on page 78.
- 60 OPENTURNS. <http://www.openturns.org/>: [s.n.]. Cited 2 times on page(s) 92 and 98.

APPENDIX A – CADES Basic Tutorial

The development of the tutorial is presented as a milestone to the project (L8). The contents of the tutorial can be found on [Dimocode](#) webpage for project INTENSE which contains the following elements:

Main properties

08 - Livrable 8 : Outil d'intégration énergétique

Description:

1° Accès à la version de CADES pour intense:

<https://www.dropbox.com/s/s5697v0o4kfg73g/cades-win64-4.0.14.beta.zip?dl=0>

2° Tutorial:

Tutorial pour CADES 4 (en version word et pdf)

[Tutorial Cades 4.0 v1.docx](#)

[Tutorial Cades 4.0 v1.pdf](#)

[Models for Tutorial.zip](#)

- presentation de CADES

Cades (Component Architecture for Design of Engineering Systems) s'agit d'un logiciel d'aide à la décision pour la conception optimale de systèmes et dispositifs et pour la construction de plan optimal de fonctionnement temporel de systèmes, en particulier énergétique. C'est un outil logiciel permettant de : développer un modèle orienté pour faire de l'optimisation (composant logiciel MUSE) ; analyser ce modèle via de simples calculs ou par une analyse de sensibilité locale ou globale ; faire une analyse des résultats d'optimisation.

CADES est un outil initialement développé au G2ELab par des doctorants (comme Petre Enciu, David Magot, Nyugen Hieu) et des chercheurs (Frédéric Wurtz, Laurent Gerbaud, Benoît Delinchant). Il est désormais valorisé et industrialisé par la société Vesta-System. C'est un formidable outil pour pré-dimensionner un système. Ce système peut-être modélisé par des modèles analytiques (en langages SML, C++, Java ou Python) ou semi-analytiques (Reluctool).

- Reprendre tutorial Ashkahn

[Tutorial Cades 4.0 utils complementaires.pdf](#)

3° Mettre les projets CADES 4 :

- Projet 1: Exemple analytique (3 fonctions test), ici juste le projet CADES 4

[Projet1.zip](#)

- Projet 2: Begin en TRNSys (TRNSYS 17. Version demo disponible sur <http://www.trnsys.com/demo/>)

[Projet2.zip](#)

le projet cades + répertoire et fichiers trnsys

- Projet 3: ZoneEvapCooler en E+ (EnergyPlus v. 8.8.0. Téléchargement disponible sur le [website](#))

[Projet3.zip](#)

A voir avec Alberto pour trouver un front de Pareto

- Projet 4: ZoneEvapCooler en E+ avec intégration avec SML

[Projet4.zip](#)

- Projet 5: Roches Blanches en E+

[Projet5.zip](#)

The [Tutorial](#) for CADES is presented as follows:

Tutorial CADES v4

Primary commands tutorial for CADES 4.0

Table des matières

CADES version 4.0.x.....	2
Description	2
Project 1 – Test functions.....	2
Unconstrained mono objective optimization.....	3
Ackley's function.....	3
CONSTRAINED mono objective optimization.....	5
Simionescu function	5
Constrained MULTI-objective optimization	7
CTP1 function	7
Interchangeability with TRNSys.....	8
Project 2 - Case Begin.....	8
Interchangeability with EnergyPlus.....	15
Project 3 - Case Zone Evap Cooler.....	16
The use of SML files along with TRNSys/EnergyPlus.....	21
Project 4 – EnergyPlus ZoneEvapCooler + SML File	21
Real case study	23
Project 5 - Case Les Roches Blanches.....	23
Sensitivity Analysis	25
CADES White-box sensitivity analysis available tools	27
CADES black-box sensitivity analysis available tools	31
Answers	33
Ackley's functions.....	33
Simionescu function	34
CTP1 function	34
Case Begin	34
CADES White-box sensitivity analysis available tools	35

CADES version 4.0.x

Description

CADES stands for Component Architecture for Design of Engineering Systems. It is a decision support software for the optimal design of systems and devices and for the construction of optimal plan of temporal operation of systems, in particular energetic. It consists of an user friendly environment software that allows the development an oriented model for optimization (with the aid of MUSE¹ software component). It is possible to analyse this model through simple calculations or local or global sensitivity analysis, as well as couple an optimization algorithm to the model in order to optimize it. Finally, it is possible to analyse those results.

CADES is a tool initially developed at G2ELab by PhD students (like Petre Enciu, David Magot, Nyugen Hieu) and researchers (Frédéric Wurtz, Laurent Gerbaud, Benoit Delinchant). It is now valued and industrialized by the company Vesta-System. It's a great tool to pre-dimensioning a system. This system can be modeled by analytical models, written in SML, C++, Java or Python languages or semi-analytical models (Reluctool).

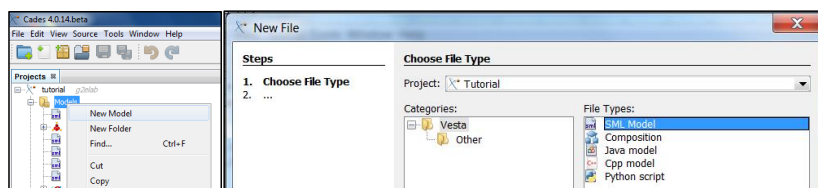
CADES (through MUSE component) also aims for the inteoperability between software, such as modeling software (EnergyPlus, TRNSys, Flux etc.), as well as other software, for instance, Matlab/Simulink.

Project 1 – Test functions

The first three cases have been provided by Benoît Delinchant² and are based on test functions for optimization found on:

https://en.wikipedia.org/wiki/Test_functions_for_optimization

1. Create a new project (File / New Project) called “tutorial”. Choose a location in your home.
2. In the Model folder, create a new Model (right click). Choose “SML” file type, call it “Ackley

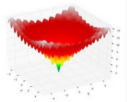


¹ <http://muse-component.org/>

² Benoit.Delinchant@G2ELab.grenoble-inp.fr

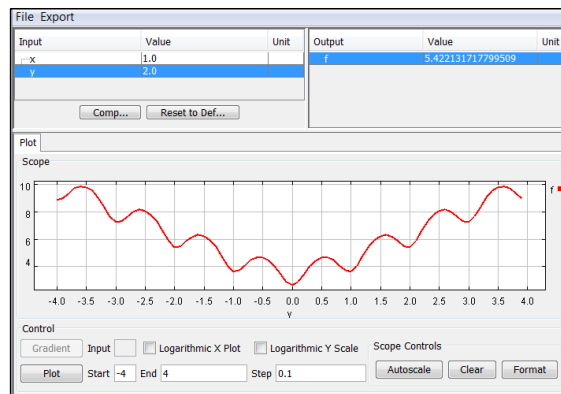
Unconstrained mono objective optimization

Ackley's function

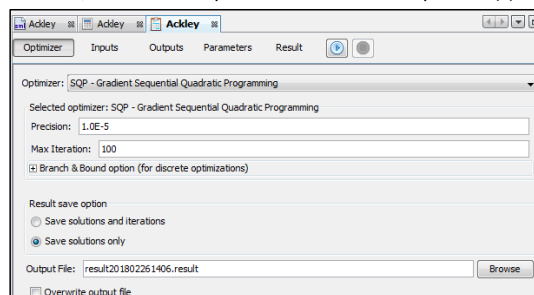
Plot	Formula	Global minimum	Search domain
	$f(x, y) = -20 \exp\left[-0.2 \sqrt{0.5(x^2 + y^2)}\right] - \exp[0.5(\cos(2\pi x) + \cos(2\pi y))] + e + 20$	$f(0, 0) = 0$	$-5 \leq x, y \leq 5$

- Write the following equation in the Ackley.sml file in the editor:

$$f = -20 * \exp(-0.2 * \sqrt{0.5 * (x * x + y * y)}) - \exp(0.5 * (\cos(2 * \pi * x) + \cos(2 * \pi * y))) + \exp(1) + 20;$$
- In the "View" menu, select "IDE Log".
- On the project browser, right click on the file and select "Generate Muse".
- If everything is good, right click on the file and select "Compute". Else call the professor.
- Make a simple computation and a plot in order to check computation:



- On the project browser, right click on the file and select "Optimize".
 - On "Optimizer" tab, choose SQP optimizer. Select "save optimum(s) and iterations".



Differently from the previous version, there is an option for Branch and Bound if dealing with discrete optimization. Besides, for this version, the output path for the file is the root file of the Model and it can be overwritten if box is selected.

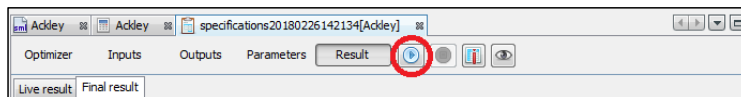
- b. On "Input" tab, for all variables, select "INTERVAL" for type of constraint, set the initial guess to 0, and min=-5 and max=5:

Variable name	Label	Unit	Type of constraint	Value/Initial guess	Min	Max	Description	Save value?
y			INTERVAL	0	-5.0	5.0	{type=INTE...	<input checked="" type="checkbox"/>
x			INTERVAL	0	-5.0	5.0	{type=INTE...	<input checked="" type="checkbox"/>

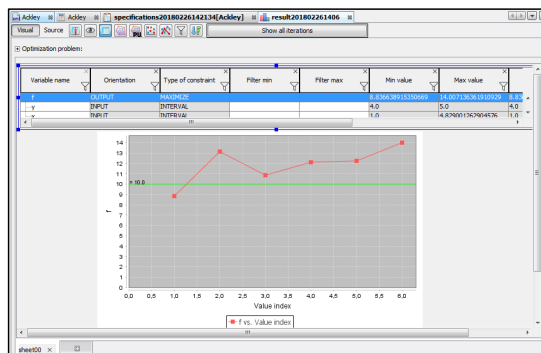
- c. On "Output" tab, for "f" variable, select "Maximize" Type. Set the initial value to 20 and leave the weight = 1.0:

Variable name	Label	Unit	Type of constraint	Value/Initial guess	Min	Max	Weight	Description	Save value?
f			MAXIMIZE	20			1.0	{type=MA...	<input checked="" type="checkbox"/>

- d. In "Optimization" menu, select "start", and see "Result" tab:



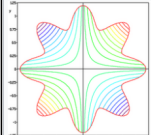
- e. Change the initial value of x and y and restart optimization. You can change the initial value several times. Conclusions?
- f. Increase the SQP optimizer "Precision" (from 1e-5 to 1e-10) and start again. Conclusions?
- g. What SQP means?
7. On the project browser, in "Solutions" folder, double click in your result file. New options for plotting and analyzing data have been added to this version. Please refer to "Tutoriel d'utilisation des outils complémentaires de la version 4.0.2 de Cades" for further details.



