

FINAL MASTER'S THESIS

**Master's degree in Interdisciplinary & Innovative
Engineering [UPC]**

**Laurea Magistrale in Computer Engineering
[UNIGE]**

Data and Knowledge Management for the Automatic Synthesis of Waste Revalorization Pathways



Report and Annex

Author:	Lluc Aresté Saló
Supervisor:	Moisès Graells Sobré
Co-supervisor:	Silvia Siri
Call:	2025, June

Abstract

In the current social and environmental context, ensuring the sustainability of production systems is essential. The transition from a linear to a circular economic model is a key solution to reduce dependence on raw materials and minimize the generation of waste and thus its social and environmental impacts.

However, closing the material cycles is not trivial, and efficient ways of recycling must be identified that transform industrial waste into useful resources for other processes. Taking advantage of specialized computational tools (graph databases, mathematical programming and process simulation), this thesis proposes a methodology to identify and evaluate the valorization pathways between different materials. The methodology has been implemented entirely with open-source tools, following the principles of open science.

Additionally, in order to demonstrate the application of the methodology, an illustrative example on the revalorization of carbon dioxide in methanol synthesis has been studied.

Resum

En l'actual context social i ambiental, garantir la sostenibilitat dels sistemes de producció és essencial. La transició d'un model econòmic lineal cap a un de circular es presenta com una solució clau per reduir la dependència en matèries primeres i minimitzar la generació de residus i, així els seus impactes socials i ambientals.

Tot i això, tancar els cicles de matèria no és trivial, i cal identificar vies eficients de reciclatge que transformin residus industrials en recursos útils per a altres processos, aprofitant eines computacionals especialitzades (bases de dades basades en grafs, programació matemàtica i simulació de processos). Aquesta tesi proposa una metodologia per identificar i avaluar les vies de valorització entre diferents materials. La metodologia ha estat implementada íntegrament amb eines de codi obert, seguint els principis de la ciència oberta.

Addicionalment, i amb l'objectiu de demostrar l'aplicació de la metodologia s'ha estudiat un exemple il·lustratiu sobre la revalorització del diòxid de carboni en la síntesi de metanol.

Resumen

En el actual contexto social y medioambiental, garantizar la sostenibilidad de los sistemas de producción es esencial. La transición de un modelo económico lineal a otro circular se presenta como una solución clave para reducir la dependencia de materias primas y minimizar la generación de residuos y, por tanto, su impacto social y medioambiental.

Sin embargo, cerrar los ciclos de materiales no es trivial, y hay que identificar formas eficientes de reciclaje que transformen los residuos industriales en recursos útiles para otros procesos, aprovechando herramientas computacionales especializadas (bases de datos basadas en grafos, programación matemática y simulación de procesos). Esta tesis propone una metodología para identificar y evaluar las vías de valorización entre diferentes materiales. La metodología se ha implementado íntegramente con herramientas de código abierto, siguiendo los principios de la ciencia abierta.

Además, para demostrar la aplicación de la metodología, se ha estudiado un ejemplo ilustrativo sobre la revalorización del dióxido de carbono en la síntesis del metanol.

Riassunto

Nell'attuale contesto sociale e ambientale, garantire la sostenibilità dei sistemi produttivi è essenziale. La transizione da un modello economico lineare a uno circolare è una soluzione chiave per ridurre la dipendenza dalle materie prime e minimizzare la produzione di rifiuti e quindi il loro impatto sociale e ambientale.

Tuttavia, la chiusura dei cicli dei materiali non è banale e occorre individuare modalità efficienti di riciclaggio che trasformino i rifiuti industriali in risorse utili per altri processi. Sfruttando strumenti computazionali specializzati (database a grafo, programmazione matematica e simulazione di processo), questa tesi propone una metodologia per identificare e valutare i percorsi di valorizzazione tra diversi materiali. La metodologia è stata implementata interamente con strumenti open source, seguendo i principi della scienza aperta.

Inoltre, per dimostrare l'applicazione della metodologia, è stato studiato un esempio illustrativo sulla rivalorizzazione dell'anidride carbonica nella sintesi del metanolo.

Acknowledgments

First of all, I would like to express my gratitude to the professor Moisès Graells Sobré from the Center for Process and Environmental Engineering (CEPIMA) at the Universitat Politècnica de Catalunya (UPC), for his continuous guidance and support not just throughout this project but since I start my research at CEPIMA research group.

I would also like to sincerely thank Professor Silvia Siri from Università di Genova (UNIGE) for co-supervising my work, her academic insight and thoughtful feedback have enriched this work and contributed significantly to its interdisciplinary nature.

My thanks also extend to the rest of professors and members of the CEPIMA research group, with whom I have had the privilege of working over the past three years and with whom I will continue this research in this project during my upcoming PhD program.

