Master’s Thesis

Master’s Degree in Industrial Engineering

Development of a hybrid algorithm for bi-level bi-objective optimization, and application to hydrogen supply chain deployment and design

October, 2021

Author: Rubén ARGUEDAS NAVARRETE

Directors: Catherine AZZARO-PANTEL
Antonin PONSICH
Víctor Hugo CANTÚ MEDRANO

Institut National Polytechnique de Toulouse

École Nationale Supérieure des Ingénieurs en Arts Chimiques et Technologiques

Abstract

The present master thesis is based on the recently presented doctoral thesis of Dr. Victor Hugo Cantu Medrano, addressing multiobjective optimization problems in Process Engineering with several alternative resolution methods using Evolutionary Computation.

In his thesis, a new algorithm to find the optimal design of the Hydrogen Supply Chain while minimizing economic costs and environmental impact is presented. For its resolution, the algorithm divides the problem into two subproblems or levels. The first level deals with the design of the HSC structure (sizing and location of the facilities). A second level that solves the subproblem corresponding to the operation of the supply chain (production and transportation). The technique used for its resolution is a hybridization of the MOEA SMS-EMOA, for the first level, with a linear programming solver that uses a scalarization function to address the two objectives considered in the second level.

In this line, this master thesis consists of developing an extension of this same algorithm with the objective of taking advantage of all the information generated in the second level to increase its efficiency. To achieve this, the second level is executed several times for each execution of the first level, using each time a different vector of weights in the scalarization function. But this new logic implies the readaptation of the whole algorithm.

First, the Hydrogen Supply Chain problem is presented and the technique for solving the original algorithm is discussed. Subsequently, the necessary modifications to the MOEA are presented in order to be able to apply the new approach to the algorithm. With the new algorithm implemented, a study is carried out for the definition of the weight vectors and different scalarization functions are studied to try to increase its efficiency. Finally, the results obtained with the new algorithm and those of the original algorithm are compared to determine whether the new version is capable of solving the same problems using fewer computational resources.

Résumé

Cette thèse de master est basée sur la thèse de doctorat récemment soutenue par Dr Víctor Hugo Cantú Medrano, dans laquelle il expérimente plusieurs méthodes de résolution alternatives à l'aide méthodes évolutionnaires pour résoudre les problèmes d'optimisation multiobjectifs dans le domaine du génie des procédés.

Dans sa thèse, le Dr Cantú présente un nouvel algorithme permettant de trouver la conception optimale de la chaîne d'approvisionnement en hydrogène tout en minimisant les coûts économiques et l'impact environnemental. Pour sa résolution, l'algorithme divise le problème en deux sous-problèmes ou niveaux. Le premier niveau traite de la conception de la structure de la chaîne logistique hydrogène (dimensionnement et emplacement des installations). Un second niveau résout le sous-problème correspondant à l'exploitation de la chaîne logistique (production et transport). La technique utilisée pour sa résolution est une hybridation du MOEA SMS-EMOA, pour le premier niveau, avec un solveur de programmation linéaire qui utilise une fonction de scalarisation pour traiter les deux objectifs considérés dans le second niveau.

Dans cette lignée, ce mémoire de master consiste à développer une extension de ce même algorithme avec l'objectif de tirer profit de toute l'information générée dans le deuxième niveau pour augmenter son efficacité. Pour ce faire, le second niveau est exécuté plusieurs fois pour chaque exécution du premier niveau, en utilisant à chaque fois un vecteur de poids différent dans la fonction de scalarisation. Mais cette nouvelle logique implique la réadaptation de l'ensemble de l'algorithme.

Tout d'abord, le problème de la chaîne logistique hydrogène est présenté et la technique de résolution de l'algorithme original est discutée. Ensuite, les modifications nécessaires au MEOA sont présentées afin de pouvoir appliquer la nouvelle approche à l'algorithme. Avec le nouvel algorithme implémenté, une étude est réalisée pour la définition des vecteurs de poids et différentes fonctions de scalarisation sont étudiées pour essayer d'augmenter son efficacité. Enfin, les résultats obtenus avec le nouvel algorithme et ceux de l'algorithme original sont comparés pour déterminer si la nouvelle version est capable de résoudre les mêmes problèmes en utilisant moins de ressources informatiques.

Resumen

Este Trabajo Final de Master parte de la tesis doctoral recientemente presentada del doctor Víctor Hugo Cantú Medrano, donde se abordan problemas de optimización multiobjetivo en Ingeniería de Procesos experimentando con varios métodos de resolución alternativos haciendo uso de la Computación Evolutiva.

En su tesis, el doctor Cantú presenta un nuevo algoritmo para encontrar el diseño óptimo de la Hydrogen Supply Chain minimizando los costes económicos y el impacto ambiental. Para su resolución, el algoritmo divide el problema en dos subproblemas o niveles. Un primer nivel que aborda el diseño de la estructura de la HSC (dimensionamiento y ubicación de las instalaciones). Un segundo nivel que resuelve el subproblema correspondiente a la operación de la cadena de suministro (producción y transporte). La técnica empleada para su resolución es una hibridación del MOEA SMS-EMOA, para el primer nivel, con un solver de programación lineal que utiliza una función de escalarización para tratar los dos objetivos considerados en el segundo nivel.

En esta línea, este trabajo consiste en desarrollar una extensión de este mismo algoritmo con el objetivo de aprovechar toda la información que se genera en el segundo nivel para aumentar su eficiencia. Para lograrlo se ejecuta varias veces el segundo nivel por cada ejecución del primer nivel, utilizando cada vez un vector de pesos diferente en la función de escalarización. Pero esta nueva lógica implica la readaptación de todo el algoritmo.

En primer lugar, se presenta el problema de la Hydrogen Supply Chain y se discute la técnica de resolución del algoritmo original. Posteriormente se presentan las modificaciones necesarias en el MOEA para poder aplicar el nuevo enfoque al algoritmo. Ya con el nuevo algoritmo implementado se realiza un estudio para la definición de los vectores de peso y se estudian diferentes funciones de escalarización para tratar de aumentar su eficiencia. Por último, se comparan los resultados obtenidos con el nuevo algoritmo y los del original para determinar si la nueva versión es capaz de resolver los mismos problemas utilizando un menor número de recursos computacionales.

Resum

Aquest Treball Final de Màster té el seu origen en la tesis doctoral recentment presentada del doctor Víctor Hugo Cantú Medrano, en la qual s'aboren problemes d'optimització multiobjectiu en enginyeria de processos, experimentant amb diversos mètodes de resolució alternatius fent ús de la Computació Evolutiva.

En la seva tesis, el doctor Cantú presenta un nou algorisme per a trobar el disseny òptim de la Hydrogen Supply Chain minimitzant els costos econòmics i l'impacte ambiental. Per a la seva resolució, l'algoritme divideix el problema en dos subproblemes o nivells. Un primer nivell aborda el disseny de l'estructura.de la HSC (dimensionament i ubicació de les instal·lacions). Un segon nivell resol el subproblema corresponent a l'operació de la cadena de subministrament (producció i transport). La tècnica empleada per a la seva resolució és una hibridació del MOEA SMS-EMOA, per al primer nivell amb un solver de programació lineal que utilitza una funció d'escalarització per a tractar els dos objectius considerats en el segon nivell.

En aquesta línia, aquest treball consisteix a desenvolupar una extensió d'aquest mateix algorisme amb l'objectiu d'aprofitar tota la informació que es genera en el segon nivell per a augmentar la seva eficiència. Per a aconseguir-ho s'executa diverses vegades el segon nivell per cada execució del primer nivell, utilitzant cada vegada un vector de pesos diferent en la funció d'escalarització. Però aquesta nova lògica implica la readaptació de tot l'algorisme.

En primer lloc, es presenta el problema de la Hydrogen Supply Chain i es discuteix la tècnica de resolució de l'algorisme original. Posteriorment es presenten les modificacions necessàries en el MOEA per a poder aplicar el nou enfocament a l'algorisme. Ja amb el nou algorisme implementat es realitza un estudi per a la definició dels vectors de pes i s'estudien diferents funcions d'escalarització per a tractar d'augmentar la seva eficiència.

Ja amb el nou algorisme implementat es realitza un estudi per a la definició dels vectors de pes i s'estudien diferents funcions d'escalarització per a tractar d'augmentar la seva eficiència. Finalment, es comparen els resultats obtinguts amb el nou algorisme i els de l'original per tal de determinar si es possible obtenir els mateixos resultats fent us d'un menor número de recursos computacionals.

SUMARI

1. INTRODUCTION ____ 9

2. RELATED WORK AND BASIC CONCEPTS ____ 11

2.1. The Hydrogen Supply Chain 11
2.1.1. Generalities on hydrogen 12
2.1.2. Presentation of the HSC main goals and challenges 13
2.1.3. The HSC design as an optimization problem (single/multi-obj) 14
2.2. Multi-objective Optimization 15
2.2.1. General presentation and formulation 15
2.2.2. Different classes of solution techniques 17
2.2.3. Different scalarizing technics 20
2.2.4. The SMS-EMOA algorithm 21
2.2.5. SBX crossover operator 23
2.2.5.1. Polynomial Mutation 23

3. PROBLEM STATEMENT AND EXISTING SOLUTION SCHEMES __25

3.1. Problem description 25
3.2. Mathematical model 27
3.3. Current resolution strategy for the HSC bi-level problem 29

4. A NEW HYBRID ALGORITHM FOR THE HSC DESIGN PROBLEM._33

4.1. Proposed improvement strategy 33
4.2. Selecting the scalarizing function 35
4.2.1. Reference point for hypervolume computation 36
4.2.2. Results and discussion 36
4.2.2.1. Analysis of the differences with respect to the best hypervolume.........37
4.3. Adaptation of the upper-level MOEA selection operator 38
4.3.1. Context 38
4.3.2. The proposed operator design 40
4.3.3. Parents population selection 44
4.4. Implementation of the algorithm 44
4.5. Smart management of weight vectors 46
4.5.1. Weight generation 46
4.5.2. Weight selection 47

5. EXPERIMENTAL COMPUTATIONS ____ 50

5.1. Experimental methodology 50

5.2. Global Results 51
5.3. Any-time analysis 52
6. CONCLUSIONS ________ 56
ACKNOWLEDGMENTS ________ 59
APPENDIX A. DATA INSTANCES ________ 60
BIBLIOGRAPHY ________ 71

1. Introduction

In a world increasingly aware of environmental issues, alerted to the need for changes in the use of energy, engineering must be able to respond to these future challenges and continue to meet our needs.

This project presents an algorithm from the Evolutionary Computing branch aimed at designing a Hydrogen Supply Chain in the French region of Midi-Pyrénées optimizing the economic and environmental costs, since, in order for hydrogen to be considered as an alternative energy source, economic efficiency must be achieved. At the same time the carbon footprint produced by fossil fuels must be reduced.

The objective of the project is the development and experimentation of an algorithm based on evolutionary techniques that provides the different optimal configurations of the design of the Hydrogen Supply Chain for the territory. More precisely, the algorithm provides a set of solutions belonging to a Pareto Front, i.e., the types and size of the production and storage facilities and the mode of transport used from a variety of energy sources.

To achieve this objective, the current situation of hydrogen as an energy source has been investigated and the literature on multi-objective optimization problems has been analyzed. Specifically, the algorithm has been built as an extension of the original proposal developed by Víctor Hugo Cantú in the work "A Novel Matheuristic based on Bi-Level Optimization for the Multi-Objective Design of Hydrogen Supply Chains"[1].

In order to propose the new algorithm, it has been necessary to carry out a study of Victor's proposal, propose a readaptation of the algorithm, develop a computer program that implements it, make improvements on the new proposal and finally submit it to a computational experimentation to be able to draw conclusions.

The project has been structured along the following lines:

1. Study of hydrogen as an energy source
2. Literature review of the MOP
3. Definition of the HSC problem
4. Presentation of the resolution used with the original algorithm
5. Development and implementation of the new algorithm
6. Computational experimentation
7. Conclusions

The process of evolution exists in all aspects of life. Species have evolved, society has evolved, technology has evolved, the same must happen with the use of energies.

2. Related work and basic concepts

2.1. The Hydrogen Supply Chain

Despite that the effects in the raise of Earth temperature produced by the presence of carbon dioxide in the atmosphere was first pointed out in 1896 by Svante Arrhenius [2], the idea of global warming being caused by the combustion of fossil fuels did not gain strength until the late 1980s, where some effects could be measured [3]. Today, it is a well-established theory to the extent that it is present in our daily basis, it to be in the news, laws, taxes or even personal decisions. The relevance of this theory can be noted in the Paris Agreement of 2015 [4], where 195 countries agreed on controlling the carbon dioxide emissions in order to keep the global temperature rise of the *XXI* century below 2 degrees Celsius.

To achieve this goal, however, it is not expected to be an easy task, since the world will need to make dramatic changes year after year and decrease energy-related CO_2 emissions by 58% until 2050 (from 30,6 Gt in 2020 to 13 Gt in 2050). To this, must be added the difficulty that final energy consumption is expected to continue growing in the coming years [5]. This future scenario represents a problem for the reduction of CO2 emissions, since the versatility, cost and abundance of fossil fuels make them highly dependent in practically all sectors and very difficult to replace.

Given the need to reduce the effects produced by the use of fossil fuels, three branches of research have been promoted in recent years: increasing efficiency in the process of obtaining utile energy, reducing pollutants in the gases expelled during combustion or their treatment, and the seek for alternative energy sources, this last one being the most effective in the long term.

Renewable energies seem to be a promising option as an alternative to fossil fuels. But this transformation presents new challenges, such as managing the intermittency of some renewable energy sources or the difficulty of electrifying certain end users.

Therefore, in recent years, and parallel to renewable energies, there has been a growing interest in hydrogen, that can be seen as a versatile, clean, and safe energy carrier that can be used as fuel for power or in industry as feedstock. This interest has been motivated mostly since it can be produced from electricity and from carbon-abated fossil fuels, produces zero emissions at point of use, can be stored and transported at high energy density in liquid or gaseous form, and it can be combusted or used in fuel cells to generate heat and electricity [6].

But, for this to become a reality, hydrogen will first have to meet a number of challenges to make it a competitive option. Chief among these challenges is the design and construction of a hydrogen supply chain (HSC) to meet end-consumer demand in a localized, cost-effective and secure manner.

2.1.1. Generalities on hydrogen

Hydrogen, represented by the symbol H, is a chemical element with atomic number 1. It is the lightest of all elements and it is also the most abundant in the universe. On Earth, it is present mostly in the form of chemical compounds such as water or hydrocarbons.

Under standard conditions of temperature and pressure its elemental form is the gas molecule dihydrogen and is found in the atmosphere, in very low proportions. Its very low density (0.08988 kg/Nm3 at 273°K) enables it to escape the Erath gravity more easily than denser gases like oxygen [7]. From an energetic point of view, dihydrogen is the form in which hydrogen can be used as fuel; either by the energy given off at the combustion of the gas or by using fuel cells to obtain electricity.

As a fuel, it has extremely benefit qualities: its lower heating value is 141,88 MJ/kg [8], three times more than Diesel; it can be produced from electricity and carbon-abated fossil fuels and the only emissions generated at the point of use are water molecules. In a world concerned with reducing the environmental footprint, the low impact of hydrogen and its high energy capacity make hydrogen one of the energy sources of the future.

As there are neither mines nor hydrogen deposits, but rather it is found in the atmosphere, hydrogen cannot be extracted directly from nature, so it must be produced. Currently, the most competitive ways to obtain hydrogen are:

- Steam methane reforming (SMR): by reacting natural gas with high-temperature steam a mixture of hydrogen, carbon monoxide, and a small amount of carbon dioxide are generated. Currently, it is the cheapest, most efficient, and most used method to produce hydrogen.
- Electrolysis: subjecting water to an electric field induces an electric current that causes the water molecules to split between hydrogen and oxygen. If the electricity used comes from renewable font, it can be considered as a renewable energy. It can also be used to transform the excess of electricity produced by the variable renewable energy, such as wind or solar, in to hydrogen, that can then be stored for a later use.
- Gasification of biomass and coal: reacting coal or biomass with high-temperature steam and oxygen in a pressurized gasifier, results in a synthesis gas that contains hydrogen and carbon monoxide. That is later reacted with steam to separate the hydrogen.

- Fermentation: biomass is converted into sugar-rich feedstocks that can be fermented to produce hydrogen.

There are other methods in a developing phase, like High-Temperature Water Splitting, Photobiological Water Splitting or Photoelectrochemical Water Splitting. All this these three methods have the same scientific basis, that consists in splitting the water molecules to separate the hydrogen from the oxygen of the water molecules [9].

Although hydrogen contains a lot of energy per unit weight, it also occupies a large volume per unit weight making it very difficult to store it. To overcome this problem, a hydrogen conditioning process is necessary. There are three main alternatives: the first one is liquefaction. It is the way that hydrogen gets more concentrated, reducing up to 800 times the initial volume. By this method, hydrogen reaches a Net Calorific Value per weight unit much higher than Diesel [8]. Another way to store hydrogen is by compression. In this process hydrogen is compressed to pressures between 35 MPa to 70 MPa. The last alternative is based on metal hydrides. Those consists of materials that absorb hydrogen under certain conditions and allow reversible reaction, acting as charge carriers [10]. These methods are complex and require special storage equipment next to production facilities or demand points.

2.1.2. Presentation of the HSC main goals and challenges

Although the technology to produce, distribute and consume hydrogen is known, it still in an early phase. The lack of infrastructures, initial large-scale investments and the cost associated to it treatments make it very difficult to enter on today’s market. For these reasons, today, hydrogen has a restricted use and is used mainly in industrial processes. So, a horizon of 10 to 20 years should be contemplated for creating a “hydrogen society”. The Council of Hydrogen, a global consortium of 92 energy companies, consider the role of hydrogen as a game changer in energy transition, accentuating seven major roles that hydrogen will have in the energy transition:

- Enabling large-scale renewable energy integration and power generation. It can be used as a leveler between electricity generation and demand, due to imbalances in the electricity generation sector caused by variable renewable energy sources. Hydrogen produced by electrolysis could function as energy storage to overcome seasonality.
- Distributing energy across sectors and regions. As a method of exchange across borders, securing a safe way to transfer storage energy by ships, trucks or pipes. Transporting hydrogen seems to be very efficient and it is not limited to small distances like electricity.
- Acting as a buffer of the energy system, in addition to fossil fuels. It is necessary as backup capacity to ensure the smooth functioning of the energy system. As energy consumption will still raise, more amount of buffering is expected.

- Decarbonize transport. Fuel cell electric vehicles (FCEVs) have an important role to play in decarbonizing transport. As an alternative to battery electric vehicles (BEVs), FCEVs are expected to achieve longer distance, refuel quickly (3 to 5 minutes), and weight less due to no need of batteries. It would also make it possible to decarbonize other types of vehicles such as ships and airplanes in which the use of batteries is limited by their high weight.
- Decarbonize industry energy use. Combusting hydrogen for low-grade heating (below 400 °C) or as an electricity supply using fuel cells.
- Serve as feedstock using captured carbon. Carbon capture and utilization is a technology that will need hydrogen to convert captured carbon into usable chemicals like methanol, methane, formic acid, or urea.
- Help decarbonize building heating. Today, this activity is responsible for 12% of the total global carbon dioxide emissions, and hydrogen could substitute fossil fuels currently used.

However, the current problem hydrogen is facing is cost competition. In the paper "Path to hydrogen competitiveness" [11] a realistic approach of the future competitiveness of the hydrogen cost is presented. In this study, 35 possible applications of hydrogen and more than 40 technologies are analyzed and compared to other rival goods with the aim of giving a reasonable cost perspective. As a result of the study, it concludes that more than 22 applications would be cost competitiveness before 2030, first becoming competitive in transportation, particularly for large vehicles with long range. Even this, in the short term, all application will struggle due to the higher cost of hydrogen technology and limited infrastructure and scale. This problem is expected to be overcome after 2030.

2.1.3. The HSC design as an optimization problem (single/multi-obj)

Various works can be found in the literature that formulate the design of the HSC as an optimization problem.

A first work was introduced in [12], where a general demand-driven model was proposed to determine the optimal design of a network for the production, transport and storage of hydrogen in the UK with a Mixed Integer Linear Programming (MILP) approach, and a single-objective of minimize the total operating and infrastructure costs. Later, the same authors expanded this model by adding new parameters to consider, such as the availability of raw material and its logistical cost, the variation in hydrogen demand along a planning horizon that leads to the development of infrastructure by phases, and the possibility of selecting different scales for the production and storage facilities [13].