8. On the optimizer tab, test other algorithms and make some general conclusions.

CONSTRAINED mono objective optimization

Simionescu function

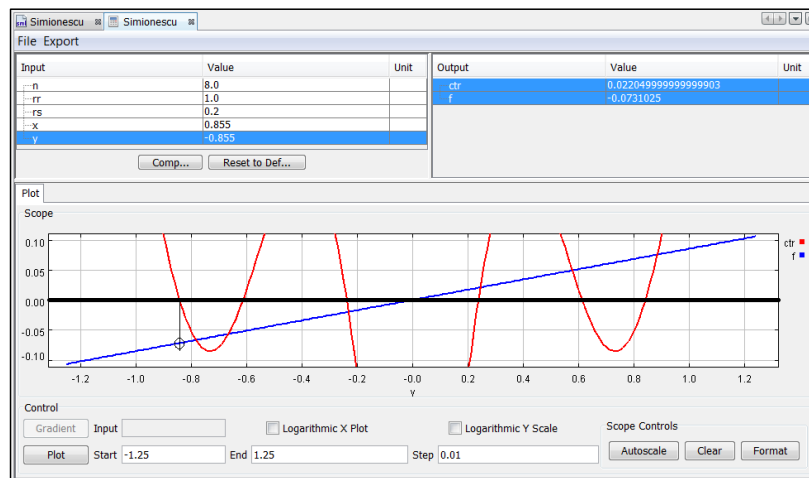
Plot	Formula	Global minimum	Search domain
	$f(x, y) = 0.1xy,$ subjected to: $x^2 + y^2 \leq \left[r_T + r_S \cos\left(n \arctan\left(\frac{x}{y} \right) \right) \right]^2$ where: $r_T = 1, r_S = 0.2$ and $n = 8$	$f(\pm 0.85586214, \mp 0.85586214) = -0.072625$	$-1.25 \leq x, y \leq 1.25$

- Write the following equations in the Simionescu.sml file in the editor:

```
f=0.1*x*y;
ctr=x*x+y*y-pow(rr+rs*cos(n*atan(x/y)),2);
```

Note that a constraint variable is defined in order to apply `ctr<0` in the optimizer. Note that the value of parameters `rr`, `rs` and `n`, is not defined in the model in order to avoid model regeneration if you want to change it.

- Make a simple computation and a plot in order to check computation. Check the optimal solution. You can save the input values in a file (*menu File / Save values*):



- Open an optimizer window. Select SQP optimizer. Select "save solutions and iterations".
- In this version it is not necessary to load the constraint, as it will appear directly on the output as it is defined as a function.
- The inputs `rr`, `rs` and `n` can be set to fixed and variables `x` and `y` to interval, respecting the given domain. Be aware that the constraint contains an $\text{atan}(x/y)$ element, so, setting initial values of $(x,y) = (0,0)$ will result in NaN:

Variable name	Label	Unit	Type of constraint	Value/Initial guess	Min	Max	Description	Save value?
r			FIXED	8			{type=FIXED,valu...	<input checked="" type="checkbox"/>
rr			FIXED	1			{type=FIXED,valu...	<input checked="" type="checkbox"/>
rs			FIXED	0.2			{type=FIXED,valu...	<input checked="" type="checkbox"/>
x			INTERVAL	2	-1.25	1.25	{type=INTERVAL,...	<input checked="" type="checkbox"/>
y			INTERVAL	1	-1.25	1.25	{type=INTERVAL,...	<input checked="" type="checkbox"/>

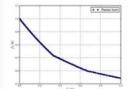
- In the Output tab, select `ctr` to "constraint" maximal value must be zero, and minimal value $-\infty$ (here put -100). f is set to be minimized, find appropriate bounds for normalization:

Variable name	Label	Unit	Type of constraint	Value/Initial guess	Min	Max	Weight	Description	Save value?
f			MINIMIZE	5			1.0	{type=MINIMIZ...	<input checked="" type="checkbox"/>
ctr			INTERVAL	5	-100.0	0.0		{type=INTERVA...	<input checked="" type="checkbox"/>

- Save your specifications in a file. Run the optimization. Check the value.
- Test different input values and algorithm, and check results.
- Draw some conclusion on the algorithm properties.

Constrained MULTI-objective optimization

CTP1 function

Plot	Functions	Constraints	Search domain
	$\text{Minimize} = \begin{cases} f_1(x, y) = x \\ f_2(x, y) = (1+y) \exp\left(-\frac{x}{1+y}\right) \end{cases}$	$\text{s.t.} = \begin{cases} g_1(x, y) = \frac{f_1(x, y)}{0.858 \exp(-0.541 f_1(x, y))} \geq 1 \\ g_2(x, y) = \frac{f_2(x, y)}{0.728 \exp(-0.295 f_1(x, y))} \geq 1 \end{cases}$	$0 \leq x, y \leq 1$

- Write the following equations in the CTP1.sml file in the editor:

```
f1=x;
f2=(1+y)*exp(-x/(1+y));
g1=1-f2/(0.858*exp(-0.541*f1));
g2=1-f2/(0.728*exp(-0.295*f1));
```

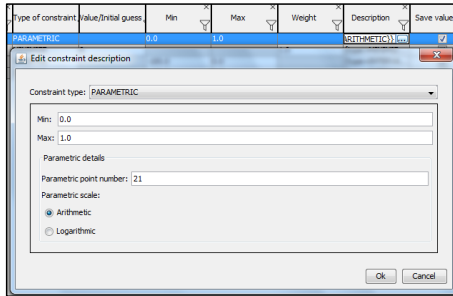
- In the Input tab, define x and y as "Constraint" and defined their range of variation to [0;1], as declared in the domain of the function. Then, before proceeding to the Pareto, make 2 mono-objective optimizations for each objective and note the corresponding value of the other objective (to do so, set one of the function constraint to FREE). It will give you the range of variation of the Pareto curve:

Variable name	Label	Unit	Type of constraint	value/Initial guess	Min	Max	Weight	Description	Save value?
f1			FREE					(type=FREE)	<input checked="" type="checkbox"/>
f2			MINIMIZE	1			1.0	(type=MINIMIZ...)	<input checked="" type="checkbox"/>
g1			INTERVAL		-1000.0	0.0		(type=INTERVA...)	<input checked="" type="checkbox"/>
g2			INTERVAL		-1000.0	0.0		(type=INTERVA...)	<input checked="" type="checkbox"/>

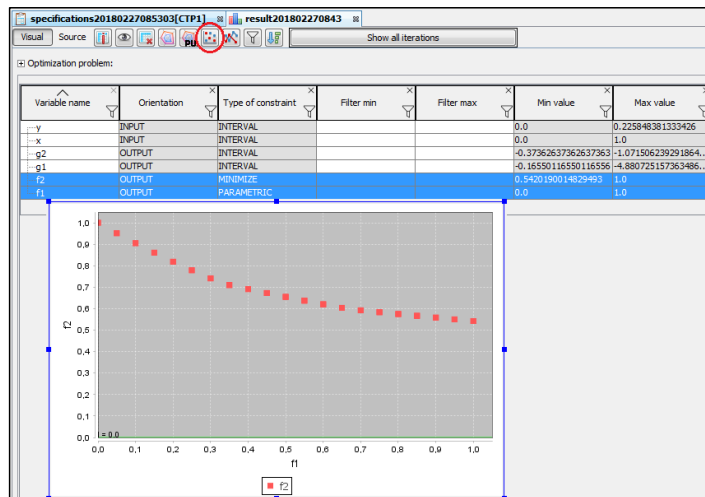
- To build the Pareto curve, set f_1 to parametric, f_2 to minimize, and constrain $g_1 < 0$ and $g_2 < 0$. This strategy is called "ε-constraint":

Variable name	Label	Unit	Type of constraint	value/Initial guess	Min	Max	Weight	Description	Save value?
f1			PARAMETRIC		0.0	1.0		(type=PARAME...)	<input checked="" type="checkbox"/>
f2			MINIMIZE	0			1.0	(type=MINIMIZ...)	<input checked="" type="checkbox"/>
g1			INTERVAL		-100.0	0.0		(type=INTERVA...)	<input checked="" type="checkbox"/>
g2			INTERVAL		-100.0	0.0		(type=INTERVA...)	<input checked="" type="checkbox"/>

Under f_1 , select the min and max according to the results from the previous step. Also for f_1 , under description click on the three dots on the right and type 21 for Parametric points number.



4. Run optimization. It must converge in 21 iterations. These iterations are not corresponding to the model call number. To compare with other, just note the optimization time. To see results go to the next step.
5. The results can be seen on the result file you have selected. To plot the Pareto curve, click on *Scatter Plot*, then add f_2 (y data for plotting), f_1 (x data for plotting) and hit Ok button.

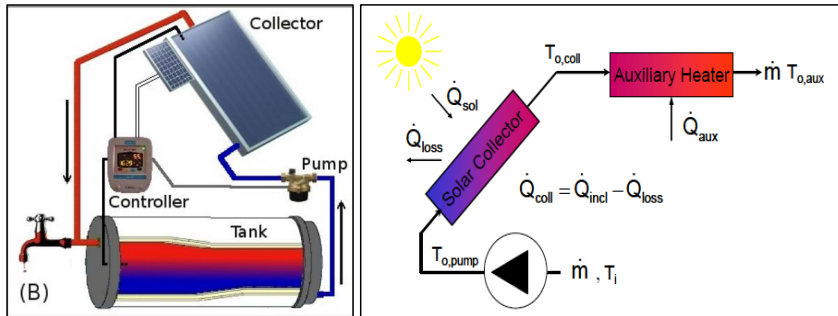


6. By plotting the constraints, explain the 2 breaks in the Pareto.
7. Compare this strategy with NSGAI Algorithm.

Interchangeability with TRNSys

Project 2 - Case Begin

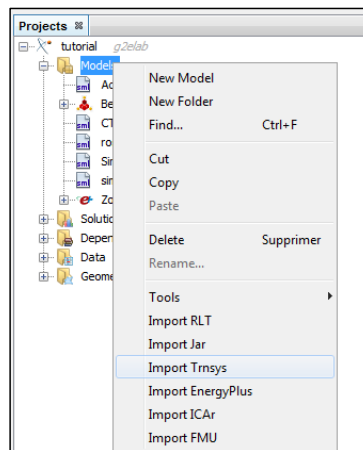
Case of study Begin is a simple TRNSys example found directly on its help contents.