Special thanks to Yeray Alcaraz Galván and Pau Trujillo Tremosa, undergraduate students in Chemical Engineering at UPC, whom I have had the honor of co-supervise. Their work on topics closely related to this thesis have been inspiring and highly valuable.

Last but not least, I want to thank my family and friends for their support, encouragement and patient not just during my Master's Thesis but throughout my entire academic journey.

Glossary

Academic and Research Institutions:

CEPIMA Center for Process and Environmental Engineering

ENCORE Energy, Catalysis, Process and Reaction Engineering

UNIGE Università di Genova

UPC Universitat Politècnica de Catalunya

Concepts and General Terms:

DIKW Data-Information-Knowledge-Wisdom hierarchy

TRL Technology Readiness Level

W3C World Wide Web Consortium

GUI Graphical User Interface

Sustainability and Circular Economy:

EIPs Eco-Industrial Parks

IS Industrial Symbiosis

SDGs Sustainable Development Goals

W2R Waste to Resource

Data and Knowledge Management:

DBMS Data Base Management System

GQL Graph Query Language

NoSQL Not Only SQL

PG Property Graph

RDF Resource Description Framework

SQL Structured Query Language

Engineering Frameworks and Models:

IRenE Information Retrieval and Extraction

P-graph Process Graph

STN State Task Network

Mathematical Optimization:

AML Algebraic Modeling Language

IP Integer Programming

LP Linear Programming

MILP Mixed Integer Linear Programming

MINLP Mixed Integer Non-Linear Programming

NLP Non-Linear Programming

QP Quadratic Programming

Process Systems Engineering:

PSE Process Systems Engineering

CAPE Computer Aided Process Engineering

CoLaN CAPE-OPEN Laboratories Network

DSS Decision Support System

MIMO Multiple Input Multiple Output

Index

Abstract	i
Resum	ii
Resumen	iii
Riassunto	iv
Acknowledgments	v
Glossary	vi
1 Preface	11
1.1 Background	11
1.2 Motivation	11
2 Introduction	12
2.1 Objectives	13
3 State of the Art	14
3.1 Production Data Collection	14
3.2 Information & Knowledge Management	17
3.3 Modeling & Optimizing Process Networks	21
3.4 Detailed Process Modelling	24
4 Methodology & Implementation	27
4.1 Graph Database	28
4.2 Reduced Graph	29
4.3 Superstructure Formalization	31
4.4 Superstructure Optimization	33
4.5 Chemical Process Simulation	34
5 Problem Definition	36
6 Proof Of Concept	40
6.1 Base Case	41
6.2 Alternative case	45
7 Conclusions	48
8 Economical Analysis	50
9 Sustainability Analysis	51
10 Ethical Implications	52
Annex A: Graphical User Interface	53
References	54
List of Figures	59
List of Tables	59

1 Preface

1.1 Background

This master's thesis is developed within the framework of the double degree master's program, combining the Master's degree in Interdisciplinary and Innovative Engineering at UPC and the Laurea Magistrale in Computer Engineering at UNIGE. The interdisciplinary nature of this program has provided a unique opportunity to integrate concepts from Computer and Process Systems Engineering.

The project follows one of the research lines of the CEPIMA subgroup, part of the Energy, Catalysis, Process and Reaction Engineering (ENCORE) research group in the Chemical Engineering Department of UPC, specifically in the field of Process Systems Engineering (PSE).

This thesis is intended to serve as the starting point for a future PhD thesis in Chemical Process Engineering in CEPIMA research group.

1.2 Motivation

The main motivation behind this project is sustainability. Sustainability in industrial activities is a topic of growing concern, especially as recent assessments show that six out of nine planetary boundaries were exceeded in 2023 (Richardson et al. 2023). One of the most significant challenges is the large amount of waste generated by traditional industrial systems, which are still largely based on linear economic models. Transitioning from a linear to a circular economy is therefore essential to reduce environmental pressures and promote more sustainable production and consumption patterns, as emphasized in Goal 12 of the Sustainable Development Goals (SDGs) (United Nations 2015).

Transitioning to a circular economy involves not just recycling waste, but efficiently transforming it into valuable raw materials through well-defined process pathways. While recycling plays a key role, it is often more energy-intensive than reduction or reuse, due to the complexity of the required chemical and physical transformations. Achieving effective and efficient Waste to Resource (W2R) conversion demands the ability to search, analyze, assess, and design optimal transformation processes. This must be done rapidly and accurately in a data-rich but noisy environment, where available data is often fragmented, uncertain, and poorly standardized. Therefore, specialized computing tools are essential tools that integrate information management, process optimization, and decision support to enable scalable, intelligent solutions for circular economy challenges.