The same model has been extended and replicated in other works readapting it in other territories like Germany [14], France [15], Korea [16] or Portugal [17], adding new parameters and considering the minimization of environmental impact or risks derived from the installation and activity of the infrastructure as objectives in the model. Some of these models even consider the uncertainty in the hydrogen demand and look for the best possible configuration of the HSC for a set of demand scenarios.

Due to the large number of variables used, and in order to keep the numerical complexity limited, the proposed models tend to be single-objective (considering only the cost) or bi-objective (cost + another objective). Usually, the approach used to solve them is by exact technics, like e-constrain.

This project replicates the model of [18], which in turn is based on the models of [13] and [19], and seeks to optimize the design of a supply chain in the Midi-Pyrénées region of France by minimizing two objectives:

- Total daily cost (TDC) of the supply chain, considering the investment costs related to plant installation and transportation routes, operational costs for production, storage and transportation, and also constraints for plant capacity, mass balance between grids and demand satisfaction.
- Global warning potential (GWP), computed as a mathematical relation for Greenhouse Gas emissions due to production, storage and transportation.

2.2. Multi-objective Optimization

2.2.1. General presentation and formulation

Multi-objective optimization can be defined as the search for the best solution(s) to a problem, i.e. the optimal solution(s), among a set of solutions that meet certain specifications, by evaluating two or more independent criteria called objectives.

Each solution is a unique combination of n decision variables and constitutes the input to the problem. The way to express them is with its vector of decision variables $\vec{x}$.

$$\vec{x} = [x_1, x_2, \dots , x_n]^T$$

The decision variables of a solution cannot adopt any value, but are subject to constraints that limit and correlate them to ensure that they comply with the problem specifications. Thus, when the value of any of the variables of a solution is altered to construct another one, the other variables might be modified accordingly in order to meet the constraints.

There are two types of constraints:

- Inequality constraints:

$$g_i(\vec{x}) \leq 0 \quad i = 1, \dots, m$$

- Equality constraints:

$$h_j(\vec{x}) = 0 \quad j = 1, \dots, p$$

The set of solutions that satisfy all constraints is called set of feasible solutions, Ω, and only these solutions are candidates to be considered.

In order to evaluate a solution, one or several objective function(s) is (are) applied, returning a numerical value(s) (output), therefore allowing to compare two solutions. In some cases, an output as low as possible is desired and, in others, as high as possible (maximization problem). Here, $k \geq 2$ objectives are contemplated, so that each solution is evaluated k times, once for each objective, ultimately obtaining a vector of objective functions:

$$\vec{f}(\vec{x}) = [f_1(\vec{x}),\ f_2(\vec{x}), \dots\ , f_k(\vec{x})]^T$$

In general, these objectives are conflicting /meaning that improving some objective leads to the deterioration of another one). Therefore, in many cases, when comparing two solutions, one may solutions have better values for some objectives and worse values for others. In these cases, it is not possible to determine the best candidate solution. In order to formalize this observation, the dominance relationship is established: assuming that all objectives are to be minimized, a solution $\vec{x}_1$ is said to dominate another solution $\vec{x}_2$, if $\vec{x}_1$ is not worse (i.e., lower or equal to) than $\vec{x}_2$ for all objectives and strictly better (lower) than $\vec{x}_2$ for at least one objective:

$$\forall \mathbf{s}\ \exists\ f_s(\vec{x}_1) :\ f_s(\vec{x}_1) \leq f_s(\vec{x}_2)\ \wedge\ \exists\ f_l(\vec{x}_1) : f_l(\vec{x}_1) < f_l(\vec{x}_2) \qquad s, l = 1, \dots, k$$

Accordingly, the definition of Pareto-optimality (for Multi-objective Optimization Problems, MOPs) is based on the dominance relationship: a solution is Pareto-optimal if it is not dominated by any other solution in Ω. The set of all non-dominated solutions is called the Pareto set and its mapping onto the objective space is called the Pareto front.

The ideal point, $\vec{f}^*$, contains the optimal values of Ω for each objective individually, regardless the value of the other objectives. This vector denotes the lower bound for each objective and is in general (when objectives are conflicting) infeasible, but it can serve as a reference. With f_s^* representing the optimal value of each objective for all s = 1, …, k, the ideal point it is defined as follow:

$$\vec{f}^* = [f_1^*, f_2^*, \dots\ , f_k^*]^T$$

The nadir point, $\vec{f}^{\text{nad}}$, represents the worst individual values of each objective among all the solutions that build the Pareto front. The following figure shows the ideal and nadir point for the Pareto front of a bi-objective optimization problem (where the feasible space is denoted here by Z).

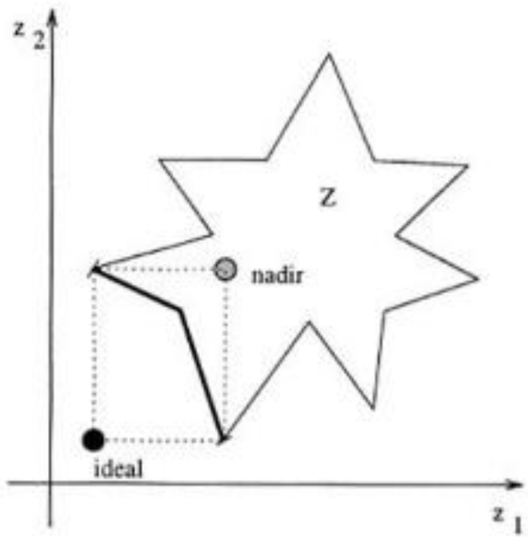


Figure 1. Representation of the ideal and nadir point of a set of solutions Z. Source:[20]

These points give a reference of the quality of the solutions obtained and allow to normalize the objectives. For most cases, it is necessary to carry out this normalization because the objectives have different units and magnitudes, since, they can evaluate very different concepts, such as for example the manufacture cost of a product and a customer's satisfaction degree. One way to normalize a target is as follows:

$$f_s{}'(\vec{x}) = \left| \frac{f_s^* - f_s(\vec{x})}{f_s^* - f_s^{nad}} \right| \qquad s = 1, \dots, k$$

2.2.2. Different classes of solution techniques

In the previous section, basic concepts such as the Pareto-optimality definition were stated for MOPs. However, this new definition of optimality involves the need for specific solution techniques and algorithms, in order to find an approximation of the Pareto set. These techniques are based on different paradigms in order to identify efficiently a defined number of Pareto-optimal solutions, i.e. in the shortest possible time.

A traditional way to approach a MOP is by transforming it into a single-objective problem through scalarizing (or utility) functions. These functions combine the vector of normalized objectives and obtain a single output for each solution. The scalarizing functions used relate the objective vector to a predefined weight vector w := (w$_1$, w$_2$, ..., w$_k$), where each weight represents the importance given to an objective. Generally, the weight vector is normalized (even if not necessary). Assuming that all objectives are to be minimized, the new single objective problem has the following form:

$$minimize \quad u(\vec{f}'(\vec{x})\,; w)$$

$$subjected\ to \quad \vec{x} \in \Omega$$

Where,

$$\sum_{s} w_s = 1 \mid w_s \in [0,1] \qquad s = 1, \dots, k$$

Any single-objective optimization technique adapted to the mathematical features of the above problem can be used. The solution represents one single Pareto-optimal point. An approximation of the whole Pareto front can be obtained by varying the objective weights, thus exploring different regions of the objective space.

There is a wide variety of scalarizing functions that combine the objectives in different manners. Some of them are described in details in the following subsection because they are employed in this work, but the reader is referred to [26] for a complete overview on this topic. The scalarizing strategy, however, has three major drawbacks: (1) since only one point is obtained for each combination of weights, if one wants to obtain an approximation of the whole Pareto front, multiple runs must be performed with different weight vectors (a number that should increase with the number of objectives considered), (2) the approximation of the Pareto front may not be well distributed, especially in concave parts of the front, and (3) this technique is sensitive to the selection of the weight coefficients, the problem treated must be well known so that the selection of the weight vector provides a satisfying result [21].

Another way to address these MO problems is evolutionary computation, in particular multiobjective evolutionary algorithms (MOEA). These algorithms have a working mode based on biological evolution, where only the best adapted individuals survive to gradually approach the Pareto front in a well-distributed manner. Their structure is usually as follows:

- The starting point is a set of µ individuals (solutions), called population.
- Each individual is evaluated for each k ≥ 2 objectives.
- Variation operators such as crossover and mutation are performed on individuals from the current population, to obtain a new generation of individuals. The crossover, which reproduces sexual recombination, determines the individuals that will be able to transmit their qualities and how they transmit them to the new generation. On the other

hand, mutation involves random variation of individual's genotype. Both these operators usually introduce a degree of randomness in the offspring. Examples of such operators are the SBX crossover and the polynomial mutation, which are classical operators adapted to real-point encoded variables. Both these operators are described in this chapter, since they will be used in this work.

- From the total population, obtained from combining the current population (parents) with the offspring population produced through variation operators (so that total population size is now greater than μ), the best μ individuals are selected according to a selection operator that assigns a fitness to each individual according to the algorithm paradigm (see next paragraphs). While a stopping criterion has not been reached, these selected individuals form the current population and the above process is repeated iteratively. The individuals that were not selected are discarded from the population. In general, the stopping criterion is related with resource usage, measured in terms of generations, objective evaluations or computational time. When the algorithm stops, the last population constitutes the output of the algorithm.

With some exceptions, the distinction between different classes of MOEAs are mostly due to differences in the paradigm used to define the selection operator, while the choice of the variation operator is generic and problem dependent [22]. There are currently three main paradigms for MOEA selection operator designs. These are:

1. Pareto based MOEAs: this type of algorithms uses a two-level selection process. First, the dominance relationship governs as a selection criterion to assign a first fitness value to each individual. In a second level, diversity indicators are used to promote the even distribution of non-dominated solutions along the Pareto front. This second level is applied only to individuals who share the same dominance-based fitness and is therefore used as a tie-breaker. A popular algorithm in this category is NSGA-II [23].
2. Decomposition based MOEAs: these algorithms decompose the problem into several scalar (single-objective) optimization subproblems, each one focusing on different regions of the Pareto front. To create these subproblems, a scalarizing function is used as an objective function and different values of the weight vector are assigned to each subproblem. Then, one individual from the population is assigned to each subproblem, i.e. searches a different region of the objective space, and interactions among individuals (either due to crossover or offspring sharing) allow a collaborative search mechanism. MOEA/D [24] is a classical decomposition-based algorithm.
3. Indicator based MOEAs: these MOEAs are guided by an indicator that measures the performance or quality of an approximation set. The selection procedure depends directly on the contribution of each individual to this indicator, which allows establishing a ranking among individuals in order to select. A state-of-the-art example of such algorithms is the SMS-EMOA [25], which uses the hypervolume indicator to determine

the quality of an approximation set. This algorithm, which is employed in this work, is described in details in section *2.2.4*.

2.2.3. Different scalarizing technics

Scalarization (or utility) functions transform the MOP into a single-objective optimization problem through some kind of distance metric between each solution and a reference point (typically, the ideal point). These functions use a weight vector, defined by the user, to determine the priorities between objectives. In general, this weight vector is normalized. By varying the weight vector, different sub-regions of the objective space are explored, so that a uniformly distributed set of weights supposedly allows to describe the entire Pareto front of the MOP. A variety of scalarization functions have been developed, each with its own properties. A complete review of these functions is proposed in [26]. Four of them, particularly important for the present work, are presented below:

- **Weighted sum (WS):** is one of the most commonly used in multiobjective optimization. However, with this function it is not possible to find solutions in the concave regions of the Pareto front. Its expression is as follows:

$$u^{ws}(\vec{f};\,\vec{w}) = \sum_{s} w_s \cdot f_s$$

- **Augmented Chebyshev (ATCH):** this function is an extension of the Chebyshev function (TCH), which adapts to all types of front shapes. The augmented version corrects the defect of TCH of finding weakly dominated solutions through an additional term that allows it to discard these solutions. This term is weighted by a coefficient α and the authors of [26] recommend using a value between [0.001, 0.01]. It is defined as:

$$u^{atch}(\vec{f};\,\vec{w}) = \max_{s}\{w_s|f_s|\} + \,\alpha \sum_{s} |f_s|$$

- **Modified Chebyshev (MTCH):** is a variant of the TCH, formulated as:

$$u^{mtch}(\vec{f};\,\vec{w}) = \max_{s}\left\{w_s(|f_s| + \,\alpha \sum_{s} |f_s|)\right\}$$

For this function a small and positive α coefficient is also recommended as suggested in [26].

- **Augmented Achievement Scalarizing Function (AASF):** unlike the TCH function, the objectives are here divided by the weights. As ATCH, it is an augmented version through an additional term to discard weakly dominated solutions and the authors of [26] recommend using a value of the coefficient α ≈ 10-4. It is expressed as indicated below:

$$u^{aasf}(\vec{f};\,\vec{w}) = \max\left\{\frac{f_s}{w_s}\right\} + \,\alpha\,\sum_{s}\frac{f_s}{w_s}$$

2.2.4. The SMS-EMOA algorithm

As noted above, the SMS-EMOA is based on the optimization of a performance indicator (third paradigm). In particular, this algorithm seeks to maximize the hypervolume indicator.

The hypervolume is a widely used indicator in multi-objective optimization to evaluate the quality of a front. This indicator, denoted as HV, is calculated as the hyper-area dominated by a set of solutions and bounded by a reference point. Therefore, the larger the hypervolume, the better the produced approximation of the actual Pareto front. The coordinates of the reference point (in the objective space) are the worst values found within the solution set according to each objective, to which a small arbitrary quantity is added. The hypervolume is one of the most widely used indicators since it is the only one that is Pareto-compliant, i.e., it meets the definition of Pareto optimality in the sense that the maximum hypervolume is obtained only for the actual Pareto front. Furthermore, the hypervolume measures the simultaneous convergence of the solutions to the real Pareto front, as well as their extent and the uniformity of their distribution along the front, as illustrated in figure 2.

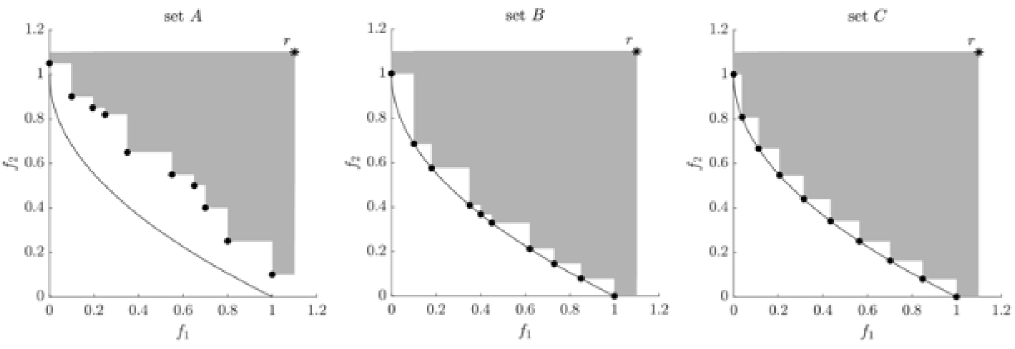


Figure 2. Hypervolume of three different sets A, B and C. Source:[1]

In the figure above, three solution sets, A, B and C, can be observed, corresponding to three approximations of the front of a classical bi-objective test function (both objectives are to be minimized). The gray colored area between the different solutions and the reference point *r* represents the hypervolume of each approximated front and the curve where the solutions converge to is the Pareto optimal (PF) front. The hypervolume values for the three sets are:

HV(A) = 0.5808, HV(B) = 0.8101 and HV(C) = 0.8237. In front A, the hypervolume penalizes the poor convergence to PF and, in front B, the poor dispersion along PF. The value of the C front is the greatest one because the solutions converged to PF and the solutions are uniformly distributed.

However, the main drawback of the hypervolume is the high cost associated with its computation. In particular, the complexity grows rapidly with the number of objectives. But, for the two-objective case of interest in this work, its computation is trivial and very efficient.

SMS-EMOA is a MOEA that from a population of μ individuals generates one new solution at each iteration, through crossover and mutation operators. Its working mode prevents from a deterioration of the hypervolume covered by the current population. This implies that new candidate solutions can only be integrated into the current population if their replacement of a current member allows increasing the hypervolume. Also, dominance is accounted for the formation of successive fronts, such as in NSGA-II [23]. Then, the rules that define when one individual is preferred over another are:

- A solution belonging to lower fronts (i.e., more likely to be non-dominated) is always preferred at any point in the evolutionary process. This rule is extracted from the popular algorithm NSGA-II [23].

- In case the first rule does not occur, the contribution to the hypervolume ∆S of each individual is measured as a criterion for selecting individuals from the last considered front (according to the non-dominating sorting procedure in NSGA-II). That is, the individual contributing the least to the hypervolume is discarded from the worst ranked front.

In the bi-objective case, $f = (f_1, \ f_2)$, the contribution to the hypervolume (ΔS) of a set of individuals $R = \{s_1, \dots, s_{|R|}\}$, with $|R| > 2$, is computed as:

$$\Delta S(s_i, R) = \big(f_1(s_{i+1}) - f_1(s_i)\big) \cdot \ \big(f_2(s_{2-1}) - f_2(s_i)\big) \qquad \forall\, n \ \in \{2, \dots, |R| - 1\}$$

Note that extreme points are excluded from this procedure. The SMS-EMOA only considers the possibility of substituting interior points and its calculation is only possible if non-dominated solutions are found in the set R.

The main drawback of SMS-EMOA is due to the computation of ∆S, which, as the hypervolume, is resource-consuming for more than two objectives, which undermines the algorithm's efficiency for many objectives or large sets [27].

2.2.5. SBX crossover operator

Like any crossover operator, this is a binary operator which, starting from two parent solutions x^1, x^2, generates two offspring solutions y^1, y^2, which integrate the characteristics of their parents according to the following expression:

$$y_i{}^1 = 0.5\left((1-\beta_i)\,x_i{}^1 + (1+\ \beta_i)\,x_i{}^2\right)$$

$$y_i{}^2 = 0.5\left((1+\beta_i)\,x_i{}^1 + (1-\ \beta_i)\,x_i{}^2\right)$$

Where $i \in \{1, \dots, n\}$ corresponds to the position of each of the parental decision variables in a population of n individuals, and β_i to a parameter calculated according to the following probability distribution function (polynomial):

$$\beta_i = \begin{cases} (2u)^{\frac{1}{\eta_c+1}}, & if\ u \leq 0.5 \\ (\frac{1}{2(1-u)})^{\frac{1}{\eta_c+1}}, & otherwise \end{cases}$$

Where u is a random number between 0 and 1, and η_c is a polynomial distribution index defined by the user. This operator is applied to each pair of parents to obtain the new offspring.

2.2.5.1. Polynomial Mutation

The polynomial mutation was proposed in the same work as the SBX crossover operator. A mutant is obtained by adding to the value of the variable under consideration a random perturbation with polynomial distribution, generated according to the following equation:

$$y_i^{1\prime} = \ y_i^1 + \left(x_i^{(U)} - x_i^{(L)}\right) \cdot\ \bar{\delta}_i$$

Where $x_i^{(L)}$ and $x_i^{(U)}$ are respectively the lower and upper bounds on variable x_i and $\bar{\delta}_i$ is the disturbance, computed as:

$$\bar{\delta}_i = \begin{cases} (2r_i)^{\frac{1}{\eta_m+1}} - 1, & if\ r_i \leq 0.5 \\ 1 -\ 2 \cdot (1-r_i)^{\frac{1}{\eta_m+1}}, & if\ r_i \geq 0.5 \end{cases}$$

Likewise, the mutation operator is applied to each variable with a probability equal to p_m = 1/*n* (where *n* is the number of decision variables).

3. Problem statement and existing solution schemes

The problem to be solved consists in designing a Hydrogen Supply Chain (HSC) capable of supplying the growing demand for hydrogen over a given territory (the case study is the French region of Midi-Pyrénées), with the objective of simultaneously minimizing the total costs and environmental impact. The solution strategy developed here is based on the strategy introduced in [1] and proposes modifications in view of improving the algorithm efficiency. In particular, this chapter focuses on the detailed description of the HSC design problem and on the definition of the mathematical bi-level optimization model proposed in [1].