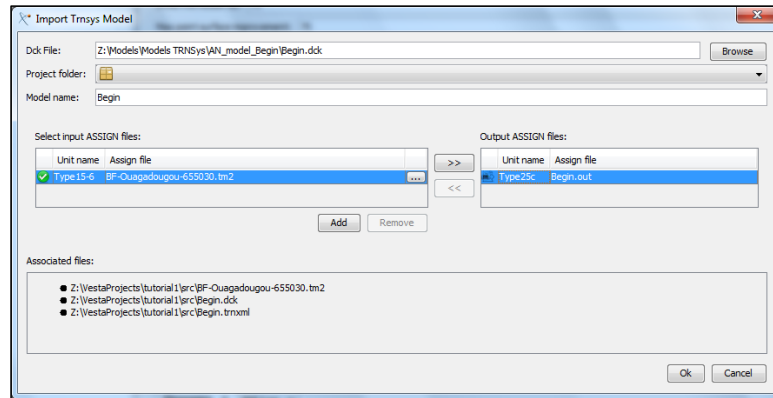
It comprehends a system for heating water containing a tank and a solar panel. With the aid of TRNSys we can model and simulate its behavior as a black-box system, meaning that we have no knowledge of the equations ruling the system, but instead, for given set inputs we get a corresponding set of outputs.

In this part of the tutorial we will see how to integrate a TRNSys model into CADES, how to edit the files in order to determine the inputs/outputs of the system, how to generate the MUSE, compute some values, and run an optimization and observe the results.

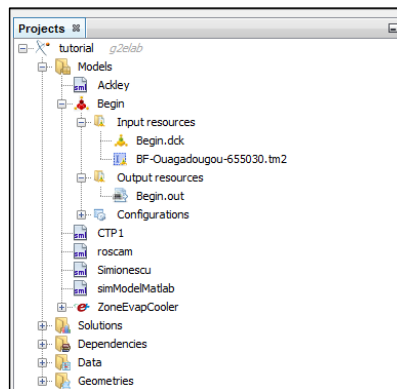
1. Import model: As we already have a model that has been generated by TRNSys software, instead of creating a new model, we will import the generated files. In the model folder, right click and select Import Trnsys:



For this example, we will need 3 files: a .dck file that corresponds to the TRNSys model, a .tm2 file, that corresponds to a meteorological data file, and, a .out file, an output file:

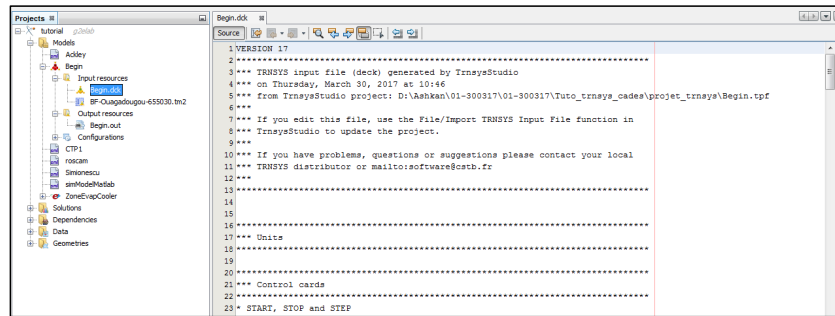


Once all are selected, just click on the Ok button. We are able to see the imported files on the project tree:

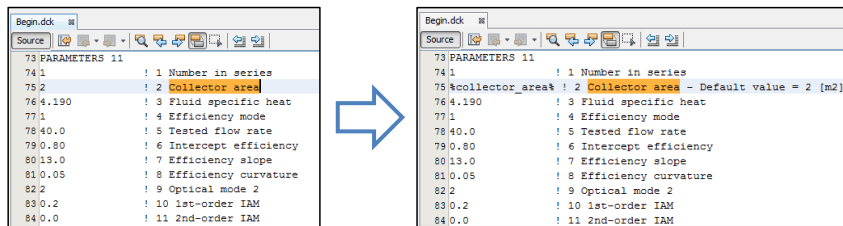


2. Edit inputs/outputs

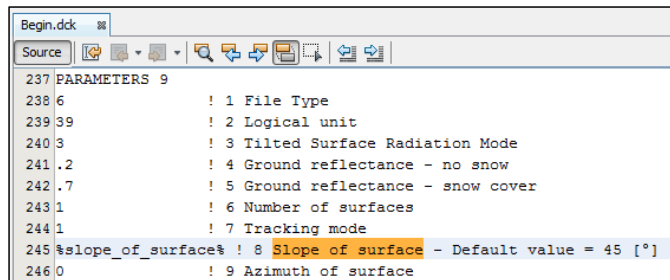
- a. Now we have to determine the inputs and outputs variables in our file, since the model only supplies the parameters. To do so, we can double click (or right click and select open) the Begin.dck file.



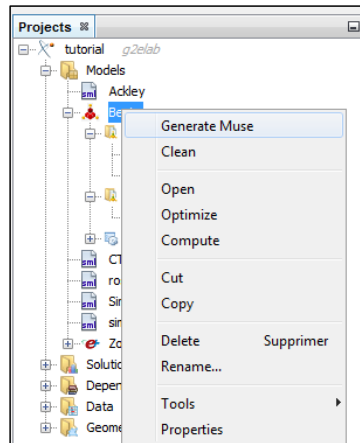
- b. We can now find the desired parameter that we want to make as a variable. We recommend the collector surface area and the collector slope angle. We can click on the right side of the screen that contains the text file and under the menu, go to edit and find (or simply CTRL + F) and search for collector area.



- c. The inputs are determined by creating a variable between the % symbol. In the example above, we have created a variable called collector_area (declared as %collector_area%) by replacing the constant value of the parameter. We can do the same for the Slope of surface:



- d. As for the outputs, we will leave the ones already set during modeling. They can be identified by the "printed" key word next to them.
3. Generate MUSE: in order to proceed with computation or optimization it is necessary to generate the MUSE component. First, make sure the last version of the file is saved and then, right click on the model parent name and select "Generate Muse".



4. Compute: similar to SML models, it is possible to compute output values from the declared inputs by using the Compute tool. To do so, right click on the parent model name and select compute. By filling the default input values, we obtain the following results:

Input	Value	Unit	Output	Value	Unit
collector_area	2.0		Type25c	...	
slope_of_surface	45.0		Fsol	0.48085217490490373	
Parameters	...		QAux	1701300.2878816193	
timeout	-1.0		QColl	1579801.5425468525	
trmsys_exe	C:/trmsys17/Exe/TRMExe.exe				

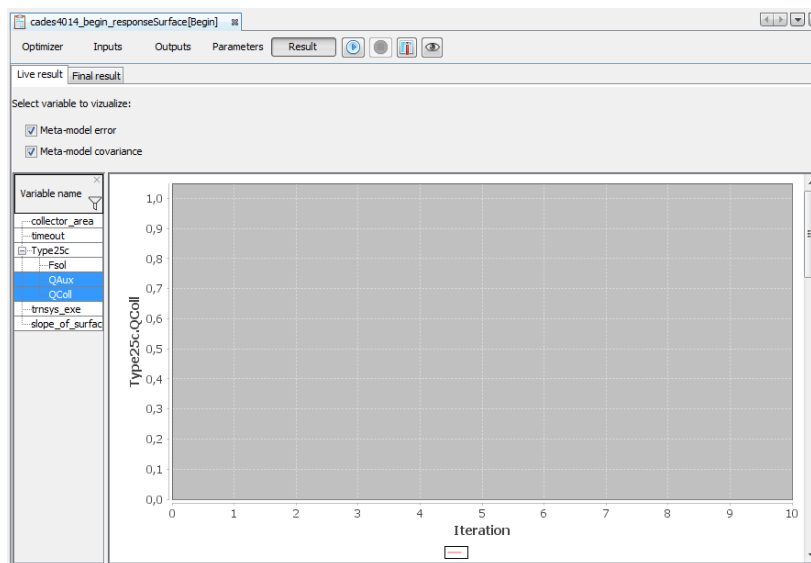
5. Optimization
 - a. Right click on on the Begin model and select "Optimize".
 - b. For the inputs, select Type of constraint as INTERVAL and collector_area in the range [1,10] and Initial guess 2; for slope_of_surface the range [0,90] and initial guess 45:

Variable name	Type of constraint	Value/Initial guess	Min	Max	Description	Save value?
collector_area	INTERVAL	2	1.0	10.0	(type=INTERVAL,valu...	<input checked="" type="checkbox"/>
slope_of_surface	INTERVAL	45	0.0	90.0	(type=INTERVAL,valu...	<input checked="" type="checkbox"/>

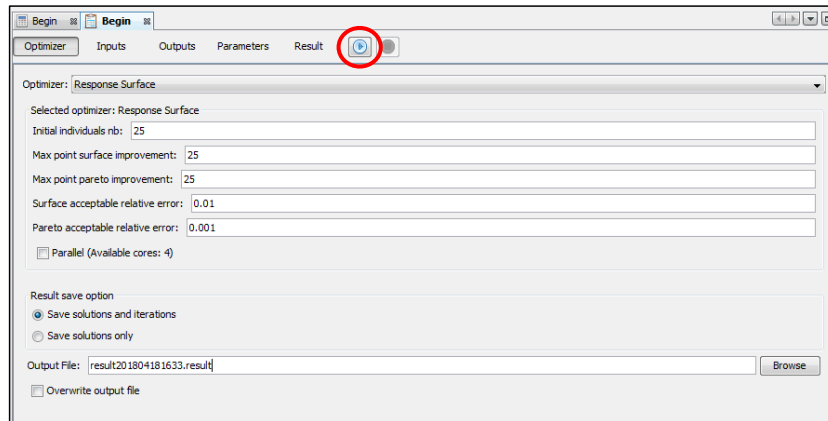
- c. For the outputs, we choose to MINIMIZE QAux and MAXIMIZE QColl:

Variable name	Type of constraint	Value/Initial guess	Min	Max	Weight	Description	Save value?
Type25c						0	<input checked="" type="checkbox"/>
Fsol	FREE					{type=FREE}	<input checked="" type="checkbox"/>
QAux	MINIMIZE	100			1.0	{type=MINIMIZE,V...	<input checked="" type="checkbox"/>
QCoil	MAXIMIZE	100			1.0	{type=MAXIMIZE,V...	<input checked="" type="checkbox"/>

- d. Under Result tab, on “Live Results” sub-tab, select both outputs and check Meta-model error and Meta-model covariance. This will allow us to see their evolution:

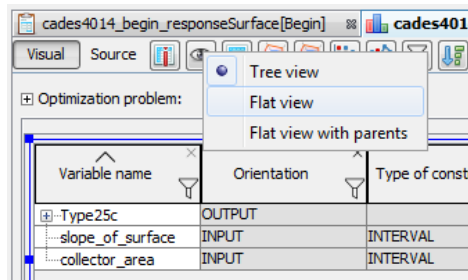


- e. Finally, on the Optimizer tab, we choose the Response Surface method for the optimizer, with 25 individuals for each of the three phases. Keep the relative errors values as they are, edit the Output File name if desirable, and click on Start Optimization and wait until done:

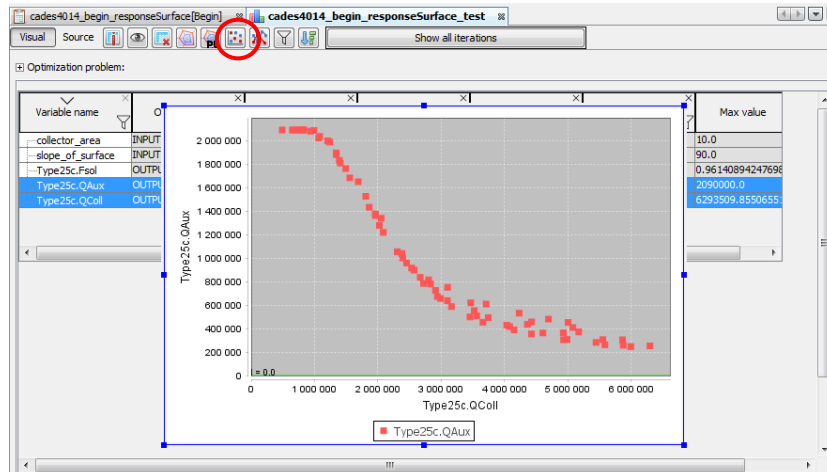


6. Results – front Pareto

- a. On the Solutions folder, double click on the solution file previously generated.
- b. Click on “Change view mode” and select either “Flat View” or “Flat view with parents”:



- c. Select both QAux and QColl and click on “Scatter Plot” to plot the Pareto front. Note, QAux needs to be on top of QColl to have the right x and y axis. To change the order, you can click on “Variable name” and it will sort the names accordingly. If the axis are not in the right selection, you can right click on the image and select “Revert axis”.



- Change the initial value of collector_area and slope_of_collector. You can change the initial value several times. Conclusions?
- Change the initial guess of QAux and QColl and restart optimization. You can change the initial value several times. Conclusions?
- Change the acceptable relative error for the Surface and the Pareto. Conclusions?
- Increase and/or decrease the number of individuals for each step (Initial, Surface improvement, Pareto improvement). What happens to the Pareto front? Any other conclusions?

Interchangeability with EnergyPlus

Before proceeding with EnergyPlus, in order to make the interoperability, we need to change the file RunEPlus.bat. We can find the respective file under the root directory of EnergyPlus installation.

Because it is a .bat file, if we double click on it, it will execute the file. So, in order to make changes, we need to right-click and select Modify. If you have Notepad++ installed, you can choose the option "Edit with Notepad++".

We need to add the following line after line 48 (set weather_path=%program_path%WeatherData\):

```
IF EXIST "%2.epw" set weather_path=
```

```
47 set post_proc=%program_path%PostProcess\  
48 set weather_path=%program_path%WeatherData\  
49 IF EXIST "%2.epw" set weather_path=  
50 set pausing=N  
51 set maxcol=250
```

That way, we can set a weather path for .epw files.

<http://www.vesta-system.fr/fr/produits/cades/vesta-cades.html>

Project 3 - Case Zone Evap Cooler

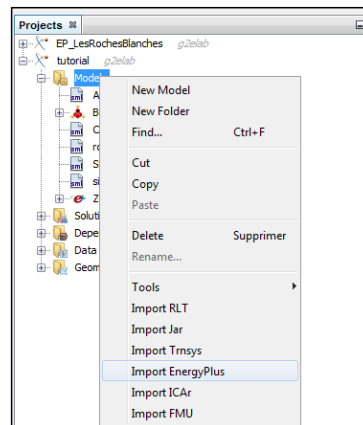
Case of study 1ZoneEvapCooler is a simple EnergyPlus example found directly on its examples contents.

The file models a small one-zone house with a whole-house direct evaporative cooler system. Because evaporative coolers cannot be controlled with the ZoneControl:Thermostat object, this file illustrates how an evaporative cooler system can be thermostatically controlled a crude manner using the AvailabilityManager objects. Further explanation regarding the modeling can be found directly in the 1ZoneEvapCooler.idf file.

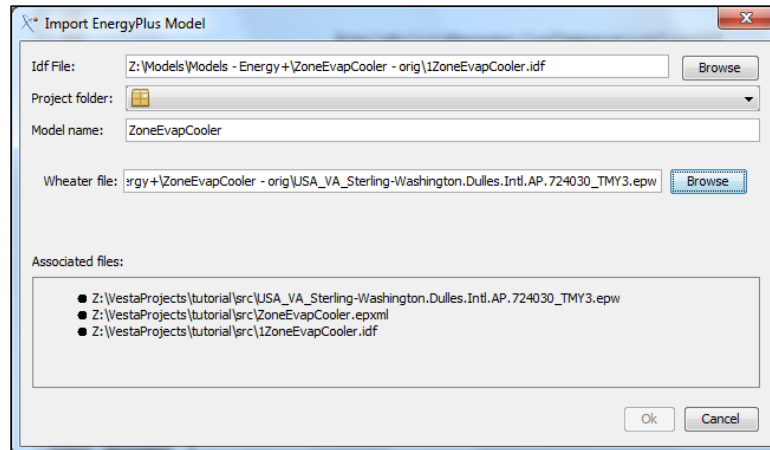
In this part of the tutorial we will see how to integrate an EnergyPlus model into CADES, how to edit the files in order to determine the inputs/outputs of the system, how to generate the MUSE, compute some values, and run an optimization and observe the results.

1. Import model

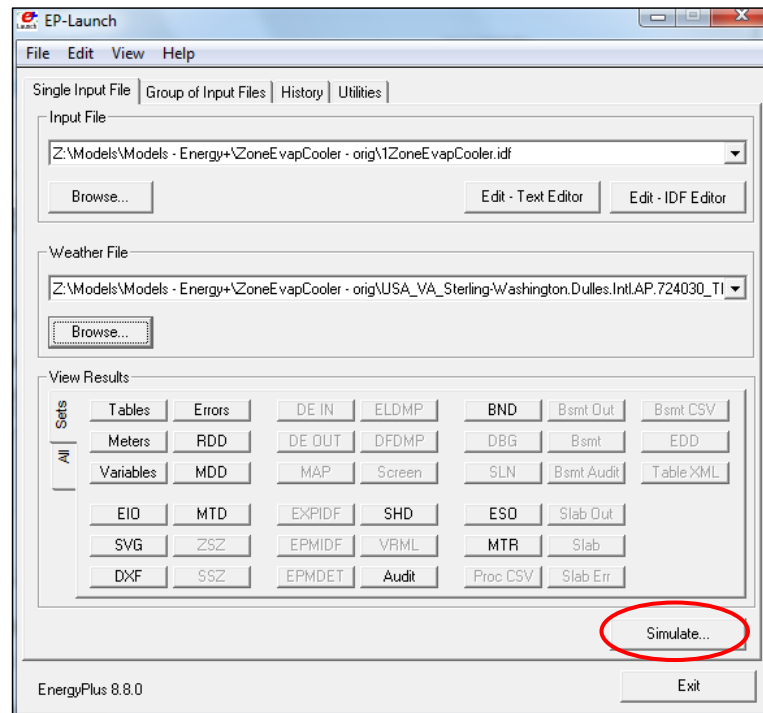
Similarly to TRNSys example, the model is already generated and we need to import it instead of creating it. With the mouse pointer on the Model folder, right click and select Import EnergyPlus:



For this example, we will need 2 files: a .idf file that corresponds to the EnergyPlus model, and a .epw file, that corresponds to a meteorological data file. Note that the model name on CADES can not begin with a number, meaning that we need to edit the name by removing the number in front of the suggested model name.

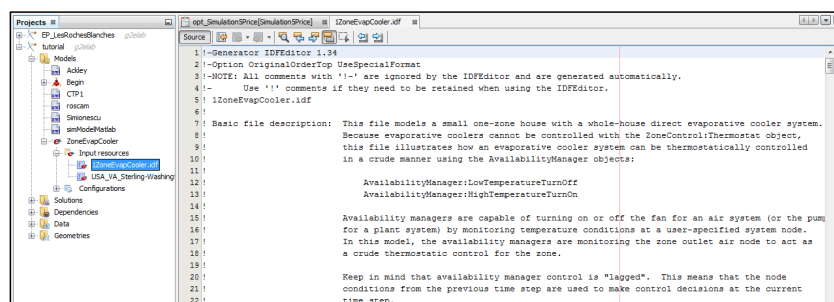


Differently from TRNSys however, the output is unknown a priori. Hence, it needs to be generated separately. In order to do that, we need to launch EP-Lanch.exe, present into the root directory of EnergyPlus installation. We can browse the .idf and .twp file and finally click on Simulate. This will generate a .tab file (1ZoneEvapCooler.tab) which contains the output for the simulation. We will check on this file later on to define the outputs on the .idf file.



2. Edit inputs/outputs

Now we have to determine the inputs and outputs variables in our file, since the model only supplies the parameters. To do so, we can double click (or right click and select open) the 1ZoneEvapCooler.idf file.



We can now find the desired parameter that we want to make as a variable. This goes exactly as for TRNSys. We can select the C5 - 4 IN HW CONCRETE thicknes and Water Pump Power:


```

195 Material,
196 CS - 4 IN HW CONCRETE,      !- Name
197 MediumRough,                !- Roughness
198 $Thickness$,                 !- Thickness - Default value = 0.1014984 {m}
199 1.729577,                   !- Conductivity {W/m-K}
200 2242.585,                   !- Density {kg/m3}
201 836.8000,                   !- Specific Heat {J/kg-K}
202 0.9000000,                  !- Thermal Absorptance
203 0.6500000,                  !- Solar Absorptance
204 0.6500000;                  !- Visible Absorptance

```

```

474 EvaporativeCooler:Direct:CeilDekPad,
475 Evaporative Cooler,        !- Name
476 System Availability Schedule, !- Availability Schedule Name
477 1.6,                        !- Direct Pad Area (m2)
478 0.5,                        !- Direct Pad Depth (m)
479 $pump_power$,              !- Recirculating Water Pump Power Consumption - Default Value = 225 {W}
480 Evap Cooler Inlet Node,    !- Air Inlet Node Name
481 Supply Outlet Node,       !- Air Outlet Node Name
482 CONSTANT;                  !- Control Type

```

As for the outputs, we can recognize them by the expression "Output:Meter". In our case, at the end of the .idf file, replace:

```

Output:Meter,Electricity:Facility,monthly;
Output:Meter,Electricity:HVAC,monthly;
Output:Meter,Electricity:Facility,runperiod;
Output:Meter,Electricity:HVAC,runperiod;

```

by:

```

Output:Meter,Heating:Electricity,monthly;
Output:Meter,Heating:Electricity,runperiod;
Output:Meter,Fans:Electricity,monthly;
Output:Meter,Fans:Electricity,runperiod;

```

However, if you want to add a different output, it can be done by going to the end of the file, and with the aid of the .tab file previously generated, finding the corresponding output sections. The figure below depicts the elements present in an output file in order to have as an output the Cooling Water for End Uses on the Annual Building Utility Performance Summary.