3.1. Problem description

The problem takes into account a territory for which a hydrogen demand can be predicted in the future years. In order to meet this demand, an HSC must be designed considering both installation and operation aspects. In particular, the installation feature involves making decisions that affect the following areas: the selection of hydrogen generation technologies, the energy sources used for generation and the location and sizing of both production equipment and storage facilities. On the other hand, with regard to the operational aspect of the supply chain, the following must be determined: the production levels of the installed generation equipment, the means used to transport hydrogen from the generation plants to the customers and the distribution routes and flows of transported hydrogen. Each of these decision levels will have an impact on the economic and environmental cost of the HSC, so the optimization problem tackled here is bi-objective.

In more detail, to solve this problem, the territory is assumed to be divided into sub-regions, called grids, which can refer to cities, administrative districts, communities, etc., of the considered territory. This division implies a possible representation of the HSC as a graph, in which each grid represents a node (which constitutes a potential hydrogen production area with its own corresponding demand) and where the potential distribution routes are the edges. To account for demand growth over the lifetime interval of the HSC, this time horizon is divided into periods, within which the demand is considered to be static. The model is therefore of the multi-period type.

Regarding the installation issue, the technologies used for hydrogen generation have to be selected, taking into account their cost and environmental impact. In general, these parameters are influenced by the "maturity" of each technology, i.e., the experience accumulated over the years, which for example allows economic or performance improvements. In this work, the following potential generation technologies are considered: steam methane reforming (SMR), central and distributed electrolysis. In addition, and according to the chosen generation technologies, a choice has to be made between the different energy sources that can be used as feedstock: solar photovoltaic, wind, hydro, nuclear (as electricity from the grid) and natural gas. It is worth mentioning that, in case of needing to import the raw materials because they are not available in the territory, importation costs will also be taken into account. On the other hand, production plants can have different sizes, associated with production capacities. Therefore, a discrete (and reduced) number of sizes available for each type of technology is considered. Finally, the optimal location of production plants must also be determined, which will have an impact on distribution aspects.

At the operation level, decisions focus on production and transportation aspects. According to the sizes of the installed production equipment, the corresponding minimum/maximum production capacities can be known. The production rate should be determined between these two bounds. On the other hand, the transportation means between a production area and a consumption area have to be determined. In this work, the following options are considered: pipelines, tanker trucks or trucks with trailers. To use each of these transports, the hydrogen must be preconditioned in a specific way (gaseous or liquid form), which also conditions the storage technology. Storage facilities can be of different sizes, as can production facilities, and are deduced from the quantities produced/transported in each grid. Finally, the distribution routes between the hydrogen generation plants and the final customers (refueling stations for consumers) are to be established.

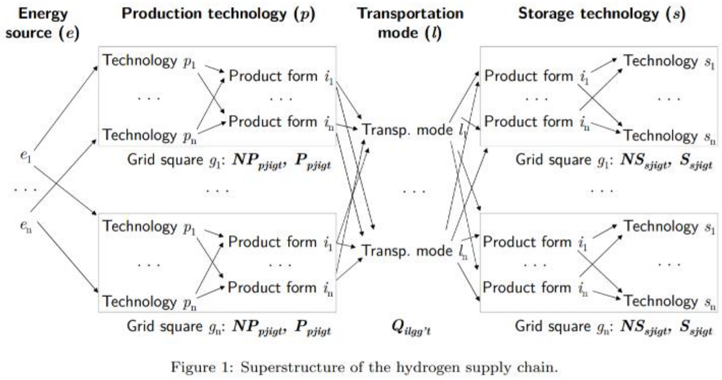

Energy source (e)
Production technology (p)
Transportation mode (l)
Storage technology (s)
Technology p_1
Technology p_n
Product form i_1
Product form i_n
e_1
e_n
Grid square g_1: NP_{pjigt}, P_{pjigt}
Grid square g_n: NP_{pjigt}, P_{pjigt}
Transp. mode l_1
Transp. mode l_n
$Q_{ilgg't}$
Technology s_1
Technology s_n
Grid square g_1: NS_{sjigt}, S_{sjigt}
Grid square g_n: NS_{sjigt}, S_{sjigt}
Figure 1: Superstructure of the hydrogen supply chain.

Figure 3. Problem diagram. Source: [1]

In conclusion, developing an HSC involves taking into account the above aspects and determining the set of most efficient solutions for the corresponding bi-objective optimization problem.

3.2. Mathematical model

The hypotheses and operational modes stated in the previous section have been formalized through different models of Mixed Integer Linear Programming (MLEP), in particular in the works of [13] and [19]. However, these models were reformulated in [1] as a bi-level mathematical programming problem, which takes advantage of the distinction between installation and operation issues to decompose the global problem into two sub-problems to reduce its original complexity. The first sub-problem (master, or upper-level) focuses on the installation aspects, while the slave sub-problem (lower-level) deals with the operation aspects (production and transportation). The same approach is used here to solve the overall HSC design problem, so the bi-level formulation is presented below:

$$\min_{x}[TDC(x), GWP(x)]^{T}$$

$$s.t. \quad \sum_{p\in P}\sum_{j\in J}\sum_{g\in G} PCap_{pji}^{min}\, NP_{pjigt} - \sum_{g\in G} D_{igt} \le 0 \qquad \forall i\ \in I, \forall t\ \in T$$

$$\sum_{g\in G} D_{igt} - \sum_{p\in P}\sum_{j\in J}\sum_{g\in G} PCap_{pji}^{max}\, NP_{pjigt} \le 0 \qquad \forall i\ \in I, \forall t\ \in T$$

$$\sum_{s\in S}\sum_{j\in J} SCap_{sji}^{min}\, NS_{sjigt} - S_{igt}^{T} \le 0 \qquad \forall i\ \in I, \forall g\ \in G, \forall t\ \in T$$

$$S_{igt}^{T}\sum_{s\in S}\sum_{j\in J} SCap_{sji}^{max}\, NS_{sjigt} \le 0 \qquad \forall i\ \in I, \forall g\ \in G, \forall t\ \in T$$

$$NP_{pjigt}, NS_{sjigt}\ \in \mathbb{N} \qquad \forall p\ \in P, \forall s\ \in S, \forall j\ \in J, \forall i\ \in I, \forall g\ \in G, \forall t\ \in T$$

$$\min_{y}[TDC(y), GWP(y)]^{T}$$

$$s.t. \quad PCap_{pji}^{min}\, NP_{pjigt} - \le 0 \qquad \forall p \in P, \forall j\ \in J, \forall i\ \in I, \forall g\ \in G, \forall t\ \in T$$

$$P_{pjigt} - PCap_{pji}^{max}\, NP_{pjigt} \le 0 \qquad \forall p \in P, \forall j\ \in J, \forall i\ \in I, \forall g\ \in G, \forall t\ \in T$$

$$\sum_{p\in P}\sum_{j\in J} P_{pjigt} - \sum_{l\ \in L}\sum_{g'\in G, g'\neq g} (Qilgg't - Qilg'gt) - Digt = 0 \qquad \forall i\ \in I, \forall g\ \in G, \forall t\ \in T$$

$$P_{pjigt}, Q_{ilgg'} \in\ \mathbb{R}_{\ge 0} \qquad \forall p \in P, \forall j\ \in J, \forall i\ \in I, \forall l\ \in L, \forall g\ \in G, \forall t\ \in T$$

Bi-level mathematical model of the HSC, for more details see [1].

As above-mentioned, the sub-problem of the upper-level (master) focuses on defining the structure of the HSC, which involves determining the type (generation technology), location and size of the production and storage equipment. The corresponding decision variables are $x = [NP_{pjigt}, NS_{sjigt}]$, both integers. NP_{pjigt} corresponds to the number of production facilities of type *p*, size *j*, which uses hydrogen in its physical form *i,* installed on the grid *g* during period *t*. Besides, the variable NS_{sjigt} corresponds to the number of storage facilities of type *s*, size *j*, which uses hydrogen in its physical form *i*, installed in grid *g* during period *t*. The constraints ensure that, with the installed production and storage capacity, the hydrogen required can be supplied in the whole territory and for each period. In the lower-level sub-problem (slave), the aim is to determine the production rate of the installed plants and the transport flows between grids, for the structure imposed at the upper-level. The corresponding decision variables are $y = [P_{pjigt}, Q_{ilgg't}]$, both continuous. P_{pjigt} corresponds to the production ratio in *kg·d*$^{-1}$ of the installations of type *p*, size *j*, which uses hydrogen in its physical form *i,* installed in grid *g* during period *t*. On the other hand, $Q_{ilgg't}$ corresponds to the amount of transported hydrogen in *kg·d*$^{-1}$ according to its physical condition *i*, through the transportation means *l*, from grid *g* to grid *g'* during period *t*. The constraints ensure that production capacities and demand requirements are respected in each grid, adjusting production levels and transported flows.

As observed, the variables of the sub-problem at the lower-level are not present in the sub-problem at the upper-level, i.e., the decisions made at the upper-level are independent of the lower-level. In contrast, the lower-level depends directly on the decision variables of the upper-level. Thus, an upper-level solution can be seen as a partially defined solution that can be complemented by solving the lower-level sub-problem: once the values of the upper-level decision variables are known, they are introduced into the slave problem which treats them as constants.

The objectives for both problems are conceptually the same ones:

- Total daily cost (TDC) of the supply chain, considering the investment costs related to plant and storage installation and transportation routes, operational costs for production, storage and transportation.
- Global warning potential (GWP), computed as a mathematical relation for gas emissions due to production, storage and transportation.

The difference between the two levels is that both the TDC and GWP are calculated only regarding the generation and storage equipment installed at the upper-level (TDC only depends on $x = [NP_{pjigt}, NS_{sjigt}]$); while they are calculated on the basis of production rates and transported flows at the lower-level (TDC depends only on $y = [P_{pjigt}, Q_{ilgg't}]$).

Thus, the bi-level reformulation introduced here results in two optimization sub-problems: the master sub-problem is MILP since it involves integer variables, while the slave sub-problem belongs to the Linear Programming (LP) class (since it involves only continuous variables). This observation motivates the use of optimization techniques adapted for each level, as proposed in [1]. This solution mechanism is described in the next subsection

3.3. Current resolution strategy for the HSC bi-level problem

The strategy used by the author of [1] to solve the problem consists of a hybrid approach, namely, a MOEA for the upper level coupled with a linear programming solver at the lower level. The iterative process is described as follows.

The upper level uses the algorithm SMS-EMOA (see section *2.2.4*) to determine the configuration of the HSC production and storage facilities. A population of μ individuals is evolved, each one defined by the variables NP_{pjigt} y NS_{sjigt} $\forall\, p, j, s, i, g, t$. These variables represent the technology, size, energy source used, location and opening period of the production and storage facilities, respectively. Due to the stochastic nature of the algorithm (initial population and variation operators), these solutions may not meet the constraints of the master problem, in particular those regarding the ability to supply the total demand of the territory considered. In this case, the solutions are repaired by randomly adding or removing facilities one by one, until the solution is feasible.

For its evaluation, each individual, i.e. each upper level partial solution, needs the resolution of the lower level sub-problem to determine the optimal values of inter-grid transportation flow and plant production rate (variables $Q_{ilgg't}$ y P_{pjigt}, respectively). The sub-problem is also bi-objective (minimizing the total costs and environmental impact associated with the operation of the SC) and it is solved exactly and efficiently through a LP solver and a scalarizing function to deal with both objectives (the author recommends the use of AASF, see section *2.2.3*, with a randomly generated vector of weights associated to each objective). Note, however, that the corresponding LP has to be solved for each time period, which may imply a great computational effort (since every partial solution of the upper-level, i.e. every individual of the used MOEA, requires solving this LP sub-problem). Thus, to simplify the problem, a heuristic is used in order to drastically reduce the number of variables: according to the values of the variables of the upper-level, NP_{pjigt}, only grids with an installed production capacity higher than their demand in a period are able to export, while those with a capacity lower than their demand can only import hydrogen. In this way, the number of potential transport flows decreases, which allows short computational times for solving each lower-level sub-problem.

Once the partial solution of the slave sub-problem has been calculated, the MOEA gets the value of the continuous variables P_{pjigt} y $Q_{ilgg't}$ of the slave problem, integrates them to the corresponding individual of the master sub-problem and computes the associated objectives (TDC and GWP). It is of particular relevance to this work that, in the original version of the algorithm, the lower-level sub-problem is solved only once for a single vector of weights, yielding a single partial solution for the upper-level. Thus, the SMS-EMOA algorithm can be used in its canonical form: through classical genetic operators (SBX crossover and polynomial mutation), offspring are generated. The combined population of parents (current generation) and children is composed of $2\cdot\mu$ individuals. The selection operator is that of SMS-EMOA: after generating successive fronts of non-dominated solutions and assigning a ranking to each individual, the next population is filled starting with the lowest ranked fronts. For the last front for which solutions are to be selected, the contribution of each individual to the hypervolume indicator is measured. The algorithm discards the individual with the lowest value for this contribution, repeating this process until the population is reduced to μ individuals. These generational stages are repeated iteratively, until the stopping criterion is reached, in this case a maximum number of generations of the evolutionary process (user-defined parameter).

Hybrid strategy procedure

```
initialize MOEA
while not terminate do
    generate offspring through variation operators
    for all individuals in population do
        for all t ∈ T do {for each period}
            if offspring solution violates upper level constrains then
                repair infeasible solution
            end if
            build LP problem (identify sink and source grids)
            solve transportation problem (LP solver)
        end for
    compute master problem's objective functions
    assign fitness value according to MOEA's working paradigm
    evolve population according to MOEA's working paradigm
    end for
end while
return current Pareto set approximation
```

Algorithm 1. Pseudocode of the hybrid algorithm developed in [1].

The results obtained through an experimentation carried out over different instances of the HSC design problem demonstrate that this algorithm is able to identify a set of solutions that have converged to the optimal front (they lie on the Pareto boundary) and well distributed along the front, in a single run [1]. Despite these satisfactory conclusions, the author of [1] himself insists in his work perspectives on the need to develop a method able to work with several solutions of the lower-level sub-problem associated to a partial solution of the upper-level. Indeed, since the LP sub-problem is also bi-objective, different solutions could be obtained using different weight vectors.

This is what is intended to do in the present work: to solve the LP sub-problem with several weight vectors and thus generate several lower-level solutions associated to a single upper-level partial solution; and, consequently, to modify the upper-level algorithm to adapt it to the operational mode of the former. The developed strategy is presented in the following section.

4. A new hybrid algorithm for the HSC design problem.

4.1. Proposed improvement strategy

The HSC design problem is a bi-objective optimization problem. In its bi-level formulation, both levels consider the same objectives (TDC and GWP), but for the lower-level it focuses on the operation sub-problem (production and transportation) corresponding to each individual defined at the upper-level, i.e., a configuration of production and storage facilities. Since the objectives are conflicting at both levels, the slave sub-problem admits several equivalent solutions, in the sense of non-dominated.

The above is illustrated in figure 4, where the installed production and storage infrastructures are shown in black and blue respectively (NP_{pjigt} and NS_{sjigt}), associated with a single upper-level individual. For this partial solution, two combinations of plant production level and transport flows (P_{pjigt} and $Q_{ilgg't}$) are examined, as illustrated by the arrows, whose thickness represents the amount of hydrogen transported and the color, the physical form in which it is transported. Considering their respective objectives, these two solutions to the slave sub-problem are non-dominated.

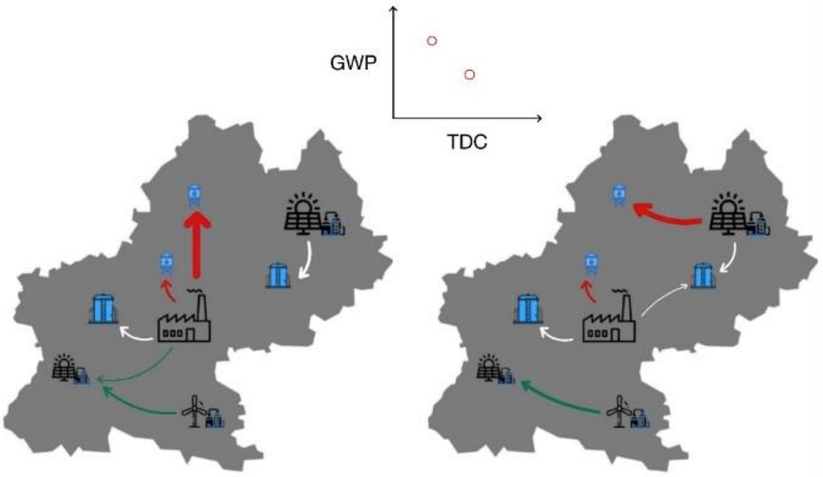


Figure 4. Example of two non-dominated solutions at the lower-level of an upper-level individual

Generalizing, one cannot say that there is only one optimal solution of the sub-problem at the lower-level, but rather a set of Pareto-optimal solutions, whose image in the objective space is a Pareto frontier, i.e., a sub-front associated to the “parent” upper-level partial solution. Thus, the optimal front of the global problem can be composed of different sub-fronts, each one corresponding to (part of) the Pareto-optimal frontier of an individual of the upper-level. This behavior is illustrated in figure 5.

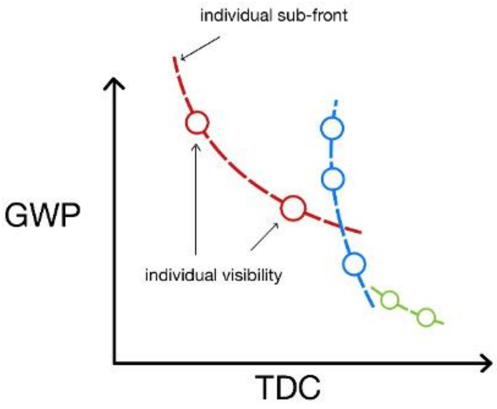


Figure 5. The problem front seen as a combination of sub-fronts associated with the upper-level solutions

Therefore, if a linear problem is solved for each individual using a scalarizing function with a single vector of weights, just as in the original algorithm presented in the previous section, only one of the sub-front points associated with an individual is known. This solution strategy can generate a loss of information regarding the individuals of the upper-level, since the selection of the best individuals by the evolutionary algorithm is made according to the point of the front obtained, without taking into account the sub-front as a whole.

For the above reasons, the aim of this work is to develop a new algorithm that generates the lower-level sub-front associated to each individual of the upper-level, through the repeated resolution of the same linear problem with a certain number of different weight vectors. The realization of this general objective raises several issues, denoted in what follows as particular objectives of the present work:

1. Perform a preliminary comparative study on different scalarizing functions to determine the most appropriate one to solve the lower-level sub-problem.

2. Propose an adaptation of the selection operator in the SMS-EMOA, since, in the new algorithm, the individuals are sub-fronts (composed of the lower-level PL solutions obtained

with different weight vectors) and not a single solution as in the original version. The comparison between individuals must therefore be adapted to this new operational mode.

3. Implement the proposed algorithm in a program to solve the HSC design problem.

4. Develop a strategy for a smart management of the weight vectors used for the generation of the sub-front associated to each individual, in order to reduce the number of PLs solved in case they are not all necessary.

5. Conduct an experimentation on several instances of the HSC design problem, to compare the two developed versions of the new algorithm (using all weight vectors or smart management) with the base algorithm proposed in [1].

4.2. Selecting the scalarizing function

In this project, an approximation of the entire sub-front associated with each individual (or rather, a representative discretization of this sub-front) is to be generated, using a defined set of weight vectors to solve the lower-level sub-problem. From this perspective, one of the important points is to determine the most appropriate scalarizing function to be optimized in the PL. In order to make a justified choice, preliminary experiments have been carried out with four different scalarizing functions. The functions tested are classical ones (see [26] for more details) and were described in section *2.2.3* : Weighted Sum (WS), Augmented Tchebychev (ATCH), Modified Tchebychev (MTCH) and Augmented Achievement Scalarizing Function (AASF). It should be recalled that, in the original algorithm [1], AASF is used.

The experimentation consisted in randomly generating a sample of $S_w \in$ {500, 2000, 3000, 5000} upper-level feasible individuals (as done for the initial MOEA population). For each individual, the corresponding linear problem is solved $\lambda_w \in$ {55, 10, 25, 50} times, i.e., using 5, 10, 15, or 50 different weight vectors. To generate the weight vectors, the Simplex Lattice Design (see [28]) is used, which becomes trivial in two dimensions.