The .tab file:

47	End Uses	Electricity [GJ]	Natural Gas [GJ]	Additional Fuel [GJ]	District Cooling [GJ]	District Heating [GJ]	Water [m3]	
48								
49	Heating	36.88	0.00	0.00	0.00	0.00	0.00	
50	Cooling	0.41	0.00	0.00	0.00	0.00	0.00	
51	Interior Lighting	0.00	0.00	0.00	0.00	0.00	0.00	
52	Exterior Lighting	0.00	0.00	0.00	0.00	0.00	0.00	
53	Interior Equipment	0.00	0.00	0.00	0.00	0.00	0.00	
54	Exterior Equipment	0.00	0.00	0.00	0.00	0.00	0.00	
55	Fans	2.33	0.00	0.00	0.00	0.00	0.00	
56	Pumps	0.00	0.00	0.00	0.00	0.00	0.00	
57	Heat Rejection	0.00	0.00	0.00	0.00	0.00	0.00	
58	Humidification	0.00	0.00	0.00	0.00	0.00	0.00	
59	Heat Recovery	0.00	0.00	0.00	0.00	0.00	0.00	
60	Water Systems	0.00	0.00	0.00	0.00	0.00	0.00	
61	Refrigeration	0.00	0.00	0.00	0.00	0.00	0.00	
62	Generators	0.00	0.00	0.00	0.00	0.00	0.00	
63								
64								
65	Total End Uses	39.63	0.00	0.00	0.00	0.00	5.89	
66	Note: Electricity appears to be the principal heating source based on energy usage.							

The .idf file:

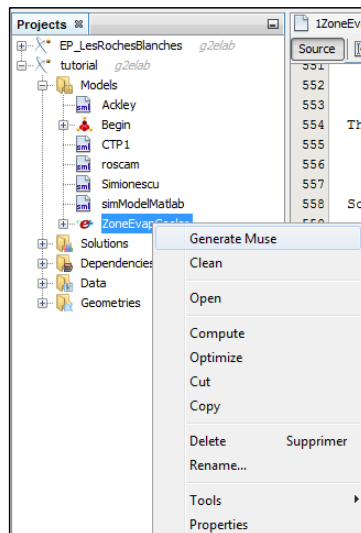
```

565 Output:VariableDictionary,Regular;
566
567 Output:Surfaces:Drawing,DXF;
568
569 Output:Constructions,Constructions;
570
571 OutputControl:Table:Style,
572   TabAndHTML;           !- Column Separator
573
574 Output:Table:SummaryReports,
575   AllSummary;           !- Report 1 Name
576   !- $Annual Building Utility Performance Summary.End Uses.Cooling.Water$ !- {m3}

```

3. Generate MUSE

In order to proceed with computation or optimization it is necessary to generate the MUSE component. First, make sure the last version of the file is saved and then, right click on the model parent name and select “Generate Muse”.



4. Compute

Similar to previous models, it is possible to compute output values from the declared inputs by using the Compute tool. To do so, right click on the parent model name and select compute. By filling the default input values, we obtain the following results:

Input	Value	Unit	Output	Value	Unit
pump_power	225.0	W	Fans	...	
thickness	0.1014984	m	Electricity	2.3347402040816197E9	
Parameters	...		Heating	...	
			Electricity	3.688491129718239E10	

Buttons: Compute, Reset to Default

5. Optimization: the same steps taken for the other examples apply to the optimization in this case. Instead of repeating the same procedures, we can move forward to the next section, that shows how to integrate an SML model to an EnergyPlus or TRNSys one. You are welcome to make some optimizations and try different algorithms if you wish so.

The use of SML files along with TRNSys/EnergyPlus

Let us consider now that we want make a cost function that is a composition of different outputs, or if we want to include the price related to a variable. We can achieve this with the aid of an SML file.

Project 4 - EnergyPlus ZoneEvapCooler + SML File

1. Create a new SML file (right click on model and select new model). Call it simZEC for instance.
2. Open simZEC and drag and drop ZoneEvapCooler into it. It will create a file like the one below with the GAV (GroupId, ArtifactId and Version) of the file, as well as the variables already declared:

```

1 import org.gu.g2elab:tutorial.ZoneEvapCooler:1.0;
2 //New model instance:
3 my_ZoneEvapCooler = new ZoneEvapCooler;
4
5 //Affect scalar inputs:
6 my_ZoneEvapCooler.pump_power = my_pump_power_value;
7 my_ZoneEvapCooler.thickness = my_thickness_value;
8
9 //Affect scalar parameters:
10 parameter my_energyPlus_exe_value = "my_energyPlus_exe_value";
11 parameter my_timeout_value;
12 my_ZoneEvapCooler.energyPlus_exe = my_energyPlus_exe_value;
13 my_ZoneEvapCooler.timeout = my_timeout_value;
14
15 //Get scalar outputs:
16 my_Fans_Electricity_value = my_ZoneEvapCooler.Fans.Electricity;
17 my_Heating_Electricity_value = my_ZoneEvapCooler.Heating.Electricity;
18
19

```

- a. For parameters, we need to change EnergyPlus path :

```

parameter my_energyPlus_exe_value = "C:/EnergyPlusV8-8-0/RunEPlus.bat";
parameter my_timeout_value = -1;

```

```

9 //Affect scalar parameters:
10 parameter my_energyPlus_exe_value = "C:/EnergyPlusV8-8-0/RunEPlus.bat";
11 parameter my_timeout_value = -1;
12 my_ZoneEvapCooler.energyPlus_exe = my_energyPlus_exe_value;
13 my_ZoneEvapCooler.timeout = my_timeout_value;

```

3. We can create labels and declare the variables units with the aid of keywords **label** and **unit**:

```

19 label my_pump_power_value = "Power of the water pump";
20 label my_thickness_value = "Thickness of the concrete wall";
21 unit my_pump_power_value = "W";
22 unit my_thickness_value = "m";

```

```
label my_pump_power_value = "Power of the water pump";
label my_thickness_value = "Thickness of the concrete wall";
unit my_pump_power_value = "W";
unit my_thickness_value = "m";
```

4. Finally, we can write the prices and the outputs:

```
24 //Prices
25 pump_Price = pow(my_pump_power_value,2/3);
26 wall_Price = (520.1*my_thickness_value + 73.602);
27
28 //Outputs to optimize
29 total_Energy = my_Fans_Electricity_value + my_Heating_Electricity_value;
30 total_Price = pump_Price + wall_Price;
```

```
//Prices
pump_Price = pow(my_pump_power_value,2/3);
wall_Price = (520.1*my_thickness_value + 73.602);

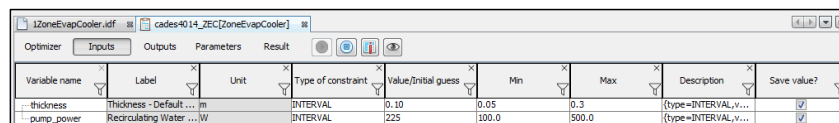
//Outputs to optimize
total_Energy = my_Fans_Electricity_value + my_Heating_Electricity_value;
total_Price = pump_Price + wall_Price;
```

5. We can then generate the MUSE.
- If not yet done, generate the MUSE for the EnergyPlus model ZoneEvapCooler.
 - After that, we can generate the MUSE for simZEC.
 - Disregard the errors regarding the generators, as the model isn't represented by algebraic equations). If, for some reason you get the following error:

```
14:32:56: [ERROR]: Unable to resolve dependencies for org.gu.g2elab:projet4.simZEC:1.0 - Could not find:
org.gu.g2elab:projet4.ZoneEvapCooler:1.0
```

please check the GAV for typos and fix the path for keyword **import** (you can check the values under model properties – right-click → properties).

6. Create a new optimization (right click on simZEC and select Optimization).
- For the inputs, select Type of constraint as INTERVAL and thickness in [0.05 0.3] with 0.1 as initial guess and pump_power in [100 500] with 225 as initial guess:

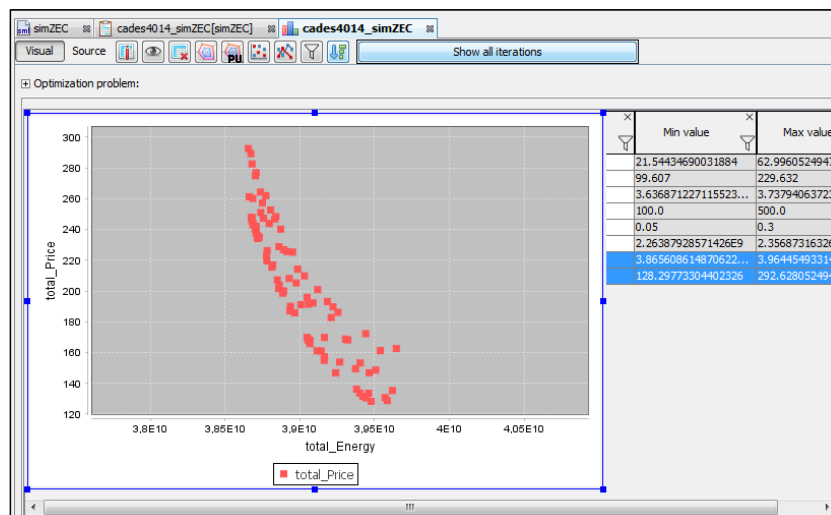


Variable name	Label	Unit	Type of constraint	Value/initial guess	Min	Max	Description	Save value?
thickness	Thickness - Default ...	m	INTERVAL	0.10	0.05	0.3	{type=INTERVAL,y...	<input checked="" type="checkbox"/>
pump_power	Recirculating Water ...	W	INTERVAL	225	100.0	500.0	{type=INTERVAL,y...	<input checked="" type="checkbox"/>

- For outputs we can minimize total_Price and minimize total_Energy:

Variable name	Label	Unit	Type of constraint	value/initial guess	Min	Max	Weight	Description	Save value?
total_Price			MINIMIZE	10			1.0	(type=MINIMIZE,...	<input checked="" type="checkbox"/>
total_Energy			MINIMIZE	10			1.0	(type=MINIMIZE,...	<input checked="" type="checkbox"/>
my_Heating_Elec			FREE					(type=FREE)	<input checked="" type="checkbox"/>
pump_Price			FREE					(type=FREE)	<input checked="" type="checkbox"/>
wall_Price			FREE					(type=FREE)	<input checked="" type="checkbox"/>
my_Fans_Electric			FREE					(type=FREE)	<input checked="" type="checkbox"/>

- c. For optimizer, set Response Surface as optimization method and set 15 individuals for each phase.
 - d. Click on play to start optimization.
7. To see the results, click on the solution file and select total_Price and total_Energy:



8. Can we draw the same conclusions by changing initial guess/values for inputs and outputs, and number of individuals for each phase, as the conclusions taken on the Begin case?