This procedure is repeated with the four scalarizing functions, computed using the same set of weight vectors. In this way, with every scalarizing function, a sub-front composed of λ_w solutions is generated for each individual. Once the sub-fronts associated to all individuals with the different functions have been obtained, the results are analyzed using the hypervolume indicator as a criterion to measure the quality of each sub-front to compare the solutions obtained with each scalarizing function. In particular, for each function, the number of individuals obtaining the maximum hypervolume value (calculated using the λ_w weight vectors) are counted. For example, if, for individual #1, the results are as follows: HV(WS)= 0.368, HV(MTCH) = 0.46, HV(ATCH) = 0.447, HV(AASF) = 0.46, then this individual is counted for MTCH and AASF, which obtained the maximum hypervolume value. Based on these results, a ranking of the four functions is established, where that one with the highest number of individuals with maximum hypervolume obtains the best ranking. Then, the worst function according to this ranking is eliminated and the above procedure is repeated iteratively until only one scalarizing function remains and is selected as the best one.

This preliminary experimentation was carried out with only one small-size instance (the smallest one of those described in section 5), In order to avoid high computational times and assuming that the results obtained can be generalized to other instances.

4.2.1. Reference point for hypervolume computation

In order to perform a fair comparison of the four scalarizing functions considered, the hypervolume associated with each individual is computed with the same reference point, regardless of the function used. This implies, in a first step, normalizing the objectives. In this perspective, each single-objective problem is solved first (i.e., each objective is optimized separately, which is independent from any scalarizing function), making it possible to determine the coordinates of the ideal and Nadir points (denoted as $p_e{}^*$ y $p_e{}^{nad}$ for individual $e \,\epsilon\, \Omega$. $p_e{}^*$ contains the minimum values of the objectives encountered when solving the single-objective problems and $p_e{}^{nad}$ the corresponding values of the non-optimized objective. Then, each objective vector found for individual $e \,\epsilon\, \Omega$ is normalized according to the following formula:

$$p_e{}' = \frac{p_e - p_e{}^*}{p_e{}^{nad} - p_e{}^*}$$

The reference point chosen for the hypervolume calculation has the following coordinates (in the target space) $r_e^* = 1.1 \cdot p_e{}^{nad}$, as is recommended in the literature for many classical problems. These reference points are the same for each individual, regardless of the scalarizing function optimized.

4.2.2. Results and discussion

The number of individuals (among the S_w considered) for which the maximum hypervolume has been obtained by applying the scalarizing function to solve the λ_w LPs, as well as the corresponding ranking, are shown in table 1. As explained above, the ranking process is repeated several times to iteratively eliminate the functions with the worst results, which is reflected in the 3 "iterations" presented in the table.

Case	Iteration	AASF	WS	MTCH	ATCH
$S_1 = 1000$	1	516 (1)	265 (4)	292 (3)	445 (2)
$\lambda_1 = 50$	2	575 (1)	X	132 (3)	488 (2)
	3	579 (2)	X	X	616 (1)
$S_2 = 2000$	1	1084 (1)	657 (4)	669 (3)	870 (2)
$\lambda_2 = 25$	2	1242 (1)	X	796 (3)	1012 (2)
	3	1246 (2)	X	X	1278 (1)
$S_3 = 3000$	1	1600 (1)	1118 (4)	1144 (3)	1272 (2)
$\lambda_3 = 10$	2	1972 (1)	X	1461 (3)	1577 (2)
	3	1979 (2)	X	X	2022 (1)
$S_4 = 5000$	1	2690 (1)	2210 (2)	2083 (4)	2176 (3)
$\lambda_3 = 5$	2	2694 (2)	2216 (3)	X	2781 (1)
	3	3347 (2)	X	X	3516 (1)

Table 1. Number of individuals with the maximum hypervolume rating according to the scaling function and iteration.

For the parameter sets tested, the function that consistently obtains the best sub-fronts (in terms of hypervolume) is ATCH, but always with a very similar rating to AASF, regardless of the number of weight vectors used. Practically, half of the best results (i.e., individuals with maximum hypervolume) is obtained with ATCH and the other half with AASF. In contrast, WS and MTCH are never the best choice. Note that the sum of best solutions per iteration is not equal to the total number of individuals examined, because there are sub-fronts for which the hypervolume valuation is the same regardless of the scalarizing function used.

Although the ATCH is always the best rated, the fact that it is not clearly distinguished from the AASF as the best solution makes the choice non-trivial. Therefore, a second study has been carried out, for the same solution sets, now evaluating the differences between the solutions obtained with the ATCH and AASF functions.

4.2.2.1. Analysis of the differences with respect to the best hypervolume

Since the first result analysis highlighted the fact that the ATCH and AASF functions achieve a similar number of “maximum hypervolume individuals”, this second analysis focuses on the differences, for each individual, between the hypervolume obtained by a scalarizing function and the best hypervolume found (this difference is logically 0 when the function achieves the best hypervolume for an individual). The mean and standard deviation of these differences with respect to the maximum hypervolume are presented in table 2.

	AASF		ATCH	
Case	Mean	Std Dev	Mean	Std Dev
S_1 = 1000, λ_1 = 50	0,0405	0,0764	0,0413	0,0831
S_2 = 2000, λ_2 = 25	0,0620	0,1032	0,0500	0,0901
S_3 = 3000, λ_3 = 10	0,0993	0,1195	0,0857	0,0995
S_4 = 5000, λ_3 = 5	0,1718	0,1672	0,1444	0,1348

Table 2. Results of differences with respect to the best solution

The obtained results show that, except for case {S_1, λ_1}, the ATCH function consistently obtains differences that are lower than those obtained with AASF. This means that, when the sub-front identified by ATCH is not the best one (in terms of hypervolume), it lies anyway closer to the best approximation than it happens when using AASF. Indeed, this trend is even clearer when the population size increases. Simultaneously, the standard deviation of the ATCH sub-front hypervolumes is also lower than that achieved with AASF, except for case {S_1, λ_1}, highlighting the consistency of the better quality of the solutions found with ATCH. Finally, it can be noted that, in the only case for which AASF provides better results ({S_1, λ_1}) the differences between both functions are insignificant. Therefore, it can be concluded that the best scalarizing function for solving the lower-level sub-problem is the Augmented Chebyshev, which is used in the remainder of this work.

4.3. Adaptation of the upper-level MOEA selection operator

4.3.1. Context

The MOEA used here for the upper-level maintains the same structure as in the original work of [1]. In the evolutionary algorithm, the variables NP_{pjigt} and NS_{sjigt} are generated for each individual, on the basis of which the linear lower-level sub-problem can be solved, obtaining variables P_{pjigt} and $Q_{ilgg't}$, which are constant parameters for the upper-level. But, unlike the original algorithm, for every combination of variables NP_{pjigt} and NS_{sjigt} (individual) of the master problem, λ solutions are obtained here in the slave problem, where λ is a parameter set by the user to define the number of weight vectors used to approximate the lower-level sub-front. The resulting solutions are optimal for the lower-level PL, with the scalarization function selected in the previous section. Thus, the output of the lower-level consists of λ solutions associated to each of the μ individuals of the MOEA.

For ease of understanding, in the following, an individual refers to the production and storage infrastructure defined in the upper-level (as a partial solution). On the contrary, speaking of a solution refers to a complete solution, i.e., obtained by solving the slave sub-problem for the upper-level variables NP_{pjigt} and NS_{sjigt} associated with an individual and with a given weight vector. Therefore, in the remainder of this work, it will be considered that each individual has a set of corresponding solutions.

The above is visualized in figure 6, which shows the value of the objectives (TDC on the horizontal axis and GWP on the vertical axis) obtained for a population of μ = 4 individuals with λ = 20 different weight vectors. There should therefore be $4 \times 20 = 80$ solutions in total, denoted with dots in figure 6 (each color identifies the solutions of the same individual). It can be observed, however, that there are actually fewer than 80 distinct solutions. This behavior is due to the fact that several solutions, associated with the same individual but generated with different weight vectors, are identical (for example, in an extreme case, only one solution is obtained for the individual identified in pale violet in figure 6).

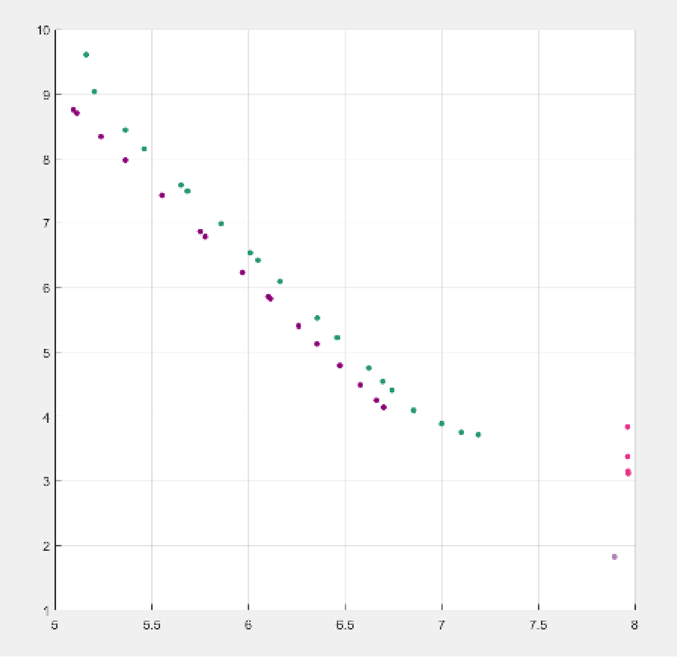


Figure 6. Value of the solutions obtained in the first generation from a sample of four individuals with 20 weight vectors each

This new working mode involves a remodeling of the selection operator governing the MOEA at the upper-level. The original algorithm [1] uses SMS-EMOA, but this algorithm only considers that each individual is equivalent to a (completely defined) solution whereas, in the present work, several solutions are taken into account for each individual. Therefore, a modification is proposed in what follows, considering each individual as a sub-front, i.e., a set of solutions, when selecting the best individuals. These changes are therefore concentrated in lines 10 (solution of the λ PLs of operation for each individual) and 13-14 (fitness assignment and generation of the next population) of Algorithm 1 (see page 30).

4.3.2. The proposed operator design

In particular, once evaluated the offspring individuals resulting from applying the genetic operators and computing, for each new individual, the *λ* solutions of the PL parameterized by different weight vectors, the combined population having 2·*μ* individuals (*μ* parents and *μ* offspring) should be reduced to *μ* individuals, selected for the next generation. Accordingly, it is necessary to design a selection criterion that maintains those individuals with the best genes in the population, making them able to transmit their genetic background to the next generation. It is worth mentioning that, to the best of our knowledge, there is no operator in the devoted literature for evaluating and selecting individuals characterized by several solutions. The new selection operator proposed in this work is based on the same SMS-EMOA paradigm, but it was adapted to allow the evaluation of an individual according to its set of associated solutions.

This new criterion consists of two steps. In the first step, the combined population {Parents}∪{Offspring} is divided into layers of equal dominance ranks, as in the popular NSGA-II genetic algorithm. Note that this non-dominated sorting procedure is carried out over solutions (and not over individuals, which have several solutions associated). Then, individuals are copied to the new population according to the rankings of its corresponding solutions. In a second step, for the individuals having solutions that belong to the last layer likely to be inserted into the next population, the contribution of these individual to the hypervolume indicator is used as a selection criterion to determine those individuals that are to survive. are discarded (thus following the SMS-EMOA paradigm). These two steps are explained in detail below. This explanation is illustrated with an example of 6 individuals, where 3 must be selected. At first, their distribution in the objective space is as shown in figure 7, where each color identifies the solutions corresponding to the same individual. Note that all individuals do not have the same number of solutions, since, as explained previously, some individuals may have fewer solutions than the number of weight vectors used for solving the LP sub-problem.

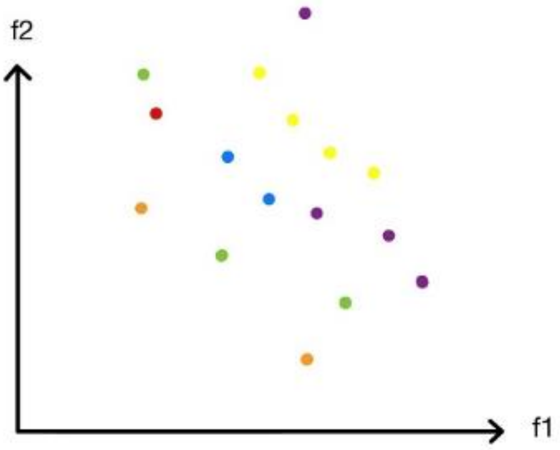


Figure 7. Initial population compose of 6 individuals

Step 1: First, all the solutions are classified according to the dominance operator, regardless of the individual to which they belong. The population is thus divided into different layers, where those of lower rank dominate those of higher rank. Each individual is assigned the ranking of its lowest-ranked solution (i.e., the the contribution of each individual to the hypervolume indicator of the lowest layer in which the individual has at least one solution). In this way, individuals with solutions in lower-ranked layers are ranked above (better) those with solutions in higher-ranked layers.

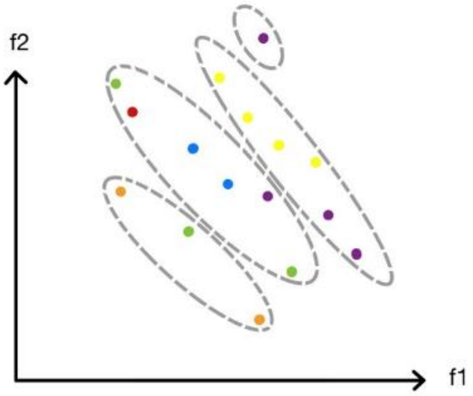


Figure 8. Segmentation of the solutions in non-dominated ranks

The ranking thus obtained constitutes the first rule of preference amongst individuals. Those individuals with solutions occupying a lower layer have preference over other individuals to survive and be selected for the next generation. Therefore, similarly to what happens in NSGA-II or SMS-EMOA, starting with the lowest layer (with rank *i* = 1), the individuals of the rank *i* are considered. Being n_i the number of individuals present on the layer i, if $\Sigma_{j=1,\ldots,i}\, n_j \leq \mu$, all the individuals of ranking i are selected and the process continue with the next layer, i+1. It is important to note here that, in layer i, there may exist solutions belonging to individuals with ranking $j < i$, which were already selected previously when examining lower-ranked layers. These individuals are not considered here, since they were already selected. Otherwise, if $\Sigma_{j=1,\ldots,i}\, n_j > \mu$, then the n_i individuals of ranking i do not all "fit" in the remaining space in the next generation's population. It is therefore necessary to select $\mu - \Sigma_{j=1,\ldots,i\text{-}1}\, n_j$ individuals of the n_i that have their best solutions in layer i, denoted as critical layer in the following. This is done in Step 2 of the selection process. In the example, as the first layer is composed of only two individuals, the critical layer is the one with ranking 2.

The above explanations can be illustrated with the example of figure 8. The first layer has solutions from the "orange" and "green" individuals, so that both these individuals are selected. Then, the first layer is discarded and the individuals having solutions in the second layer are considered. Only one individual should be selected (since two are already chosen), while four individuals have solution in this second layer, which is therefore the critical layer. Among this four individuals, one is the "green" individual, already selected and, thus, not considered anymore. The three remaining individuals in the critical layer are the "red", the "blue" and the "lilac" ones (see figure 9), from which only one should be selected, as explained in what follows.

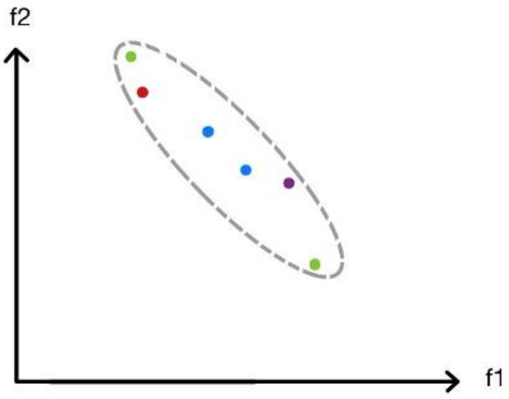


Figure 9. Critical layer

Step 2: In this step, a number of individuals is to be selected from those having ranking i (i.e, having their best solution in the critical layer i). This number is that necessary to complete the population of the following generation ($\mu - \Sigma_{j=1,\ldots,i\text{-}1}\, n_j$). To this end, and in agreement with the classic SMS-EMOA operating mode, those individuals who contribute the least to the total hypervolume of this critical layer are discarded here. Note that, in this critical layer, there may be solutions belonging to individuals with a ranking lower than i (i.e, these individuals have solutions in lower-ranked layers) and, therefore, have already been selected. The corresponding solutions in the critical layer *i* are anyway considered for the calculation of the total hypervolume of this layer *i* and of the contribution to the hypervolume of the solutions belonging to other individuals (of ranking *i*), which are candidates to be discarded.

Then, an iterative process that computes the contribution to the hypervolume of the solution(s) associated to each individual of rank *i* is performed. Since, for a same individual, there can be several solutions in the rank *i* layer, the total contribution of an individual is computed as the sum of the partial contributions due to all of its solutions that belong to the critical layer. Note that the contribution of each solution to the hypervolume is calculated as the area dominated by this solution and bounded by the areas dominated by its two neighboring solutions, as illustrated in figure 10. With only two objectives, the corresponding computational process is very efficient, as it only consists of ordering the layer solutions according to the first objective and, for each solution, calculating the area bounded by the area dominated by the two neighboring solutions. Therefore, there is no need for a reference point associated to the classical hypervolume computations.

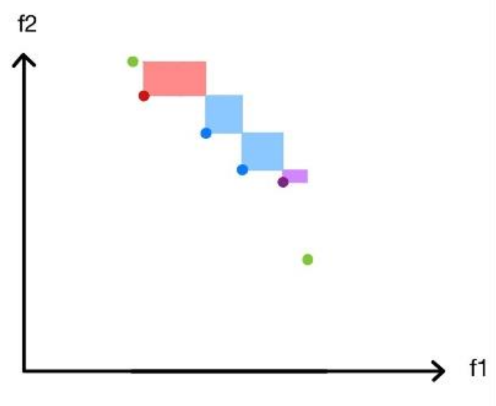


Figure 10. Contribution to the hypervolume of the critical layer per individual represented by colors

Also, it is worth highlighting the fact that the extreme points of the layer are exempted from this calculation: as in SMS-EMOA, these points are considered to have an infinite hypervolume contribution, to ensure that the corresponding individuals are selected. If the number of individuals to be selected from layer *i* is 2, then the individual(s) corresponding to these extreme points are automatically selected. If only one individual is to be selected from layer *i*, and the two endpoints correspond to two different individuals, then one of these two individuals is randomly selected.

Once all the contributions to the hypervolume of the individuals of ranking *i* have been computed, the individual with the lowest contribution value is discarded. In case of a tie, one is discarded at random. If, at this point, more individuals need to be discarded, the process is repeated iteratively (recalculating the contributions of each individual) until the required number of individuals remains. Thus, by integrating this new selection operator, the fitness assignment and selection parts of the original algorithm are modified (lines 13-14 of the Algorithm 1, presented in section *3.3*).

4.3.3. Parents population selection

In the same way as for the canonical version of SMS-EMOA, the environmental selection operator (which determines the survival of individuals from one generation to the next) has its consequence in the selection of the parents subsequently used for the recombination (crossover) process. Here as in NSGA-II, the selection of parents is performed by a binary tournament: a pair of individuals is chosen randomly and then compared to select the individual that will act as a parent. The comparison criterion follows the same logic as the selection operator: first, it prefers individuals having a solution in lower dominance layers. Otherwise (both individuals have the same ranking), they are compared according to the second criterion, i.e., the contribution of each individual to the hypervolume of the layer. It is worth mentioning that this contribution value was not necessarily calculated in previous iterations (when the layer considered is not the critical one). In these cases, the contribution to hypervolume is calculated following the indications of Step 2 described in the previous section. For each instance of the crossover process, this tournament is repeated twice to select two parents. Once the parents are selected, the next generation is generated by applying the SBX crossover and polynomial mutation (see section *2.2.5*).

4.4. Implementation of the algorithm

The above-described algorithm was implemented in MATLAB. The code development first involved reading the input data defining the specifications of the HSC design problem. These datasets, called instances, include the following parameters:

- The different options available regarding: energy sources for production, type of production facility, means of transportation and type of storage facility.
- The available production and storage plant sizes.
- The economic and environmental costs associated with each of the options in each period.
- The number of periods considered.
- The number of grids into which the territory is divided and their corresponding hydrogen demand for each period.

Different datasets have been used to test the proposed algorithm (see Appendix A). Depending on the number of input variables considered in each instance, their resolution complexity will be different. Also, the working parameters of the algorithm have to be defined, in particular:

- The population size (μ).
- The number of weight vectors (λ), whose optimal value can vary depending on the treated instance.
- The crossover (η_c) and mutation (η_m) distribution indexes. In this work, $\eta_c = 20$ and $\eta_m = 20$, as recommended by the authors of [29].
- The stopping criterion (lp_{max}). Since the computational time is almost entirely concentrated by the solution of the lower-level LP, it has been decided that the number of calls to the LP solver is the stopping criterion (rather than the number of generations).
- The scalarizing function of the lower-level sub-problem: being consistent with the results obtained in section *4.2*, the ATCH scalarizing function is chosen.

Once the problem and algorithm parameters have been defined, the algorithm designed in the previous sections is implemented. The architecture of the computer program consists of a main function that requires the above-described necessary inputs and the name of the problem instance to be solved, which is imported from an external file. This main function establishes the order of execution of the algorithm, which consists of calling various sub-functions that create and alter the population, and also collects the data associated with the population at each moment. In particular:

1. A sub-function initializes the algorithm, which includes the generation of a population of individuals; repairing those individuals that do not meet the upper-level constraints; the creation of a set of weight vectors to be used for solving the LP lower-level sub-problems.
2. A sub-function that computes the TDC and GWP objectives for a (complete solution), i.e., including the lower-level variables.
3. A sub-function performing the genetic variation (crossover and mutation operators).
4. A sub-function generating the LP sub-problem associated to an upper-level individual and a specified weight vector.
5. A sub-function implementing the selection operator proposed in this work.

At the end of an execution, the program reports the value of the decision variables and objective functions of each individual in the last population (corresponding to the generation when lp_{max} is reached). In addition, further data associated with each generation are also saved, in order to analyze the any-time performance of the algorithm, such as the number of LP calls of the sub-problem at the lower-level up to that instant. In particular, the hypervolume of the front generated by the non-dominated solutions in each population is reported. Please note that the hypervolume computation is performed considering a fixed reference point, in order to be able to fairly evaluate the dynamics of the hypervolume indicator evolution during the evolutionary process. This reference point was obtained from the single-objective solution of the instance with CPLEX for each objective.

4.5. Smart management of weight vectors

4.5.1. Weight generation

In the first experiments, the same weight vectors were used for all the individuals that integrated λ values uniformly distributed between 0 and 1. This experimentation revealed two issues. The first one is that due to the similarity between different individuals, regions of highly concentrated solutions appeared in the objective space, forming staggered structures (see figure 11). The second is that since the weight vectors are limited to specific values, weight vectors that could provide better information may be omitted.

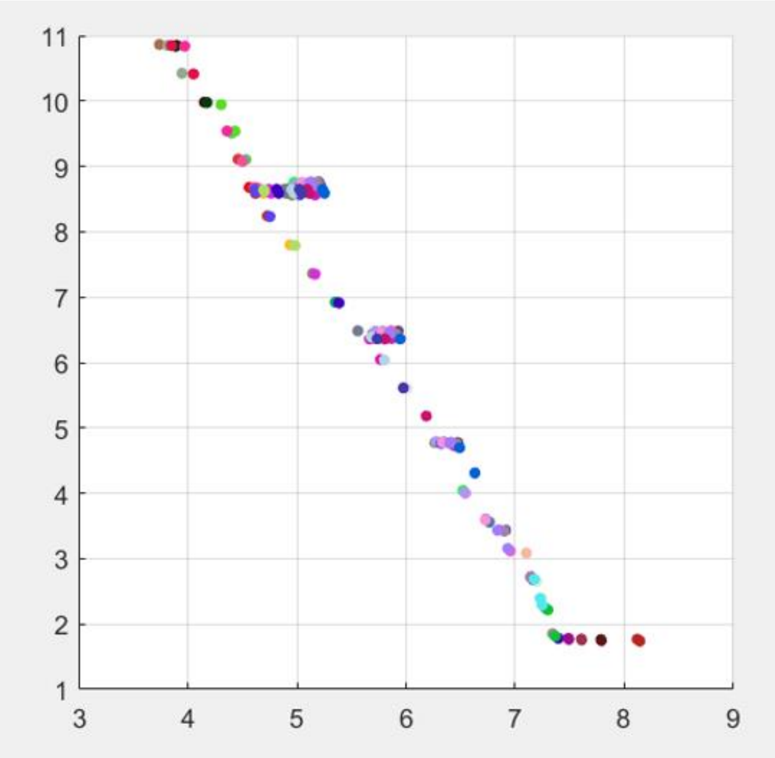


Figure 11. Population distribution after executing the program with the same weight vectors to all individuals

This results in a loss of information about how individuals are actually distributed, which may lead to inefficient selection in future generations. To avoid this issue, the weight vectors were generated including a random component: the segment [0, 1] is divided into *λ* uniform intervals and one random value is randomly generated within each interval. These values constitute the first element of the weight vectors, while the second one is trivially deduced (since the problem has only two objectives).

This process allows maintaining a “globally uniform” distribution of the weight vectors but introduces a random component which showed empirically its validity for solving the above-mentioned issue. Indeed, this new approach corrects the accumulation of points in certain areas of the objective space, as can be seen in figure 12, which shows the same case as in figure 11 but with the new weight vector generation technique.

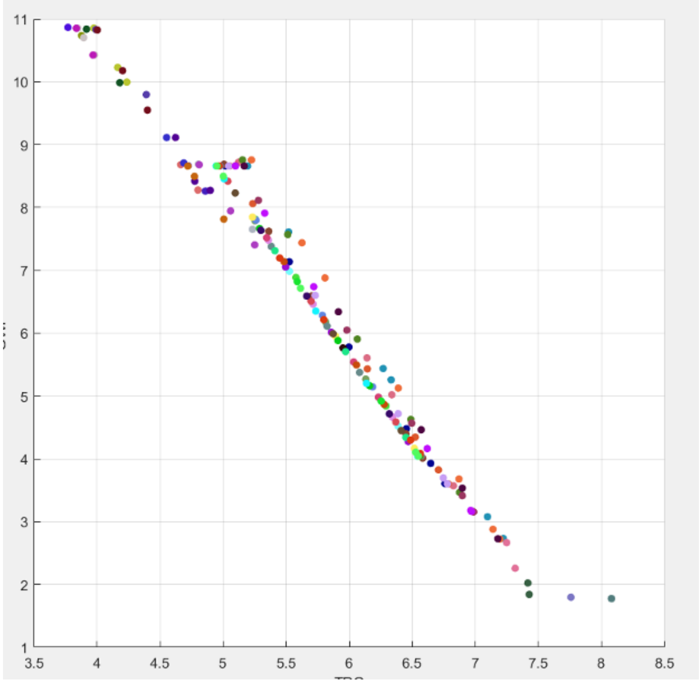


Figure 12. Population distribution after executing the randomized weight vector generation approach

4.5.2. Weight selection

As already mentioned in section *4.3.1*, the fact that *λ* weight vectors are assigned to an individual does not imply that *λ* distinct solutions are obtained by the solution of the *λ* PLs with different parametrizations. Two different weight vectors may indeed result in the same solution. Actually, this observation turned out to be a recurring situation for all the treated instances of the HSC design problem. In some cases, the number of distinct solutions obtained for an individual is lower than *λ* /3, meaning that more than 60% of the resources used to solve the *λ* sub-problems were useless. This fact is especially critical, since the resolution of the LP (solution generation) accounts for more than 90% of the computational time of the algorithm. Therefore, it seemed necessary to design a heuristic for weight vector selection, to reduce the computation of repeated solutions.

A detailed observation of the solutions obtained for each weight vector showed that the non-repeated solutions are generated only within a continuous interval of the first element of the weight vector. The remainder of the solutions, obtained for extreme values of the weight vectors, are repeated solutions. For instance, if w_1 is the first element of the weight vector, distinct solutions are obtained when $w_1 \in [0.4, 0.8]$, while the solutions obtained for $w_1 \in [0, 0.4]$ are identical and so are the solutions obtained for $w_1 \in [0.8, 1]$. Thus, the procedure proposed here first solves the LP sub-problems corresponding to the two extreme weight vectors ($[0,1]^T$ and $[1,0]^T$). If the solutions obtained are identical, there is no need to try with other weight vectors (the individual has only one associated solution). Otherwise, for a weight vector $[w_1^0, w_2^0]^T$ such that $0<w_1^0,w_2^0<1$, if the obtained solution is the same as that found for an extreme weight vector (for instance, $[0,1]^T$), then there is no need to further explore the interval bounded by these weight vectors (in our example, there is no need to perform LP optimizations for $0 \le w_1 \le w_1^0$ (equivalent to $w_2^0 \le w_2 \le 1$) since the same solution will be repeatedly found. On the other hand, if the solution found for $[w_1^0, w_2^0]^T$ is different from that found for an extreme weight vector, it is worth further generating intermediate weight vectors in order to (maybe) identify new solutions of the LP sub-front.

According to the above strategy, the following heuristic is proposed and applied to each individual, when $\lambda > 2$:

1. A user-defined number of candidate weight vectors is generated as detailed in *4.5.1*, obtaining a matrix, $W_{\lambda x2}$ that stores the values of the weight vectors.
2. The LP of the lower-level linear sub-problem is solved for the vectors of extreme weights ($[W_{1,1}, W_{1,2}]^T$ and $[W_{\lambda,1}, W_{\lambda,2}]^T$). If the extreme values of the objective functions coincide, this means that all the points will yield the same solution, therefore, the solution obtained is copied to the $\lambda - 2$ remaining solutions, the process is stopped and proceeds with the following individual. If, on the other hand, the two solutions found are not equal, the variables $a = 1$ and $b = \lambda$ are defined and the process continues to step 3.
3. The weight vector $[W_{c,1}, W_{c,2}]^T$, such as $c = ceil\left(\frac{a+b}{2}\right)$, is selected. If the value of the objective function is the same as that of the extreme solution found with $[W_{1,1}, W_{1,2}]^T$, go to step 3a. Otherwise, if the solution found is the same as the extreme solution found with $[W_{\lambda,1}, W_{\lambda,2}]^T$, go to step 3b. Finally, if the solution found for $[W_{c,1}, W_{c,2}]^T$ is different from both extreme solutions, set $r = c + 1$ and go to step 4.

   a. The value of the variables in position c is copied to those in the position between $a + 1$ and $c - 1$, a is updated such as $a = c$. Return to step 3.
   b. The value of the variables in position c is copied to those in the position between $c + 1$ and $b - 1$, b is updated such as $b = c$. Return to step 3.
4. The LP is solved for the weight vector at position r. If the value obtained is different from the value of solution b, this step is repeated with $r = r + 1$. If, reversely, the value coincides with b, set $l = c - 1$ and go to next 5.
5. The LP is solved for the weight vector at position i. If the value obtained is different from the value of solution a, this step is repeated with $l = l - 1$. If, reversely, the value coincides with a, the iterative process stops and moves on to the next individual.

Therefore, this heuristic may allow to save a significantly high number of calls to the LP solver when $\lambda > 2$ and will be particularly relevant when λ increases, as shown in the next section.

By computing the algorithm with this heuristic, the unnecessary calls to the LP solver are greatly reduced, avoiding the need to perform calculations in situations where the solver would yield a repeated solution. It starts to be effective when $\lambda = 3$, and its effectiveness is immediate. For $\lambda = 3$ it reduces the total number of calls to the linear problem by 15% per generation compared to a version where the heuristic is not applied; as the number of weight vectors increases, this value increases logarithmically, reaching a reduction in the number of calls to the LP solver of more than 70% for values of lambda greater than 20.

Due to the clear improvement in computational performance when using this heuristic, it is incorporated in the final algorithm and used in the experimentation of block *5*.

5. Experimental computations

In this section the numerical experiments performed to analyze the performance levels of the proposed algorithm are presented. First, the instances and parameter settings are defined in section *5.1*. Then, global results are provided in section 5.2 and finally, an any-time analysis is proposed in section *5.3*.

5.1. Experimental methodology

Once the algorithm has been developed, implementing the new functionalities such as the reconditioning of the upper-level selection process, the creation strategy of weight vectors and the selection of ATCH as the scalarizing function to be optimized at the lower-level, a numerical experimentation was carried out to provide results for analysis, comparison and conclusions.

The objective of this experimentation has been to check if the new algorithm allows for an improvement over the original one [1] and, if so, to determine the optimal number of weight vectors that should be associated to each individual. The HSC design problem has been solved for five instances: HSC08g001p, HSC08g01p and HSCg22g01p (single-period) and HSC08g04p, HSC08g07p that contemplates 4 and 7 periods (see Appendix A). All instances have been solved considering a population of μ = 100 individuals (as in [1]). Six numbers of weight vectors were tested, these being $\lambda_e = \{1, 2, 3, 5, 11, 19\}$. To account for the stochastic nature of the solution technique (due to the MOEA at the upper-level), each instance has been solved nine times for each number of weight vectors. The case $\lambda_1 = 1$ is equivalent to the original algorithm, so its results have been used as a basis for comparing whether the modifications provided an improvement. The stopping criterion was the same for all instances, i.e. $100.000 \cdot T$ calls to the subproblem, where T is the number of periods of the instance. This number has been taken from [1] where it is considered to be a sufficient number to ensure convergence to the Pareto front.

It is worth mentioning that maintaining the stopping criterion means that the total number of generations is considerably reduced each time the number of weight vector is increased, which has the drawback of reducing the genetic mixing among individuals, allowed by the crossover operator of the MOEA. On the other hand, the use of more weight vectors will give more comprehensive information regarding the sub-fronts of the individuals. In order to evaluate this relationship, the evolution of the population has been followed through the dynamic output parameters described in section *4.4*, analyzing the hypervolume generated by the non-dominated solutions of each generation and keeping track of the number of calls to the LP solver at each generation. When making comparisons using the hypervolume indicator, it is desirable that the number of points on each approximated front is the same. Thus, to ensure fair comparisons the hypervolume calculation has been performed on a reduced set of 21 solutions. The selection of these sub-sets is obtained by applying a heuristic that selects those points that are best distributed along the non-dominated solution front of a population, selected by iteratively discarding one by one those that contribute the least to the hypervolume until the 21 solutions are reached.

5.2. Global Results

To analyze the quality of the final fronts obtained for each instance and the impact of varying the number of weight vectors used to solve the lower-level sub-problem, table 3 shows the average and the standard deviation (over the of the nine executions performed in each case), of the value of the hypervolume of the last population, computed after reducing the solutions of the front to 100. The best mean value obtained for each instance is underlined.

Instancia	$\lambda = 1$	$\lambda = 2$	$\lambda = 3$	$\lambda = 5$	$\lambda = 11$	$\lambda = 19$
HSC08g001p	0,9846 (0,0010)	0,9846 (0,0004)	0,9848 (0,0002)	0,9843 (0,0011)	0,9844 (0,0003)	0,9844 (0,0004)
HSC08g01p	0,9905 (0,0012)	0,9911 (0,0010)	0,9911 (0,0008)	0,9903 (0,0026)	0,9896 (0,0026)	0,9899 (0,0011)
HSC08g04p	0,7781 (0,0063)	0,7656 (0,0055)	0,7544 (0,0087)	0,7471 (0,0073)	0,7447 0,0045	0,7470 (0,0052)
HSC22g01p	0,9869 (0,0021)	0,9862 (0,0098)	0,9846 (0,0027)	0,9840 (0,0007)	0,9827 (0,0020)	0,9823 (0,0020)
HSC08g07p	0,7871 (0,0019)	0,7862 (0,0013)	0,7832 (0,0027)	0,7806 (0,0036)	0,7817 (0,0012)	0,7799 (0,0027)

Table 3. Hypervolumes of the latest generation for a population reduced to 100 solutions

These results demonstrate that, for the two instances of lower complexity (HSC08g01p and HSC08g01p), the best hypervolume averages are obtained with λ equal to 2 or 3 weight vectors. However, the differences among the different values of λ seems insignificant. On the other hand, for the more complex instances, the best results are obtained with $\lambda = 1$ (original case). In these cases, the differences between different number of weight vectors increase and consistently, it can be observed that the greater the number of weight vectors used, the worse the hypervolume of the final front. Therefore, these global results seem to indicate that increasing the number of weight vectors does not provide benefits, even though values such as $\lambda =$,2 or three obtains very good results, close to those found when using one single weight vector. In what follows, an any-time analysis is provided to verify if these conclusions are true during the whole search process.

5.3. Any-time analysis

This study following consists in analyzing, for the five instances, the evolution of the hypervolume value in each generation. The result obtained for the HSC08g01p instance is illustrated in figure 13. It can be considered as a representative example of the trends obtained for the rest of the instances.

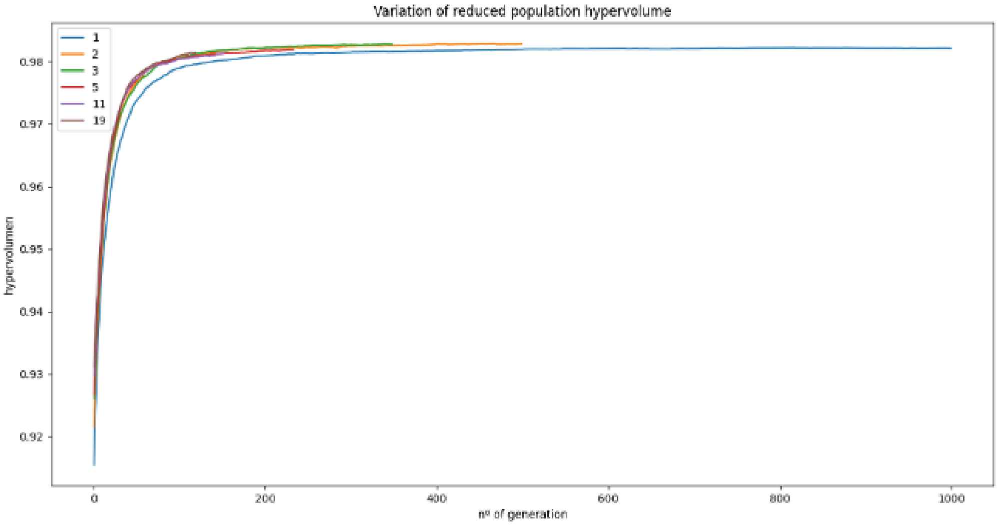


Figure 13. Variation of the hypervolume for the reduced population in the instance HSC08g01p

In this figure, the curves do not have the same length since, when using higher numbers of weight vectors, the number of generations computed to reach the stopping criterion (which is the number of calls to the LP solver) evolves in an inversely proportional manner. For this instance, all hypervolumes converge to similar values (as already mentioned previously), but larger differences between curves can be observed in the first generations of the algorithm. In order to have a better visibility of this period of the search process, the evolution of the hypervolume between generations [1 and 200] has been plotted for each instance in figure 14.