Real case study

Project 5 - Case Les Roches Blanches

The building studied has four levels. It has a living area of about 1200m² and its South and East views are shown in the figure below. The building is composed of 17 apartments, a landing, a garage / crawl space and attic. Each of these areas has been simulated as a thermal zone. The model has 20 thermal zones in total. It is located in Chambéry (Savoie, France) at an altitude of 235m.



The buildings walls, intermediate floor, partitions and floor are made of concrete. Walls have gypsum plaster board and exterior insulation. Floor on crawl space has Polystyrene insulation; however the Flat roof and Floor under attic have polyurethane insulation and glass wool insulation respectively. The windows in rooms are PVC double glazed windows with overall heat transfer coefficient $U_w = 1.3 \text{ W}/(\text{m}^2 \cdot \text{K})$. The number of occupants in an apartment kept equal to the number of room in the apartment. The occupancy for different hours during weekdays and weekend are varied from 25% - 50% - 100%. The internal heat gains of housing (related to cooking, electrical appliances, etc.) are taken into account on hourly basis and vary between $1.5 \text{ W}/\text{m}^2$ to $7.68 \text{ W}/\text{m}^2$. The cooling and natural ventilations have been taken into account from 15 July to 15 August. The infiltration was considered $0.15 \text{ vol} \cdot \text{h}^{-1}$. The mechanical ventilation varies with size of the apartment.

Dynamic thermal simulations were run in Energy+ 8.5 software. This software models the thermal comfort, ventilation, infiltrations, internal heat gains, and the heating and cooling loads of the building. The weather file (Meteonorm) used is a typical year weather of the city of Chambéry (Savoie, France).

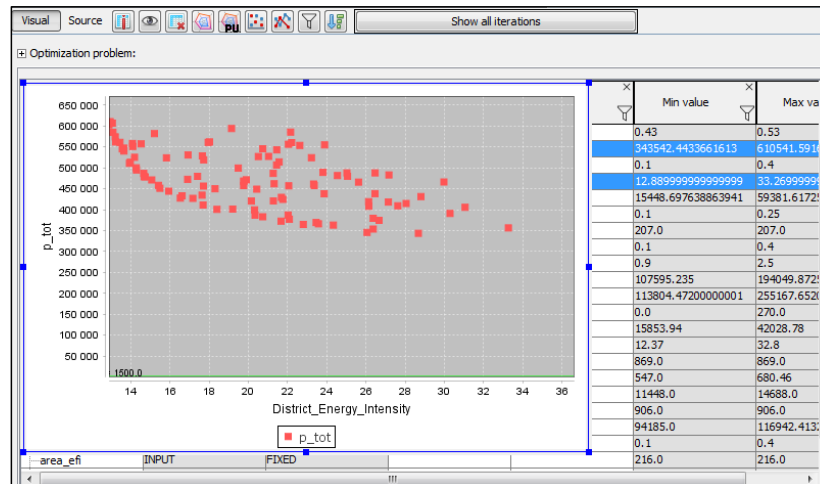
The model for this case is a lot more complex than the previous examples, meaning that the calculation time is a lot longer. We will provide the tagged files for the models, as well as the interface sml files. In order to be able to run the optimizations on this model, you have to:

1. Import the EnergyPlus model and select the .idf and .epw files, OR, open Project5 directly.
2. Create a new SML file and copy/paste the contents of the sml file.
3. Generate the MUSES (first for the EnergyPlus model, and then for the sml model).
4. Complete the inputs information if you wish to use the corresponding values:

Variable name	Label	Unit	Type of constraint	Value/Initial guess	Min	Max	Description	Save value?
area_efl	Surface Area of the...	m2	FIXED	216.0			{type=FIXED,value...	<input checked="" type="checkbox"/>
area_ewl	Surface area for Ext...	m2	FIXED	906.0			{type=FIXED,value...	<input checked="" type="checkbox"/>
area_ifc	Surface Area of the...	m2	FIXED	869.0			{type=FIXED,value...	<input checked="" type="checkbox"/>
area_iri	Surface Area of the...	m2	FIXED	243.0			{type=FIXED,value...	<input checked="" type="checkbox"/>
area_uwl	Surface Area of the...	m2	FIXED	207.0			{type=FIXED,value...	<input checked="" type="checkbox"/>
efft	Ext Floor Insulator ...	m	INTERVAL		0.1	0.4	{type=INTERVAL,v...	<input checked="" type="checkbox"/>
ewit	External Wall Insula...	m	INTERVAL		0.1	0.4	{type=INTERVAL,v...	<input checked="" type="checkbox"/>
ifct	Inter Floor Concret...	m	INTERVAL		0.1	0.25	{type=INTERVAL,v...	<input checked="" type="checkbox"/>
naxis	The building orienta...	°	INTERVAL		0.0	270.0	{type=INTERVAL,v...	<input checked="" type="checkbox"/>
trit	Terrace Roof Insula...	m	INTERVAL		0.1	0.4	{type=INTERVAL,v...	<input checked="" type="checkbox"/>
Uw	Thermal performanc...	W/m2K	INTERVAL		0.9	2.5	{type=INTERVAL,v...	<input checked="" type="checkbox"/>

5. As outputs, minimize the p_tot (that corresponds to the total price) and minimize the District_Energy_Intensity.

6. As for the output, we can observe the pareto formation as on the following figure, that represents the result for the case with 35 individuals on each phase and surface acceptable relative error of $1e-4$ and Pareto acceptable relative error of $1e-5$:



Sensitivity Analysis

Another tool available on CADES is the sensitivity analyser. It is possible to use it from an SML model for both white and black-box models, the latter with lesser options.

Sensitivity analysis is the study of how the uncertainty in the output of a mathematical model or system (numerical or otherwise) can be apportioned to different sources of uncertainty in its inputs.

The process of recalculating outcomes under alternative assumptions to determine the impact of a variable under sensitivity analysis can be useful for a range of purposes including:

- Testing the robustness of the results of a model or system in the presence of uncertainty.
- Increase the understanding of the relationships between input and output variables in a system or model.
- Searching for errors in the model (by encountering unexpected relationships between inputs and outputs).
- Model simplification – fixing model inputs that have no effect on the output, or identifying and removing redundant parts of the model structure.
- Enhancing communication from modelers to decision makers (e.g. by making recommendations more credible, understandable, compelling or persuasive).
- To seek to identify important connections between observations, model inputs, and predictions or forecasts, leading to the development of better models.

There are several available approaches to deal with sensitivity analysis. The core methods can given as:

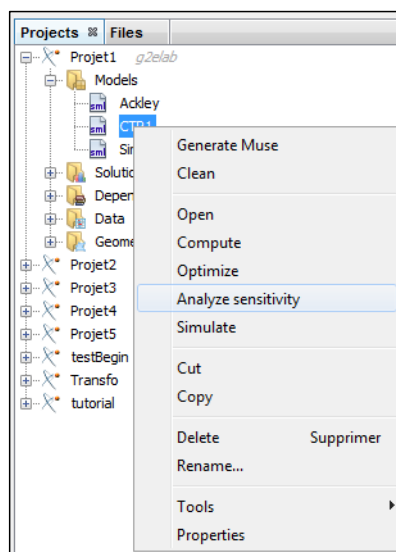
- **One-(factor)-at-time (OAT/OFAT)** : One of the simplest and most common approaches is that of changing one-factor-at-a-time, to see what effect this produces on the output. Sensitivity may then be measured by monitoring changes in the output, e.g. by partial derivatives or linear regression. This appears a logical approach as any change observed in the output will unambiguously be due to the single variable changed. Despite its simplicity however, this approach does not fully explore the input space, since it does not take into account the simultaneous variation of input variables. This means that the OAT approach cannot detect the presence of interactions between input variables.
- **Local methods**: Local methods involve taking the partial derivative of the output Y with respect to an input factor X_i . Local methods do not attempt to fully explore the input space, since they examine small perturbations, typically one variable at a time.
- **Scatter Plot**: A simple but useful tool is to plot scatter plots of the output variable against individual input variables, after (randomly) sampling the model over its input distributions.
- **Regression analysis**: Regression analysis, in the context of sensitivity analysis, involves fitting a linear regression to the model response and using standardized regression coefficients as direct measures of sensitivity. The regression is required to be linear with respect to the data (i.e. a hyperplane, hence with no quadratic terms, etc., as regressors) because otherwise it is difficult to interpret the standardised coefficients. This method is therefore most suitable when the model response is in fact linear; linearity can be confirmed, for instance, if the coefficient of determination is large. The advantages of regression analysis are that it is simple and has a low computational cost.
- **Variance based methods**: Variance-based methods are a class of probabilistic approaches which quantify the input and output uncertainties as probability distributions, and decompose the output variance into parts attributable to input variables and combinations of variables. The sensitivity of the output to an input variable is therefore measured by the amount of variance in the output caused by that input.
- **Screening**: Screening is a particular instance of a sampling-based method. The objective here is rather to identify which input variables are contributing significantly to the output uncertainty in high-dimensionality models, rather than exactly quantifying sensitivity (i.e. in terms of variance). Screening tends to have a relatively low computational cost when compared to other approaches, and can be used in a preliminary analysis to weed out uninfluential variables before applying a more informative analysis to the remaining set. One of the most commonly used screening method is the elementary effect method.

For further information and mathematical concepts and definitions on the methods used in CADES, please refer to the [bibliographic report](#) that has been previously developed or access <http://www.dimocode.fr/packkm/f/adcd2f1d-2aef-493d-8b8a-3696f271b19a>.