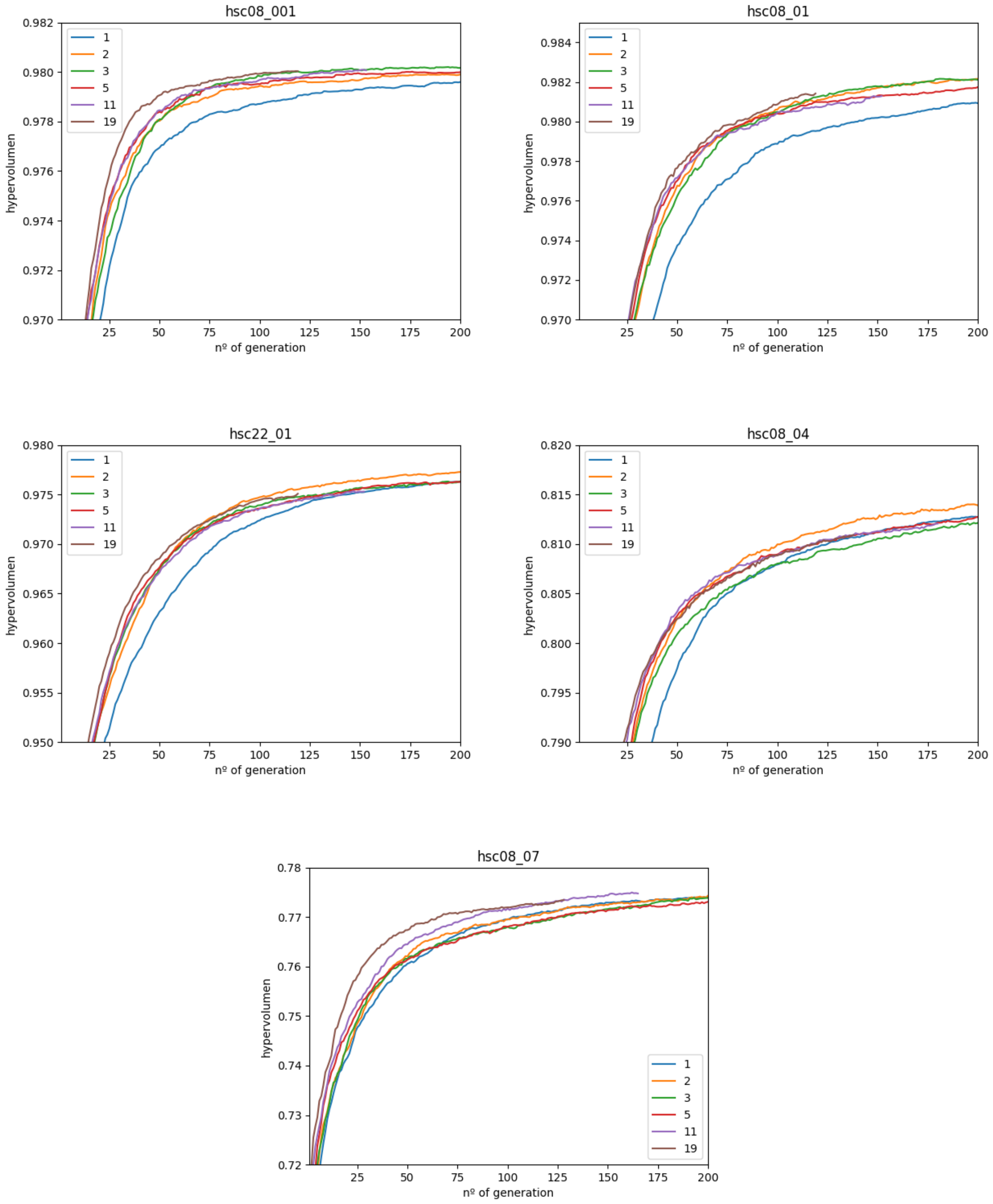


Figure 14. Evolution of the hypervolumes between generations 1 and 200

In all instances it is observed that, independently from the value of λ, all the curves show similar trends: the hypervolume grows quickly and, after a point close to generation 50, the increase is steadier. Before reaching that point, there is a clear difference can be observed among the curves corresponding to $\lambda > 2$ when compared to the trend when $\lambda = 1$. More in details, for the smallest instances, during this first period of the search, the best hypervolume values are obtained if λ is high. Subsequently, for a number of generations between 50 and 200, the best option clearly seems to be $\lambda = 2$, as illustrated in figure 15.

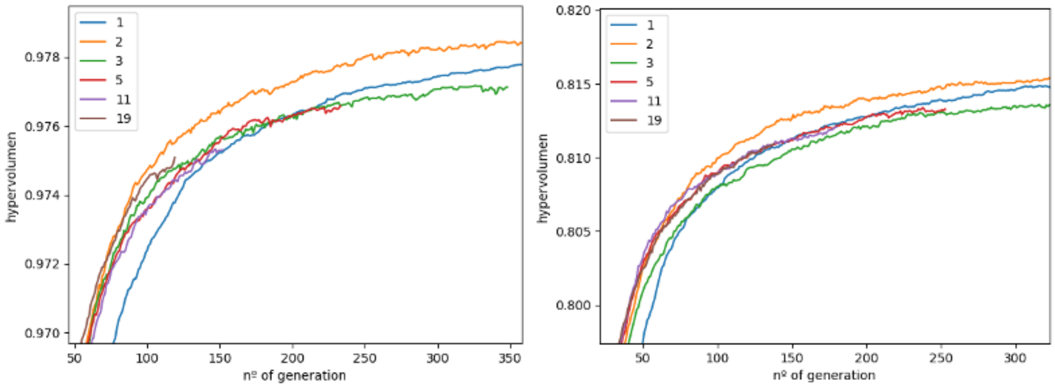


Figure 15. Search process after 50 generations (HSC22g01p on left and HSC08g04p on right)

During this second period of the search (between 50 and 200 generations) the results obtained with the other values $\lambda \geq 5$, are very similar, resulting in almost indistinguishable curves. This observation suggests that, from five weight vectors per individual, the extra information obtained from the sub-fronts do not generate significant differences to improve the search process. Also, as stressed when analyzing the global results, high values of λ do not allow the algorithm to converge to the optimal Pareto front for the instances of higher complexity.

In order to provide further insights regarding the behavior of the algorithm according to the value of λ, the variation of the hypervolume according to the computational resources allocated, i.e., the number of iterations or calls to the LP solver, is studied in what follows. In this perspective, the final number of iterations (the stopping criterion is 100,000 iterations) has been divided into 11 instants, indicated in the headers of table 4, where *T* is the number of periods of the instance. At each instant, the average value of the hypervolume for each value of λ is used to establish a ranking among these different parameter settings and the results are presented in table 4 where the value noted in each cell corresponds to the number of weight vectors for which the hypervolume is maximum at that instant.

The results obtained show that, at the beginning of the search the best results are always obtained with $\lambda = 1$. However, for the first two instances, after approximately 30,000 iterations, the best hypervolumes using $\lambda = 2$ or $\lambda = 3$, which is consistent with the results presented in section *5.2*.

	5e3·T	1e4 · T	2e4 · T	3e4 · T	4e4 · T	5e4 · T	6e4 · T	7e4 · T	8e4 · T	9e4 · T	1e5 · T
HSC08g001p	1	1	1	3	1	3	3	3	3	3	3
HSC08g01p	1	1	1	2	2	2	2	2	2	2	2
HSC08g04p	1	1	1	1	1	1	1	1	1	1	1
HSC22g01p	1	1	1	1	1	1	1	1	1	1	1
HSC08g07p	1	1	1	1	1	1	1	1	1	1	1

Table 4. Best rated case at each time and instance

Thus, from the above study, it can be concluded that the best results are consistently obtained $\lambda = 1$. However, it can also be observed that there are other competitive number of weight vectors ($\lambda = 2$ and $\lambda = 3$), in particular for small instances, but the differences with $\lambda = 1$ do not seem significant. On the other hand, the any-time analysis show that, at the beginning of the search, the intensive search procedures performed at the lower-level (with more weight vectors) allows for quick improvements of the hypervolume, but this trend is compensated by the genetic mixing produced during the last part of the evolutionary process, when fewer weight vectors are used.

6. Conclusions

This master thesis had the clear objective of developing a novel hybrid algorithm for bi-level bi-objective optimization and to evaluate its performance level through the application to the problem of hydrogen supply chain deployment and design. In order to achieve this, the structure of the solution strategy presented recently by Cantú et al. [1], has been used as a starting point and enriched through a new approach based on a finer evaluation of the sub-front corresponding to each lower-level sub-problem associated to the upper-level variables. In this perspective, a higher number of weight vectors are used for solving the lower-level sub-problems, aiming to provide a greater amount of information to improve the selection of future generations in the evolutionary process. As a consequence, the MOEA used at the upper-level had to be significantly modified to enable the algorithm to handle individuals constituted by sub-fronts and not single solutions.

During the development of the project and when carrying out preliminary experiments, some difficulties have appeared, such as not uniform distributions when using non-randomized weight vectors, or obtaining repeated solutions for the same individual in the lower-level sub-problem when too much weight vectors were computed. These early observations led to introduce randomness in the generation of the weight vectors associated to each individual, and to design a heuristic process selecting the worthy weight vectors in a smart way, improving the efficiency of the global algorithm.

From the design and experimentation performed over with different instances of the HSC design problem, the following conclusions can be drawn.

1- The comparative study of the sub-fronts generated by the different scalarizing functions has shown that the best alternative is the Augmented Chebyshev (ATCH) function, slightly better than the Augmented Achievement Scalarizing Function (AASF) proposed in [1] and, therefore, subsequently used in the algorithm developed here.

2- The proposition first of a random selection criterion of the weights of the individuals and then the intelligent management in the selection of which weights should be used to obtain the different solutions are crucial for the efficient construction of the sub-fronts and the competitiveness of the algorithm, especially for large number of weight vectors.

3- No evidence has been obtained, with the selection operator proposed in this work, regarding the benefits of working with more weight vectors at the lower-level. In other words, getting a more complete information on the sub-fronts associated to each combination of upper-level variables did not lead to a more efficient genetic selection within the MOEA used at the upper-level.

4- In the most complex instances, using more than two weight vectors and equal resource usage (calls to the LP solver) can even deteriorate the convergence to the true Pareto front. Note, however, that convergence is observed in any case for equal numbers of generation.

5- The latter indicates that the genetic mixing produced by letting the population evolve a during a greater number of generations (with 1 or a small number of weight vectors) outperforms the more accurate description of lower-level sub-front (with many weight vectors) but over a shorter evolutionary process.

In summary, when evaluating the algorithm according to computational resources, the original algorithm proposed in [1] (using a single weight vector) is consistently more efficient for obtaining a good approximation of the Pareto front. However, these conclusions should be balanced considering two observations. First, the obtained results are valid for the particular formulation of the HSC design problem treated here. It would be worth evaluating the proposed solution strategy over a wider range of problems, in order to confirm the above-mentioned conclusions regarding the use of one or several weight vectors. On the other hand, the new selection procedure of the MOEA, designed and developed in this work, might be seen as a first attempt and it may, for some reason, be responsible for the adverse results found in the present study. It is reasonable to think that the design of this new operator could be improved in order to overcome possible drawbacks.

Accordingly, both these observations are likely to constitute perspectives for future work, i.e.: (1) extending the application of the proposed algorithm to different bi-level bi-objective optimization problems and (2) modifying the selection operator of the upper-level MOEA, in order to determine if improvements can be obtained, in the perspective of a world where hydrogen is getting closer to become an environmental-friendly and economically viable alternative to conventional energy source.

Acknowledgments

To Catherine Azzaro-Pantel for the opportunity she has given me to work in the research carried out by the "Laboratoire de Génie Chimique" of ENSIACET, integrating me in a team developing leading research in the field of "Hydrogen Supply Chain".

To Antonin Ponsich for welcoming me in the team, giving me his time, guiding me in the development of the research, solving my doubts and motivating me.

To Víctor Hugo Cantú for his time, his patience, his interest in my work and his generosity in sharing his research with me.

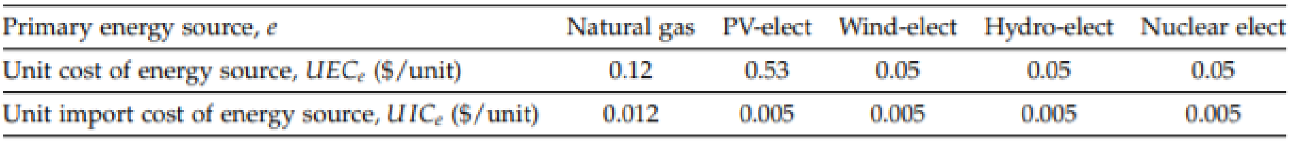
I would like to thank all of them for helping to foster in me the desire to deepen my research. The work I present would not have been possible without their help.

Finally, I would like to thank my family, who have always advised me and supported me throughout the project.

Appendix A. Data instances

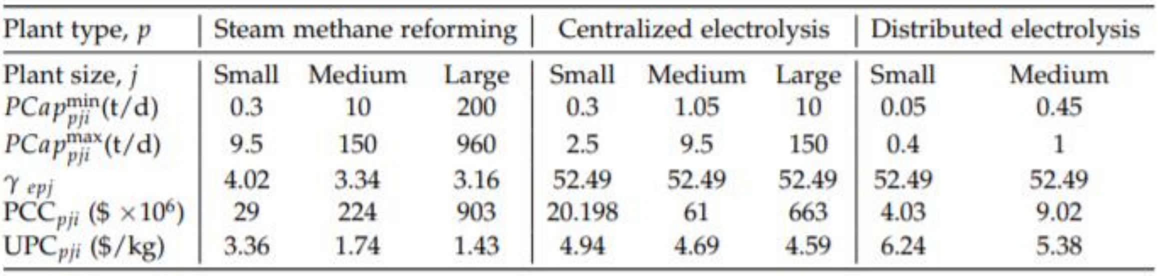
The data instances used are the same as those proposed in the paper [1], with the exception of HSC08g001p which was not included in that paper. Differs from the other instances because only three types of production technology, three possible sizes of production facilities, two sizes of storage facilities and three types of primary energy forms are considered as opposed to nine production technologies, three possible sizes of production facilities, four sizes of storage facilities and five different primary energy sources considered in the other instances.

Table A.1: costs associated with primary energy sources.



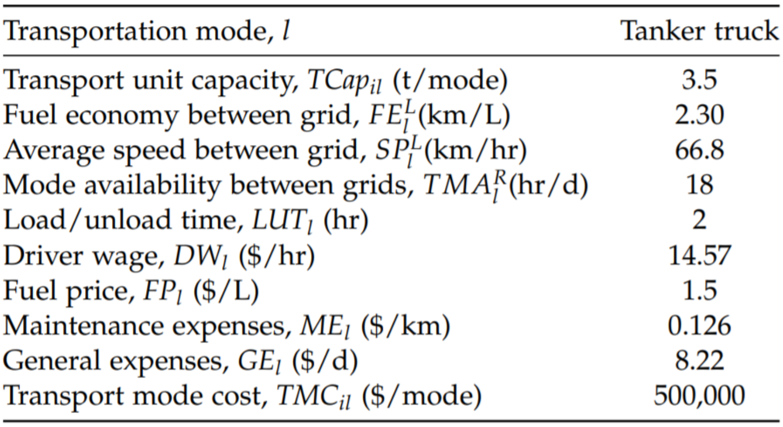
Primary energy source, e	Natural gas	PV-elect	Wind-elect	Hydro-elect	Nuclear elect
Unit cost of energy source, UEC_e ($/unit)	0.12	0.53	0.05	0.05	0.05
Unit import cost of energy source, UIC_e ($/unit)	0.012	0.005	0.005	0.005	0.005

Table A.2: production capacities and costs of hydrogen plants.



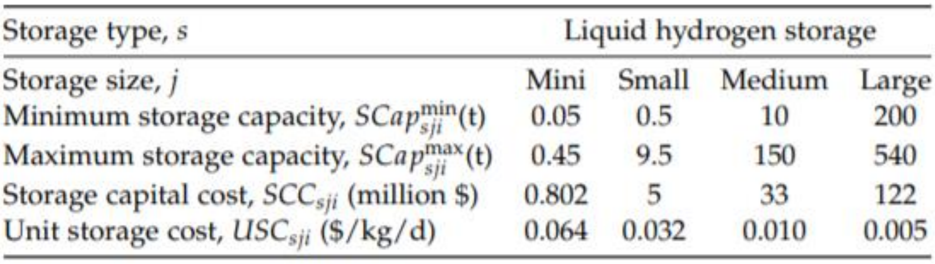
Plant type, p	Steam methane reforming			Centralized electrolysis			Distributed electrolysis	
Plant size, j	Small	Medium	Large	Small	Medium	Large	Small	Medium
$PCap_{pji}^{\min}$(t/d)	0.3	10	200	0.3	1.05	10	0.05	0.45
$PCap_{pji}^{\max}$(t/d)	9.5	150	960	2.5	9.5	150	0.4	1
γ_{epj}	4.02	3.34	3.16	52.49	52.49	52.49	52.49	52.49
PCC_{pji} ($ $\times 10^6$)	29	224	903	20.198	61	663	4.03	9.02
UPC_{pji} ($/kg)	3.36	1.74	1.43	4.94	4.69	4.59	6.24	5.38

Table A.3: costs and characteristics of transportation modes.



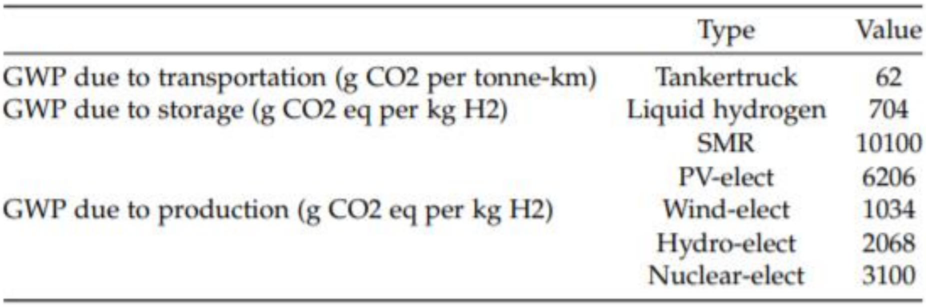
Transportation mode, l	Tanker truck
Transport unit capacity, $TCap_{il}$ (t/mode)	3.5
Fuel economy between grid, FE_l^L(km/L)	2.30
Average speed between grid, SP_l^L(km/hr)	66.8
Mode availability between grids, TMA_l^R(hr/d)	18
Load/unload time, LUT_l (hr)	2
Driver wage, DW_l ($/hr)	14.57
Fuel price, FP_l ($/L)	1.5
Maintenance expenses, ME_l ($/km)	0.126
General expenses, GE_l ($/d)	8.22
Transport mode cost, TMC_{il} ($/mode)	500,000

Table A.4: storage capacities and costs of liquid hydrogen storage facilities.



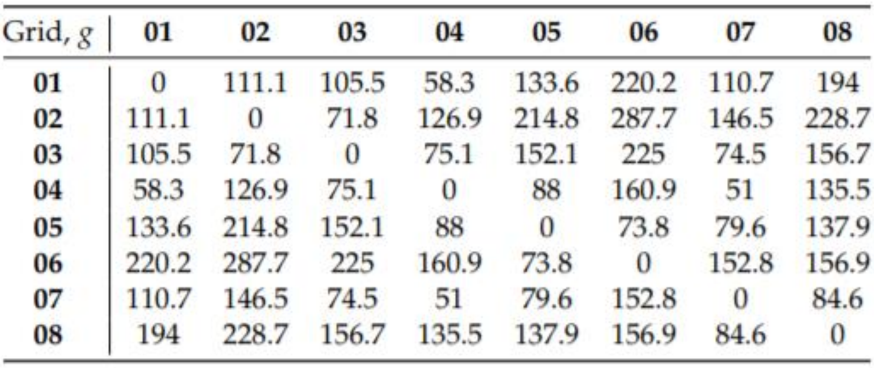
Storage type, s	Liquid hydrogen storage			
Storage size, j	Mini	Small	Medium	Large
Minimum storage capacity, $SCap_{sji}^{\min}$(t)	0.05	0.5	10	200
Maximum storage capacity, $SCap_{sji}^{\max}$(t)	0.45	9.5	150	540
Storage capital cost, SCC_{sji} (million \$)	0.802	5	33	122
Unit storage cost, USC_{sji} (\$/kg/d)	0.064	0.032	0.010	0.005

Table A.5: global warming potential.