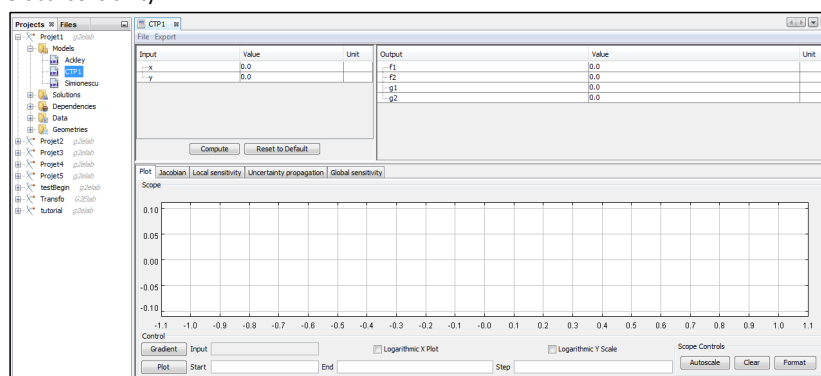
CADES White-box sensitivity analysis

For white-box models, we have access to the algebraic expressions. For differentiable functions, we can perform their derivatives or Jacobian. Because of that, we have a wider number of tools for the sensitivity analysis.

1. To make a sensitivity analysis on CADES, right click on the desired SML model and choose the option "Analyze sensitivity". For the tutorial you can use the previously created CTP1 function.



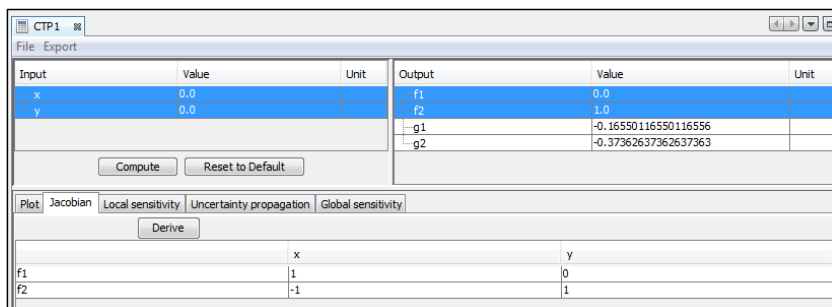
2. A window same as the calculator will open, however, with several tabs next to the plot tool (previously explained), which are: Jacobian, Local sensitivity, Uncertainty propagation and Global sensitivity



3. The Jacobian tab:

In order to calculate the Jacobian for an specific point:

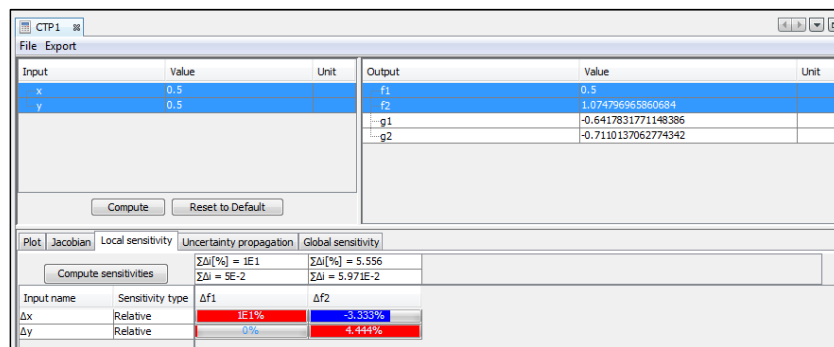
- First set the corresponding values of the inputs and click enter to confirm, if you wish, click on “Compute” (not necessary).
- Select the desired inputs and outputs (the nxm matrix of the Jacobian will be formed from this selection) and click on “Derive”



- You will get the $\frac{\delta f_j(x)}{\delta x_i}$ for the selected inputs and outputs. Changing the inputs will obviously change the values of the Jacobian matrix. We can also compute the Jacobian for the constraint functions if desired.

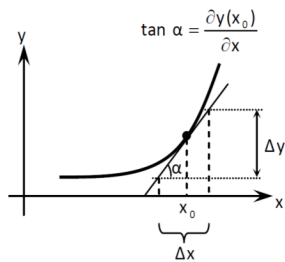
4. The Local sensitivity tab:

- Set 0.5 for both inputs and click on “Compute”.
- Select one or both inputs and one or both outputs (f1 and f2).
- Keep the sensitivity type as Relative.
- Click on “Compute sensitivities”



- Sensitivity values are displayed:
 - In Percent if the calculation is done in relative mode.
 - In absolute values if the calculation is performed in absolute mode.

- iii. In red, if they are increasing output variables.
- iv. In blue, if they are decreasing output variables.
- v. Blank, if there is no variation in the output variable concerned.
- vi. At the bottom of each output variable the interface also displays the arithmetic sum of all sensitivities.



These sensitivities are calculated, by linearizing around a point (see figure) the variation of the output(s) concerned. This linearization is obtained by approximating the variation curve of the output with its first-order partial derivative. The sensitivity calculation formula is as follows:

- In absolute terms: $\Delta y = \tan \alpha \cdot \Delta x$

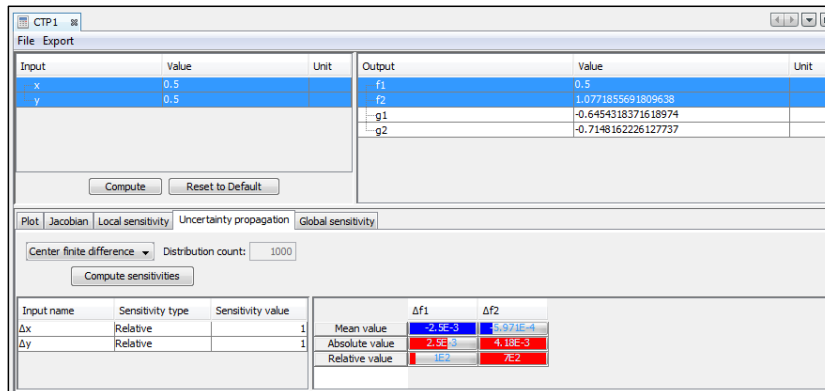
- In relative terms: $\Delta S(\%) = \frac{\Delta Y}{Y_0} \cdot 100$

- f. Change the input values and compute the sensitivity. Conclusions?

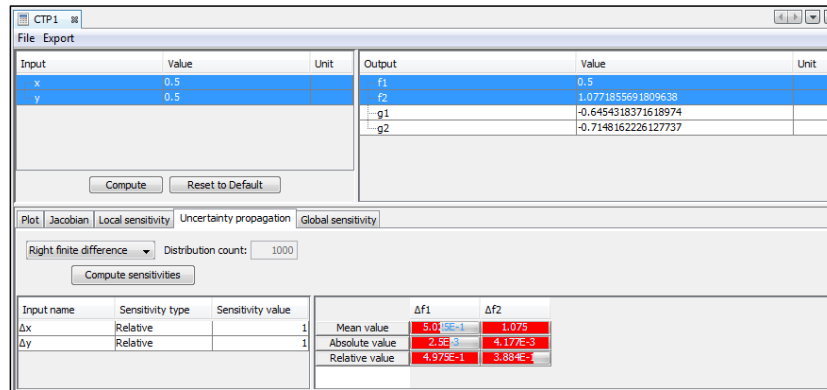
5. **The Uncertainty propagation tab:**

This tab allows us to obtain the uncertainty propagation using three different methods: Monte Carlo, Center Finite Difference and Right Finite Difference and will give as outputs, the Mean, Absolute and Relative values for the uncertainty with respect to the corresponding input points.

- a. Set 0.5 for both inputs values and compute.
- b. Select the Center Finite Difference and click on “Compute sensitivities”



- c. Select Right Finite Difference and click on “Compute sensitivities”



- d. Select Monte Carlo and click on “Compute sensitivities”
 - i. Reclick on “Compute sensitivities”. As Monte Carlo Method selects random samples each time we will have a different result for the Mean, Absolute and Relative values. It is important to notice though, that the order of magnitude of the values remains the same.
- e. The difference in the results of each method corresponds to the difference on the sampling of the point, since each method uses a different one for the approximation of derivatives.
- f. Change the value of the inputs and compute the sensitivities. Conclusions?

6. The Global sensitivity tab:

This tab allows us to calculate the global sensitivity using the Sobol method (first order and total index), which corresponds to a method based on the analysis of variance, and DGSM, which corresponds to a local derivative method applied to N random points of the space to deduce the global importance of the variables and the existence of interaction between them. As for the type of uncertainty, we can choose Relative, Uniform or Interval.

- a. Set the values of the inputs to 0.5 and compute.
- b. Select one or both inputs and one or both outputs.
- c. Select Sobol first order with 10000 evaluation and compute.

The screenshot shows the CTP1 software interface. It has a menu bar with 'File' and 'Export'. Below the menu bar, there are two tables: 'Input' and 'Output'. The 'Input' table has columns for 'Input', 'Value', and 'Unit', with rows for 'x' (0.5) and 'y' (0.5). The 'Output' table has columns for 'Output', 'Value', and 'Unit', with rows for 'f1' (0.5), 'f2' (1.0795745027594188), 'g1' (-0.6490810016877866), and 'g2' (-0.7186192646997502). Below these tables are 'Compute' and 'Reset to Default' buttons. Further down, there are tabs for 'Plot', 'Jacobian', 'Local sensitivity', 'Uncertainty propagation', and 'Global sensitivity'. The 'Global sensitivity' tab is active, showing 'Sobol first order index' and 'Evaluations 10000'. Below this, there are 'Compute Indices' and 'Compute' buttons. At the bottom, there are two summary tables. The first table shows 'E(y) = 4.999E-1' and 'E(y) = 1.097', and 'Var(y) = 8.333E-2' and 'Var(y) = 1.176E-1'. The second table shows 'f1' and 'f2' with values 1.216 and 4.201E-1, and 0 and 7.665E-1 respectively.

Input	Value	Unit
x	0.5	
y	0.5	

Output	Value	Unit
f1	0.5	
f2	1.0795745027594188	
g1	-0.6490810016877866	
g2	-0.7186192646997502	

Parameter	Variation type	ΔX	Distribution...	f1	f2
x	Relative	1.0	Uniform	1.216	4.201E-1
y	Relative	1.0	Uniform	0	7.665E-1

- Change the value of the inputs in a range from 0 to 1 (domain of the function) and compute the sensitivity. Conclusions?
- Vary the number of evaluation from 10 to 100000. Conclusions?
- Change the global sensitivity method to Sobol total index and DGSM. Also Vary the number of evaluation from 10 to 100000 and the input values. Conclusions?