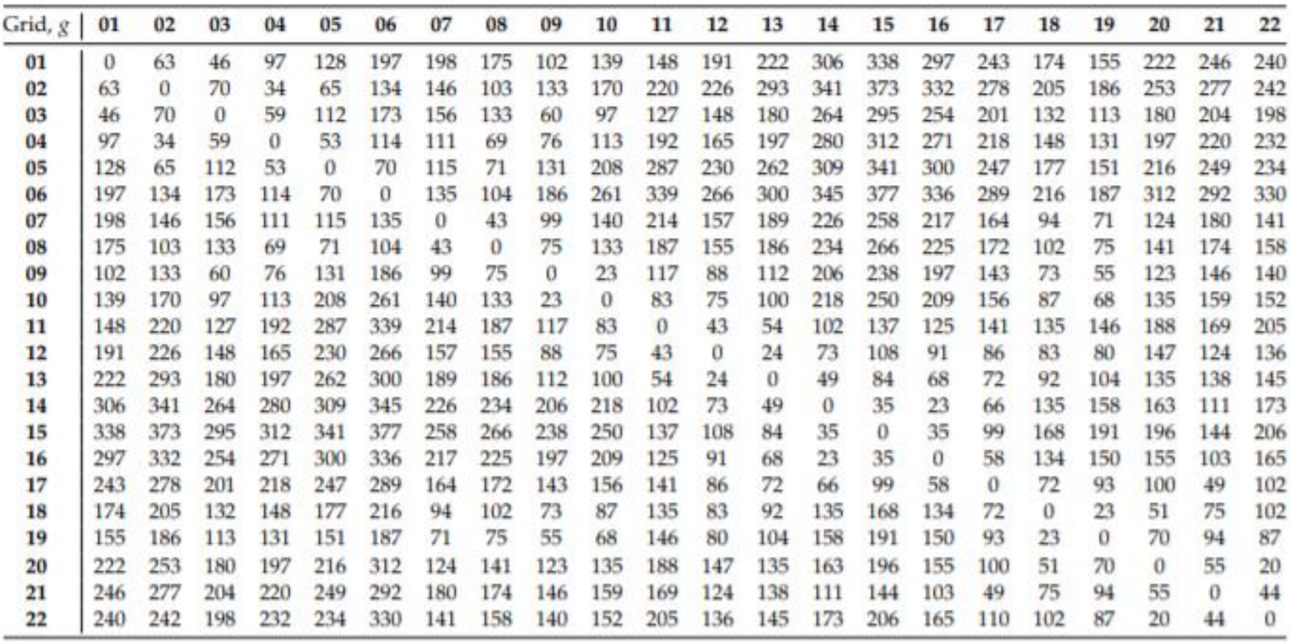
	Type	Value
GWP due to transportation (g CO2 per tonne-km)	Tankertruck	62
GWP due to storage (g CO2 eq per kg H2)	Liquid hydrogen	704
GWP due to production (g CO2 eq per kg H2)	SMR	10100
	PV-elect	6206
	Wind-elect	1034
	Hydro-elect	2068
	Nuclear-elect	3100

Table A.6: local and regional delivery distances for 8 grid instances.



Grid, g	**01**	**02**	**03**	**04**	**05**	**06**	**07**	**08**
01	0	111.1	105.5	58.3	133.6	220.2	110.7	194
02	111.1	0	71.8	126.9	214.8	287.7	146.5	228.7
03	105.5	71.8	0	75.1	152.1	225	74.5	156.7
04	58.3	126.9	75.1	0	88	160.9	51	135.5
05	133.6	214.8	152.1	88	0	73.8	79.6	137.9
06	220.2	287.7	225	160.9	73.8	0	152.8	156.9
07	110.7	146.5	74.5	51	79.6	152.8	0	84.6
08	194	228.7	156.7	135.5	137.9	156.9	84.6	0

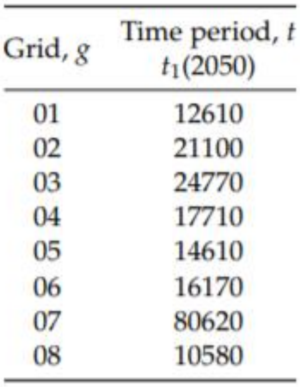
Table A.7: local and regional delivery distances for 22 grid instances.



Grid, g	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
01	0	63	46	97	128	197	198	175	102	139	148	191	222	306	338	297	243	174	155	222	246	240
02	63	0	70	34	65	134	146	103	133	170	220	226	293	341	373	332	278	205	186	253	277	242
03	46	70	0	59	112	173	156	133	60	97	127	148	180	264	295	254	201	132	113	180	204	198
04	97	34	59	0	53	114	111	69	76	113	192	165	197	280	312	271	218	148	131	197	220	232
05	128	65	112	53	0	70	115	71	131	208	287	230	262	309	341	300	247	177	151	216	249	234
06	197	134	173	114	70	0	135	104	186	261	339	266	300	345	377	336	289	216	187	312	292	330
07	198	146	156	111	115	135	0	43	99	140	214	157	189	226	258	217	164	94	71	124	180	141
08	175	103	133	69	71	104	43	0	75	133	187	155	186	234	266	225	172	102	75	141	174	158
09	102	133	60	76	131	186	99	75	0	23	117	88	112	206	238	197	143	73	55	123	146	140
10	139	170	97	113	208	261	140	133	23	0	83	75	100	218	250	209	156	87	68	135	159	152
11	148	220	127	192	287	339	214	187	117	83	0	43	54	102	137	125	141	135	146	188	169	205
12	191	226	148	165	230	266	157	155	88	75	43	0	24	73	108	91	86	83	80	147	124	136
13	222	293	180	197	262	300	189	186	112	100	54	24	0	49	84	68	72	92	104	135	138	145
14	306	341	264	280	309	345	226	234	206	218	102	73	49	0	35	23	66	135	158	163	111	173
15	338	373	295	312	341	377	258	266	238	250	137	108	84	35	0	35	99	168	191	196	144	206
16	297	332	254	271	300	336	217	225	197	209	125	91	68	23	35	0	58	134	150	155	103	165
17	243	278	201	218	247	289	164	172	143	156	141	86	72	66	99	58	0	72	93	100	49	102
18	174	205	132	148	177	216	94	102	73	87	135	83	92	135	168	134	72	0	23	51	75	102
19	155	186	113	131	151	187	71	75	55	68	146	80	104	158	191	150	93	23	0	70	94	87
20	222	253	180	197	216	312	124	141	123	135	188	147	135	163	196	155	100	51	70	0	55	20
21	246	277	204	220	249	292	180	174	146	159	169	124	138	111	144	103	49	75	94	55	0	44
22	240	242	198	232	234	330	141	158	140	152	205	136	145	173	206	165	110	102	87	20	44	0

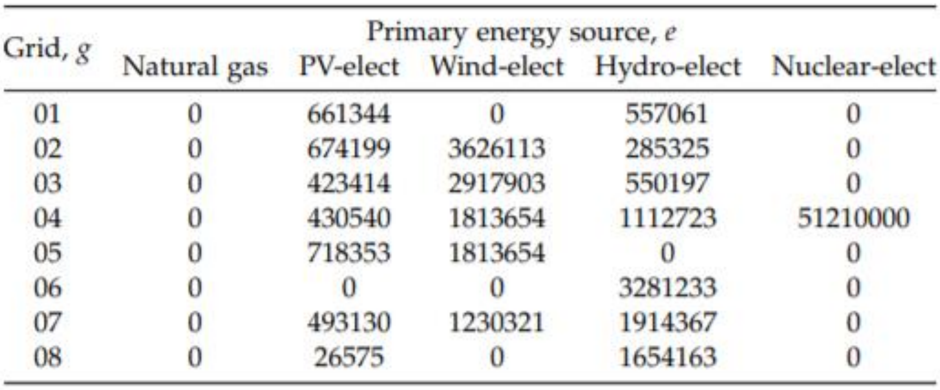
Data for instance HSC08g01p & HSC08g001p

Table A.9: Hydrogen demand of each grid ad time period (kg/d) for instance HSC08g01p.



Grid, *g*	Time period, *t* t_1(2050)
01	12610
02	21100
03	24770
04	17710
05	14610
06	16170
07	80620
08	10580

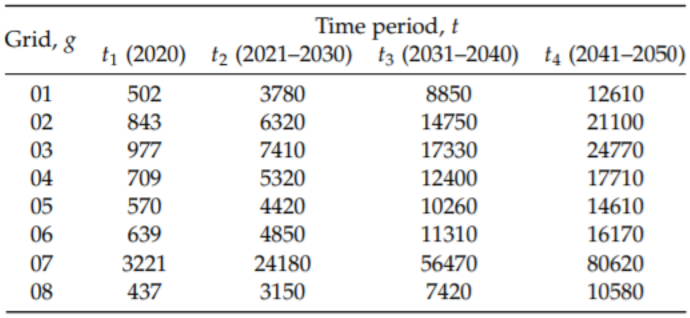
Table A.10: initial availability of energy sources (unit/d) for instance HSC08g01p.



Grid, *g*	Primary energy source, *e*				
	Natural gas	PV-elect	Wind-elect	Hydro-elect	Nuclear-elect
01	0	661344	0	557061	0
02	0	674199	3626113	285325	0
03	0	423414	2917903	550197	0
04	0	430540	1813654	1112723	51210000
05	0	718353	1813654	0	0
06	0	0	0	3281233	0
07	0	493130	1230321	1914367	0
08	0	26575	0	1654163	0

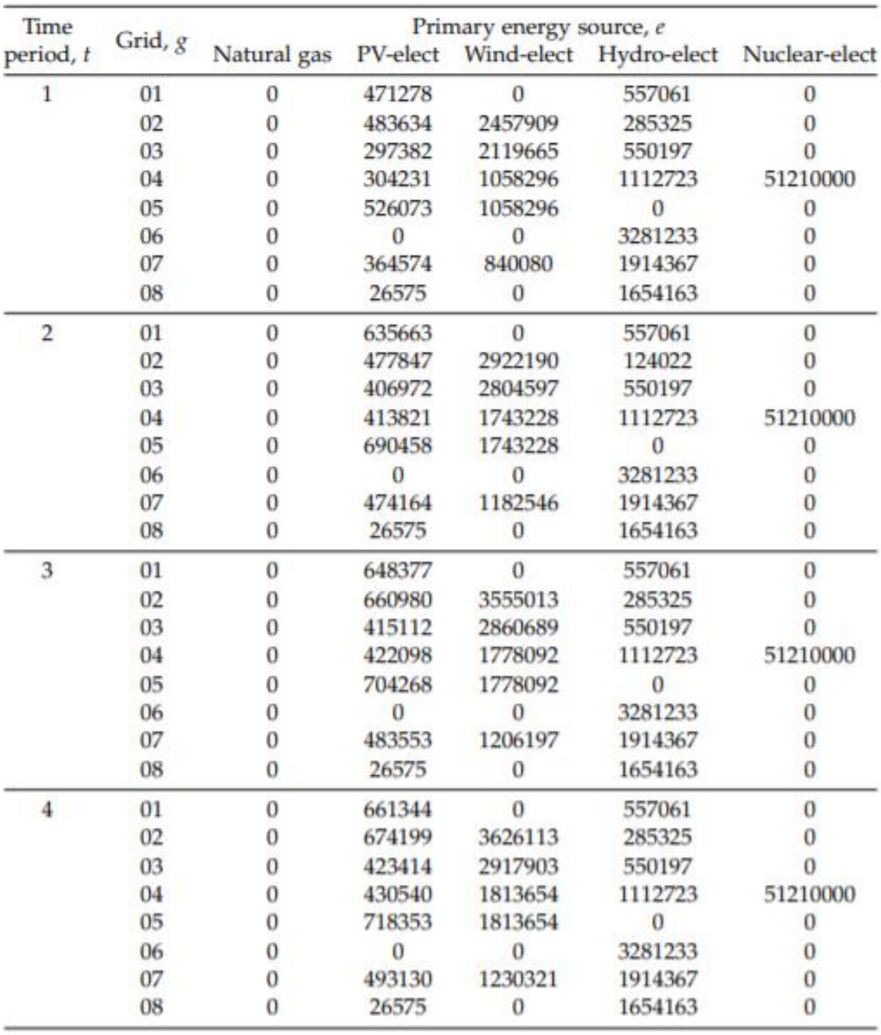
Data for instance HSC08g04p

Table A.11: Hydrogen demand of each grid ad time period (kg/d) for instance HSC08g04p.



Grid, g	Time period, t			
	t_1 (2020)	t_2 (2021–2030)	t_3 (2031–2040)	t_4 (2041–2050)
01	502	3780	8850	12610
02	843	6320	14750	21100
03	977	7410	17330	24770
04	709	5320	12400	17710
05	570	4420	10260	14610
06	639	4850	11310	16170
07	3221	24180	56470	80620
08	437	3150	7420	10580

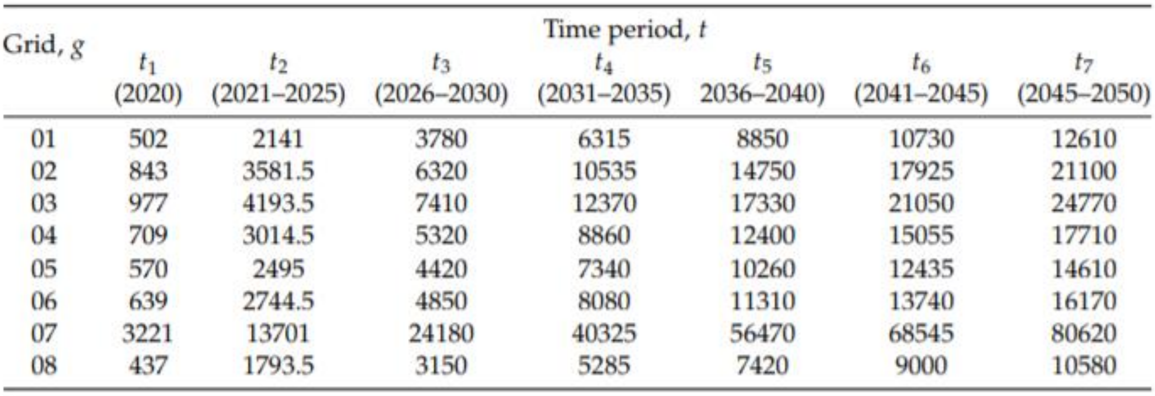
Table A.12: initial availability of energy sources (unit/d) for instance HSC08g04p.



Time period, t	Grid, g	Primary energy source, e				
		Natural gas	PV-elect	Wind-elect	Hydro-elect	Nuclear-elect
1	01	0	471278	0	557061	0
	02	0	483634	2457909	285325	0
	03	0	297382	2119665	550197	0
	04	0	304231	1058296	1112723	51210000
	05	0	526073	1058296	0	0
	06	0	0	0	3281233	0
	07	0	364574	840080	1914367	0
	08	0	26575	0	1654163	0
2	01	0	635663	0	557061	0
	02	0	477847	2922190	124022	0
	03	0	406972	2804597	550197	0
	04	0	413821	1743228	1112723	51210000
	05	0	690458	1743228	0	0
	06	0	0	0	3281233	0
	07	0	474164	1182546	1914367	0
	08	0	26575	0	1654163	0
3	01	0	648377	0	557061	0
	02	0	660980	3555013	285325	0
	03	0	415112	2860689	550197	0
	04	0	422098	1778092	1112723	51210000
	05	0	704268	1778092	0	0
	06	0	0	0	3281233	0
	07	0	483553	1206197	1914367	0
	08	0	26575	0	1654163	0
4	01	0	661344	0	557061	0
	02	0	674199	3626113	285325	0
	03	0	423414	2917903	550197	0
	04	0	430540	1813654	1112723	51210000
	05	0	718353	1813654	0	0
	06	0	0	0	3281233	0
	07	0	493130	1230321	1914367	0
	08	0	26575	0	1654163	0

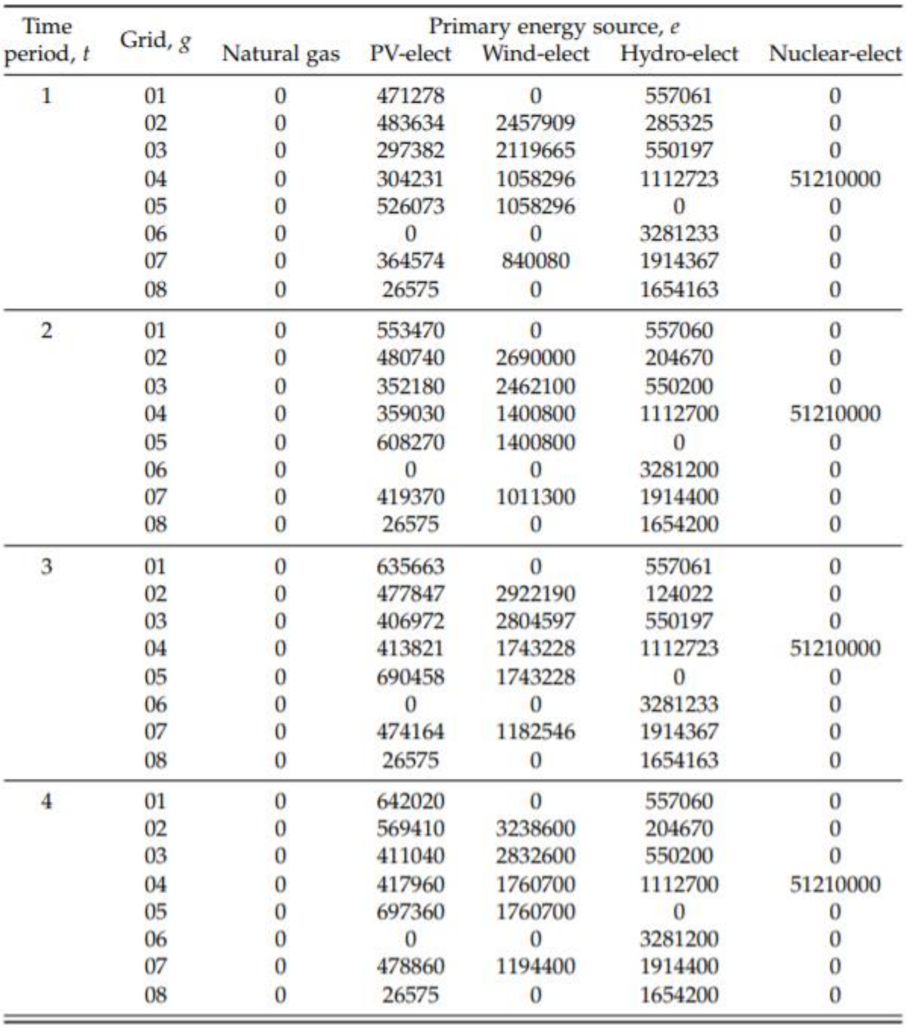
Data for instance HSC08g07p

Table A.13: Hydrogen demand of each grid ad time period (kg/d) for instance HSC08g07p.

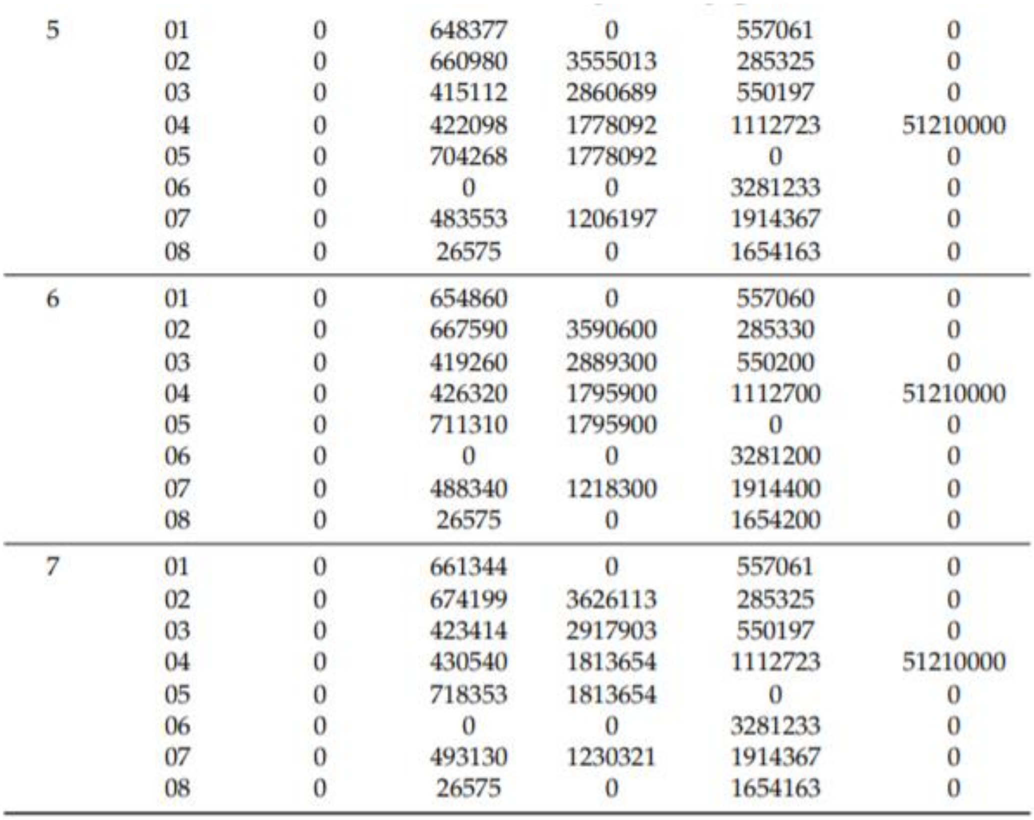


Grid, *g*	Time period, *t*						
	t_1 (2020)	t_2 (2021–2025)	t_3 (2026–2030)	t_4 (2031–2035)	t_5 2036–2040)	t_6 (2041–2045)	t_7 (2045–2050)
01	502	2141	3780	6315	8850	10730	12610
02	843	3581.5	6320	10535	14750	17925	21100
03	977	4193.5	7410	12370	17330	21050	24770
04	709	3014.5	5320	8860	12400	15055	17710
05	570	2495	4420	7340	10260	12435	14610
06	639	2744.5	4850	8080	11310	13740	16170
07	3221	13701	24180	40325	56470	68545	80620
08	437	1793.5	3150	5285	7420	9000	10580

Table A.14: initial availability of energy sources (unit/d) for instance HSC08g07p.



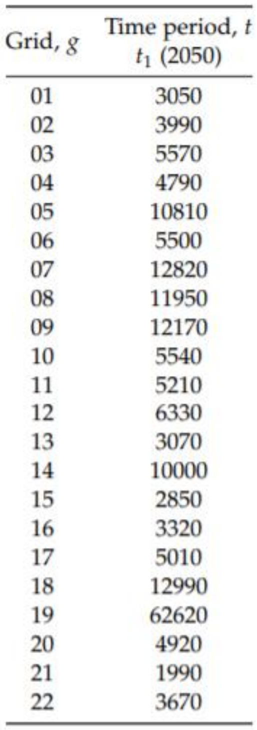
Time period, *t*	Grid, *g*	Primary energy source, *e*				
		Natural gas	PV-elect	Wind-elect	Hydro-elect	Nuclear-elect
1	01	0	471278	0	557061	0
	02	0	483634	2457909	285325	0
	03	0	297382	2119665	550197	0
	04	0	304231	1058296	1112723	51210000
	05	0	526073	1058296	0	0
	06	0	0	0	3281233	0
	07	0	364574	840080	1914367	0
	08	0	26575	0	1654163	0
2	01	0	553470	0	557060	0
	02	0	480740	2690000	204670	0
	03	0	352180	2462100	550200	0
	04	0	359030	1400800	1112700	51210000
	05	0	608270	1400800	0	0
	06	0	0	0	3281200	0
	07	0	419370	1011300	1914400	0
	08	0	26575	0	1654200	0
3	01	0	635663	0	557061	0
	02	0	477847	2922190	124022	0
	03	0	406972	2804597	550197	0
	04	0	413821	1743228	1112723	51210000
	05	0	690458	1743228	0	0
	06	0	0	0	3281233	0
	07	0	474164	1182546	1914367	0
	08	0	26575	0	1654163	0
4	01	0	642020	0	557060	0
	02	0	569410	3238600	204670	0
	03	0	411040	2832600	550200	0
	04	0	417960	1760700	1112700	51210000
	05	0	697360	1760700	0	0
	06	0	0	0	3281200	0
	07	0	478860	1194400	1914400	0
	08	0	26575	0	1654200	0



5	01	0	648377	0	557061	0
	02	0	660980	3555013	285325	0
	03	0	415112	2860689	550197	0
	04	0	422098	1778092	1112723	51210000
	05	0	704268	1778092	0	0
	06	0	0	0	3281233	0
	07	0	483553	1206197	1914367	0
	08	0	26575	0	1654163	0
6	01	0	654860	0	557060	0
	02	0	667590	3590600	285330	0
	03	0	419260	2889300	550200	0
	04	0	426320	1795900	1112700	51210000
	05	0	711310	1795900	0	0
	06	0	0	0	3281200	0
	07	0	488340	1218300	1914400	0
	08	0	26575	0	1654200	0
7	01	0	661344	0	557061	0
	02	0	674199	3626113	285325	0
	03	0	423414	2917903	550197	0
	04	0	430540	1813654	1112723	51210000
	05	0	718353	1813654	0	0
	06	0	0	0	3281233	0
	07	0	493130	1230321	1914367	0
	08	0	26575	0	1654163	0

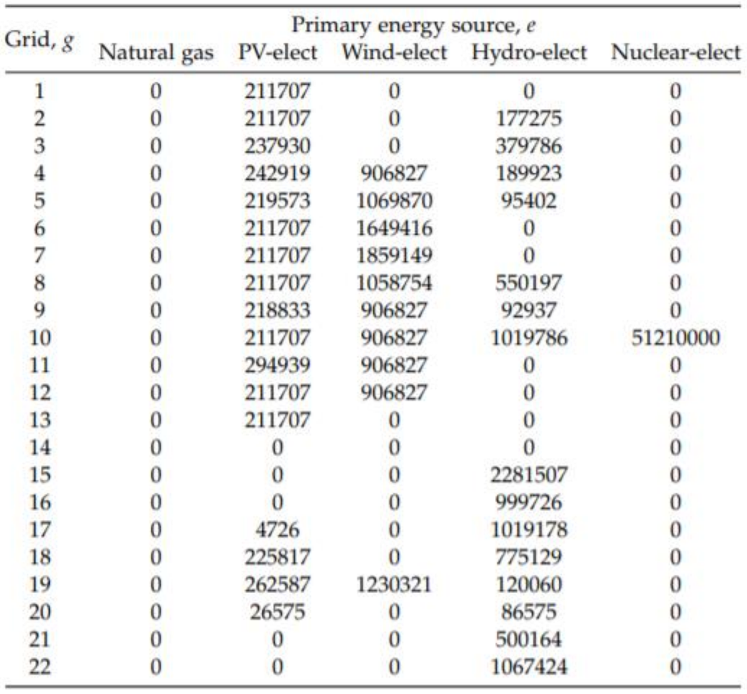
Data for instance HSC22g01p

Table A.15: Hydrogen demand of each grid ad time period (kg/d) for instance HSC22g01p.



Grid, g	Time period, t t_1 (2050)
01	3050
02	3990
03	5570
04	4790
05	10810
06	5500
07	12820
08	11950
09	12170
10	5540
11	5210
12	6330
13	3070
14	10000
15	2850
16	3320
17	5010
18	12990
19	62620
20	4920
21	1990
22	3670

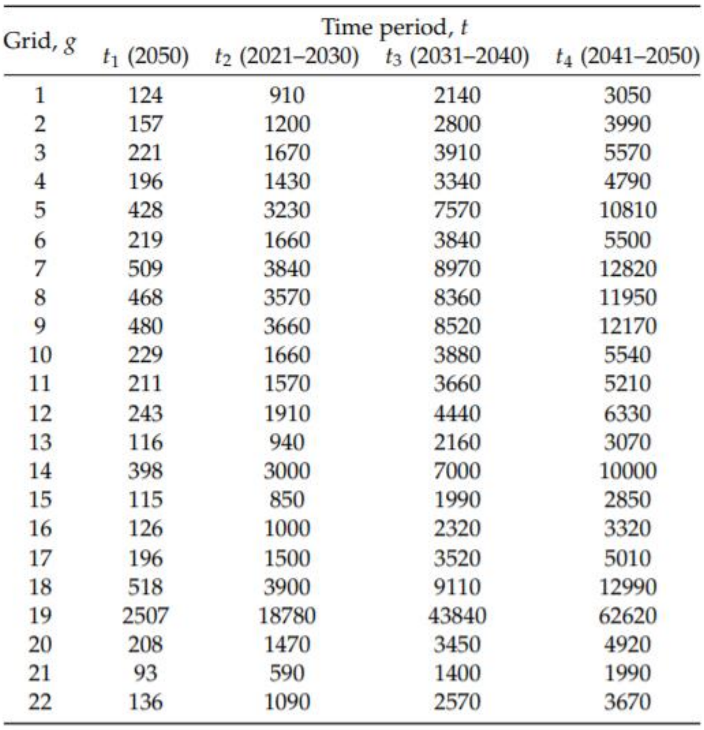
Table A.16: initial availability of energy sources (unit/d) for instance HSC22g01p.



Grid, g	Primary energy source, e				
	Natural gas	PV-elect	Wind-elect	Hydro-elect	Nuclear-elect
1	0	211707	0	0	0
2	0	211707	0	177275	0
3	0	237930	0	379786	0
4	0	242919	906827	189923	0
5	0	219573	1069870	95402	0
6	0	211707	1649416	0	0
7	0	211707	1859149	0	0
8	0	211707	1058754	550197	0
9	0	218833	906827	92937	0
10	0	211707	906827	1019786	51210000
11	0	294939	906827	0	0
12	0	211707	906827	0	0
13	0	211707	0	0	0
14	0	0	0	0	0
15	0	0	0	2281507	0
16	0	0	0	999726	0
17	0	4726	0	1019178	0
18	0	225817	0	775129	0
19	0	262587	1230321	120060	0
20	0	26575	0	86575	0
21	0	0	0	500164	0
22	0	0	0	1067424	0

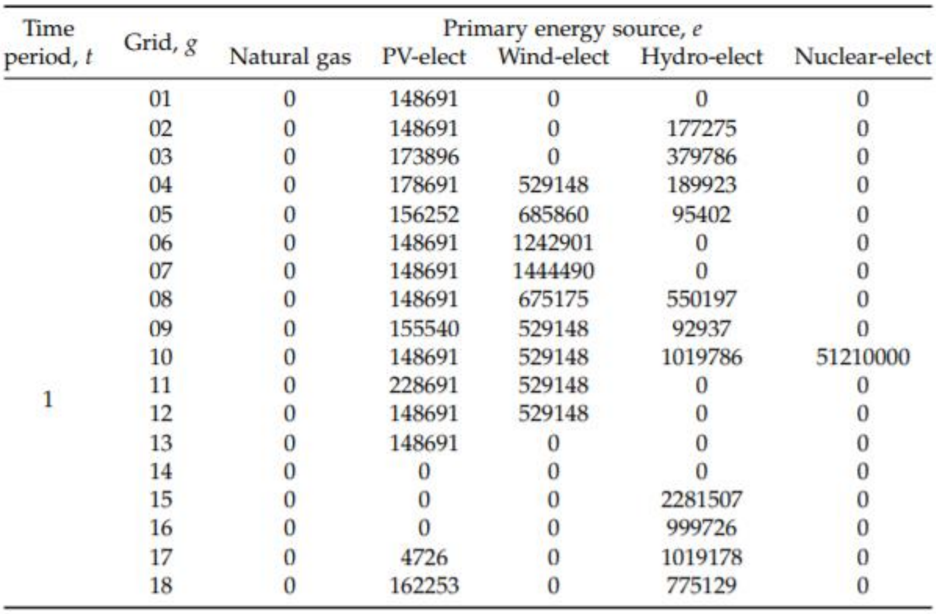
Data for instance HSC22g04p

Table A.17: Hydrogen demand of each grid ad time period (kg/d) for instance HSC22g04p.



Grid, g	Time period, t			
	t_1 (2050)	t_2 (2021–2030)	t_3 (2031–2040)	t_4 (2041–2050)
1	124	910	2140	3050
2	157	1200	2800	3990
3	221	1670	3910	5570
4	196	1430	3340	4790
5	428	3230	7570	10810
6	219	1660	3840	5500
7	509	3840	8970	12820
8	468	3570	8360	11950
9	480	3660	8520	12170
10	229	1660	3880	5540
11	211	1570	3660	5210
12	243	1910	4440	6330
13	116	940	2160	3070
14	398	3000	7000	10000
15	115	850	1990	2850
16	126	1000	2320	3320
17	196	1500	3520	5010
18	518	3900	9110	12990
19	2507	18780	43840	62620
20	208	1470	3450	4920
21	93	590	1400	1990
22	136	1090	2570	3670

Table A.18: initial availability of energy sources (unit/d) for instance HSC22g04p.



Time period, t	Grid, g	Primary energy source, e				
		Natural gas	PV-elect	Wind-elect	Hydro-elect	Nuclear-elect
1	01	0	148691	0	0	0
	02	0	148691	0	177275	0
	03	0	173896	0	379786	0
	04	0	178691	529148	189923	0
	05	0	156252	685860	95402	0
	06	0	148691	1242901	0	0
	07	0	148691	1444490	0	0
	08	0	148691	675175	550197	0
	09	0	155540	529148	92937	0
	10	0	148691	529148	1019786	51210000
	11	0	228691	529148	0	0
	12	0	148691	529148	0	0
	13	0	148691	0	0	0
	14	0	0	0	0	0
	15	0	0	0	2281507	0
	16	0	0	0	999726	0
	17	0	4726	0	1019178	0
	18	0	162253	0	775129	0

	19	0	197595	840080	120060	0
	20	0	26575	0	86575	0
	21	0	0	0	500164	0
	22	0	0	0	1067424	0
2	01	0	203486	0	0	0
	02	0	203486	0	177275	0
	03	0	228691	0	379786	0
	04	0	211047	1028326	95402	0
	05	0	63314	308497	28620	0
	06	0	203486	1585367	0	0
	07	0	203486	1786956	0	0
	08	0	203486	1017641	550197	0
	09	0	210335	871614	92937	0
	10	0	203486	871614	1019786	51210000
	11	0	283486	871614	0	0
	12	0	203486	871614	0	0
	13	0	203486	0	0	0
	14	0	0	0	0	0
	15	0	0	0	2281507	0
	16	0	0	0	999726	0
	17	0	4726	0	1019178	0
	18	0	217048	0	775129	0
	19	0	252390	1182546	120060	0
	20	0	26575	0	86575	0
	21	0	0	0	500164	0
	22	0	0	0	1067424	0
3	01	0	207556	0	0	0
	02	0	207556	0	177275	0
	03	0	233265	0	379786	0
	04	0	238156	889046	189923	0
	05	0	215268	1048893	95402	0
	06	0	207556	1617074	0	0
	07	0	207556	1822695	0	0
	08	0	207556	1037994	550197	0
	09	0	214542	889046	92937	0
	10	0	207556	889046	1019786	51210000
	11	0	289156	889046	0	0
	12	0	207556	889046	0	0
	13	0	207556	0	0	0
	14	0	0	0	0	0
	15	0	0	0	2281507	0
	16	0	0	0	999726	0
	17	0	4726	0	1019178	0
	18	0	221389	0	775129	0
	19	0	257438	1206197	120060	0
	20	0	26575	0	86575	0
	21	0	0	0	500164	0
	22	0	0	0	1067424	0
	01	0	211707	0	0	0
	02	0	211707	0	177275	0
	03	0	237930	0	379786	0
	04	0	242919	906827	189923	0
	05	0	219573	1069870	95402	0
	06	0	211707	1649416	0	0
	07	0	211707	1859149	0	0
	08	0	211707	1058754	550197	0
	09	0	218833	906827	92937	0

10	0	211707	906827	1019786	51210000
11	0	294939	906827	0	0
12	0	211707	906827	0	0
13	0	211707	0	0	0
14	0	0	0	0	0
15	0	0	0	2281507	0
16	0	0	0	999726	0
17	0	4726	0	1019178	0
18	0	225817	0	775129	0
19	0	262587	1230321	120060	0
20	0	26575	0	86575	0
21	0	0	0	500164	0
22	0	0	0	1067424	0

Bibliography

[1] V. H. Cantú, C. Azzaro-Pantel, A. Ponsich, A Novel Matheuristic based on Bi-Level Optimization for the Multi-Objective Design of Hydrogen Supply Chains, Computers & Chemical Engineering, 152 (2021), Available online at: https://doi.org/10.1016/j.compchemeng.2021.107370

[2] S. Arrhenius, On the Influence of Carbonic Acid in the Air upon the Temperature of the Ground, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science (Fifth Series) April 1896 49 (5), 237-276. Available online at: http://www.rsc.org/images/Arrhenius1896_tcm18-173546.pdf

[3] M. Maslin, Global Warming, a very short introduction. Oxford University Press, Oxford 2004.

[4] Paris Agreement to the United Nations Framework Convention on Climate Change, Dec. 12, 2015, T.I.A.S. No. 16-1104.

[5] IEA, Global energy and CO2 emissions in 2020, Report extract, IEA Publications (2020). Available online at: https://www.iea.org/reports/world-energy-outlook-2020

[6] McKinsey, et al., How hydrogen empowers the energy transition Tech. Rep., Hydrogen Council (2017).

[7] Science Learning Hub – Pokapū Akoranga Pūtaiao, Hydrogen – the number 1 element (2009). Available online at: https://www.sciencelearn.org.nz/resources/1729-hydrogen-the-number-1-element

[8] I. Stafell, The Energy and Fuel Data Sheet, Univeristy of Birmingam, UK, (2011). Available online at: https://www.claverton-energy.com/wordpress/wp-content/uploads/2012/08/the_energy_and_fuel_data_sheet1.pdf

[9] U.S Department of Energy, Hydrogen production and Distribution, Alternative Fuels Data Center, Energy efficincie & Renawable Energy (2021). Available online at: https://afdc.energy.gov/fuels/hydrogen_production.html

[10] S. D.-L Almaraz, C. Azzaro-Pantel, L. Montastruc, Multi-objective optimisation of a hydrogen supply chain, doctorat de l'université de Toulouse, Laboratorie de Génie Chimique (2014).

[11] Path to hydrogen competitiveness, a cost perspective, Tech. Rep., Hydrogen Council (2020).

[12] A. Almansoori, N. Shah, Design and operation of a future hydrogen supply chain: Snapshot model, Chemical Engineering Research and Design 84 (6) (2006) 423 – 438. Abailable online at: https://doi.org/10.1205/cherd.05193

[13] A. Almansoori, N. Shah, Design and operation of a future hydrogen supply chain: Multi-period model, International Journal of Hydrogen Energy 34 (19) (2009) 7883 – 7897. Available online at: https://doi.org/10.1016/j.ijhydene.2009.07.109

[14] A. Almansoori, A. Betancourt-Torcat, Design of optimization model for a hydrogen supply chain under emission constraints - A case study of Germany, Energy 111 (2016) 414 – 429.

[15] S. D.-L. Almaraz, C. Azzaro-Pantel, L. Montastruc, M. Boix, Deployment of a hydrogen supply chain by multi-objective/multi-period optimisation at regional and national scales, Chemical Engineering Research and Design 104 (2015) 11 – 31.

[16] J. Kim, I. Moon, Strategic design of hydrogen infrastructure considering cost and safety using multiobjective optimization, International Journal of Hydrogen Energy 33 (21) (2008) 5887 – 5896.

[17] D. Câmara, T. Pinto-Varela, A. P. Barbósa-Povoa, Multi-objective optimization approach to design and planning hydrogen supply chain under uncertainty: A Portugal study case, in: Computer Aided Chemical Engineering, Vol. 46, Elsevier, 2019, pp. 1309 – 1314.

[18] A. Almansoori, N. Shah, Design and operation of a stochastic hydrogen supply chain network under demand uncertainty, International Journal of Hydrogen Energy 37 (5) (2012) 3965–3977.

[19] S. D.-L. Almaraz, C. Azzaro-Pantel, L. Montastruc, S. Domenech, Hydrogen supply chain optimization for deployment scenarios in the Midi-Pyrénées region, France, International Journal of Hydrogen Energy 39 (23) (2014) 11831–11845.

[20] C. A. Coello Coello, Introducción a la Optimización Evolutiva Multiobjetivo, Departamento de Computación, CINVESTAV-IPN (2009).

[21] E. Uresti, Optimización Multiobjetivo; una introducción.

[22] T. M. Emmerich, A. H. Deutz, A tutorial on multiobjective optimization: fundamentals and evolutionary methods, Natural Computing: a natural journal, 17 (3) (2018) 585 – 609. Available online at: https://doi.org/10.1007/s11047-018-9685-y

[23] J. D. Knowles, D. Corne, Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy, Evolutionary Computation 8 (2) (2000). Available online at: 10.1162/106365600568167

[24] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712 – 731. Abailable online at: 10.1109/TEVC.2007.892759

[25] M. Emmerich, N. Beume, B. Naujoks, An EMO algorithm using the hypervolume measure as selection criterion, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2005, pp. 62 – 76.

[26] M. Pescador-Rojas, R. Hernández Gómez, E. Montero, N. Rojas-Morales, M..-C. Riff, C. A. Coello Coello, An Overview of Weighted and Unconstrained Scalarizing Functions, International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2017, 499-513. Available online at: https://doi.org/10.1007/978-3-319-54157-0_34

[27] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182 – 197. Available online at: 10.1109/4235.996017

[28] I. Das, J. E. Dennis, Normal-Boundary Intersection: A New method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems, SIAM J. Optim, 8 (3) (2006) 631 – 647. Available online at: https://doi.org/10.1137/S1052623496307510

[29] K. Deb, R. B. Agrawal, Simulated binary crossover for continuous search space, Complex Systems 9 (2) (1995) 115 – 148.