CADES black-box sensitivity analysis

As stated before, for black-box models, there is a limitation on the available tools, since we don't have access to the derivative or to the Jacobian of the functions. That way, on CADES, our analysis is restricted to methods using the analysis of variance, in our case, the Sobol method.

- On project 2, right click on models and create a new SML model and call it smlBegin.
- Drag and drop Begin (TRNSys) model into the SML model.
- Replace "my_trnsys_exe_value" by the path of TRNSys (for instance, "c:/Trnsys17/Exe/TRNExe.exe").
- Declare "parameter my_timeout_value = -1;" (just complete the -1 at the end).

The screenshot shows the smlBegin software interface with a source code editor. The code is as follows:

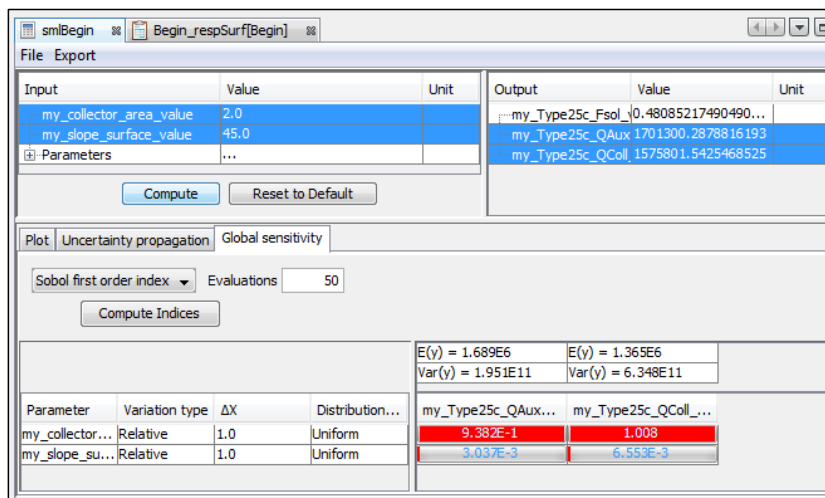
```

1 import org.gu.g2elab.project2.Begin:1.0;
2 //New model instance:
3 my_Begin = new Begin;
4
5 //Affect scalar inputs:
6 my_Begin.collector_area = my_collector_area_value;
7 my_Begin.slope_of_surface = my_slope_of_surface_value;
8
9 //Affect scalar parameters:
10 parameter my_trnsys_exe_value = "c:/Trnsys17/Exe/TRNExe.exe";
11 parameter my_timeout_value = -1;
12 my_Begin.trnsys_exe = my_trnsys_exe_value;
13 my_Begin.timeout = my_timeout_value;
14
15 //Get scalar outputs:
16 my_Type25c_Fsol_value = my_Begin.Type25c.Fsol;
17 my_Type25c_QColl_value = my_Begin.Type25c.QColl;
18 my_Type25c_QAux_value = my_Begin.Type25c.QAux;

```

- Right click and "Generate MUSE" for TRNSys model and then for the SML model.

6. Right click on “simBegin” and pick “Analyze sensitivity”.
7. Select the Global sensitivity tab.
8. Set 2 for the collector area and 45 for the collector slope input values and click on “Compute”
9. Select one or both inputs and one or both outputs.
10. Select Sobol first order index with 50 evaluations and click on “Compute Indices”.



Input	Value	Unit	Output	Value	Unit
my_collector_area_value	2.0		my_Type25c_Fsol_0.48085217490490...		
my_slope_surface_value	45.0		my_Type25c_QAux 170.1300.28788.16.193		
Parameters	...		my_Type25c_QColl 1575801.5425468525		

Parameter	Variation type	ΔX	Distribution...
my_collector...	Relative	1.0	Uniform
my_slope_su...	Relative	1.0	Uniform

my_Type25c_QAux...	my_Type25c_QColl_...
9.332E-1	1.008
3.037E-3	6.553E-3

11. Vary the input values ([1 10] for collector area and [0 90] for collector slope). Conclusions?
12. Change the variation (ΔX) to 10% and 20%. Conclusions?
13. Change the number of evaluations to 100 and 1000 (this may take several minutes, even hours to calculate). Conclusions?

Besides the limitations on sensitivity analysis tools, black-box sensitivity analysis also has the drawback of taking a lot of time when dealing with more complex models, since a co-simulation is necessary. Usually, over 1000 evaluations are necessary in order to get the right grasp on the magnitude and ranking of sensitivities.

Answers

Ackley's functions

Change the initial value of x and y and restart optimization. You can change the initial value several times. Conclusions?

It is important to know that SQP methods solve a sequence of optimization subproblems, each of which optimizes a quadratic model of the objective subject to a linearization of the constraints. If the problem is unconstrained, which is the case of Ackley's function, then the method reduces to Newton's method for finding a point where the gradient of the objective vanishes. If the problem has only equality constraints, then the method is equivalent to applying Newton's method to the first-order optimality conditions, or Karush–Kuhn–Tucker conditions, of the problem.

Newton's method is only guaranteed to converge if certain conditions are satisfied. If the assumptions made in the proof of quadratic convergence are met, the method will converge. For the following subsections, failure of the method to converge indicates that the assumptions made in the proof were not met.

If we find a point far enough, the result might diverge instead of converging. If we choose 100 for both initial conditions, we get the following result:

```
15:18:09: SQP - Gradient Sequential Quadratic Programming started
15:18:09: 6 model evaluations were necessary.
6 model jacobian evaluations were necessary.
Optimizer message: No solution: The derivatives are not correct or the required precision can not be reached.
Try to increase the value of the precision !
Result found:
f = 22.35040211167449 {type=MAXIMIZE,value=20.0,weight=1.0}
Done in 62ms
```

Meaning that the quadratic convergence was not met and the method didn't converge to the optimal. Usually, the initial guess has to be close enough from the convergence point.

By graphical analysis we can see that the global minimum is on $(x,y) = (0,0)$. However, because of SQP approach for non constrained optimization, depending on the value of the initial guess, we will find different values of local optima.

Increase the SQP optimizer "Precision" (from 1e-5 to 1e-10) and start again. Conclusions?

If the solver doesn't reach the desired precision, the optimization will stop due to the Max iteration criteria and will return the with a precision larger than the desired one.

What SQP means?

SQP means Sequential quadratic programming. It is an iterative method for constrained nonlinear optimization. Those methods are used on mathematical problems for which the objective function and the constraints are twice continuously differentiable.

On the optimizer tab, test other algorithms and make some general conclusions:

Depending on the algorithm we can obtain different results. For instance, for $(x,y) = (2,1)$ as inputs, we get $f = 12.642411176571153$ for SQP, $f = 14.302631084952282$ for DIRECT and $f = 14.113776031203743$ for a 50x100 NSGA-II.

This difference, roughly speaking, is due to the way the algorithm is built, meaning that the result will depend on the algorithm implementation, and the choice of the best algorithm will actually depend on the type of the problem, and a compromise of precision/calculation time is usually made.

Simionescu function

Draw some conclusion on the algorithm properties.

The main point in trying different algorithms is to observe how they deal with different local minima and if it is able to find the global optimal from any given initial values.

CTP1 function

Compare this strategy with NSGAII Algorithm.

The difficulty using NSGA II algorithm is to determine the number of population and generation that will give us a good result. Besides, NSGA II algorithm takes a lot longer in order to determine the solution. So, why using NSGA II algorithms after all?

As stated before, in order for SQP to properly work, we need a mathematical model from the problem, and more than that, the cost functions and objectives have to be at least twice differentiable. Not all optimization problems fall under this case. Besides, errors during the optimization process may arise from unsolvable solution spaces (badly defined or non-differentiable equations for instance) and/or unfeasible combination of variables. A single simulation failure may crash the entire optimization process. Evolutionary algorithms, such as NSGA II, can simply discard a bad solution, as it is a solution among a population, making this sort of algorithms surprisingly robust to high failure rate problems.

The aid in the choice of an appropriate algorithm is predicted in CADES future development.

Case Begin

Change the initial value of collector_area and slope_of_collector. You can change the initial value several times. Conclusions?

Response surface method uses evolutionary algorithms and LHS, meaning that the initial samples and the population will never be the same. As a result, we will never get the exactly same result, but the outputs will always be close one to another.

However, as for the initial values, because of algorithm implementation, it will not interfere on the results from the previous discussion.

Change the initial guess of QAux and QColl and restart optimization. You can change the initial value several times. Conclusions?

Same analysis is made for the inputs is valid for the outputs.

Change the acceptable relative error for the Surface and the Pareto. Conclusions?

If the kriging metamodel reaches a value below the one determined by the relative error, it will stop before the max iterations stop criteria based on the number of individuals of each phase.

```
09:52:51: Meta-model acceptable relative error reached (xrerror = {0}, covariance = {1}). Surface improvement done.
09:52:52: Meta-model acceptable relative error reached. Pareto improvement done.
09:52:52: 52 model evaluations were necessary.
0 model jacobian evaluations were necessary.
Optimizer message: Good
```

Increase and/or decrease the number of individuals for each step (Initial, Surface improvement, Pareto improvement). What happens to the Pareto front? Any other conclusions?

By increasing the number of individuals of each phase, we increase the precision to the original model. However, as it requires more iterations, the time to compute the solution will also increase.

CADES White-box sensitivity analysis available tools

The Local sensitivity tab: Change the input values and compute the sensitivity. Conclusions?

As we are calculating the local sensitivity, by setting new values for the inputs, we will get different values of sensitivity. Besides, each set of input points can have a different rankings when determining which point is more critical with respect to a given output.

The Uncertainty propagation tab: Change the value of the inputs and compute the sensitivities. Conclusions?

The uncertainty propagation depends on the location of the points on the function. So, when we change the the input, we change the output point, what gives a different result when computing the uncertainties.

The Global sensitivity tab: Change the value of the inputs in a range from 0 to 1 (domain of the function) and compute the sensitivity. Conclusions?

Althought the value of the sensitivity may change a little, the rank and the order of magnitude of the variables shall remain the same if we consider a large enough number of evaluations.

The Global sensitivity tab: Vary the number of evaluation from 10 to 100000. Conclusions?

After a certain number of evaluations the values of the sensitivities stabilize. This analysis is also valid for black-box sensitivity analysis.

The Global sensitivity tab: Change the global sensitivity method to Sobol total index and DGSM.

Since the methodology of each method is different, we can expect different values of sensitivity. However, for the right number of evaluations, which we get stable results, both methods will present the same input as most critical with respect to each output.