2 Introduction

In process engineering, decision-making procedures are increasingly evolving to address not only economic performance but also sustainable concerns. Sustainability is defined as "the quality of being able to continue over a period of time" (Cambridge English Dictionary 2025) and is guided by the three pillars, economy, society and environment, first introduced by Brundtland and Khalid (1987).

In this context, the transition from a linear to a circular economy model is crucial. The circular economy approach advocates to reduce both disposed waste and raw materials consumption by promoting reuse and recycle as shown in Figure 1.

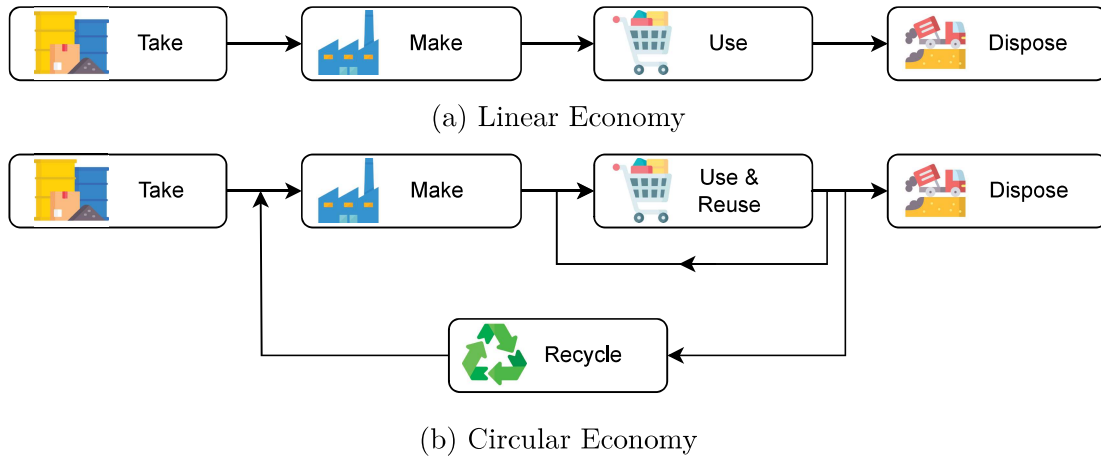


Figure 1: Linear and Circular Economy Models

The focus of this thesis is on the recycle stage, more specifically in the chemical recycling, also referred to as upcycling, as valuable products for industry are intended to be produced from waste. The determination of upcycling or W2R pathways is not always straightforward and will be the subject of this work.

Within the framework of determining W2R pathways, economic considerations focus on identifying the most cost-effective solutions; environmental priorities involve the treatment of harmful wastes to reduce ecological impact; and social aspects emphasize the broader benefits of waste treatment for communities and public health.

This project primarily seeks to answer the following research questions:

- What resources of industrial interest can be generated from specific waste streams, and what is the most efficient pathway to do so?
- Given a specific raw material, what types of wastes can be used to produce it, and which pathways enables the most efficient transformation?

The answer to those questions fosters this transition allowing the development of Eco-Industrial Parks (EIPs), which promote Industrial Symbiosis (IS) by enabling the reuse of waste as valuable inputs for other processes by transforming it into useful feedstock.

The complexity of this task lies in the large amount of processes, scientific knowledge, and heterogeneous data involved. To address these challenges, a data driven computational methodology will be developed. This methodology will integrate the acquisition of data, its storage and processing.

2.1 Objectives

The objectives of this thesis are as follows:

1. Identify the key steps to automate the pathway selection and evaluation procedure.
2. Develop modules for addressing each one of the steps.
3. Integrate the individual modules into a cohesive system.
4. Construct a unified and automated Decision Support System (DSS).
5. Demonstrate the performance of the DSS using a specific case study.

3 State of the Art

In this chapter the necessary background for the comprehension of the thesis will be presented. The organization of the following sub-sections is loosely inspired in Data-Information-Knowledge-Wisdom hierarchy (DIKW) (Ackoff 1989; Zeleny 1987) as they span from the management of basic chemical information to the modelling of process networks and the optimization procedures supporting decision-making. As it is represented in Figure 2, DIKW concepts build one upon the other and so in order to arrive to Wisdom we have to go across Data, Information and Knowledge.

Data is the unprocessed facts, that once they are processed and connected giving answers to the who, what, where and when turns into Information. This Information can be analyzed in order to extract conclusions answering to the how constructing in this way Knowledge, from which can be discerned the why, and use it to make judgments and take decisions, what is Wisdom.

Its important, however to state that the borders among those 4 concepts is diffuse and so they won't be clearly separated in the following subsections.

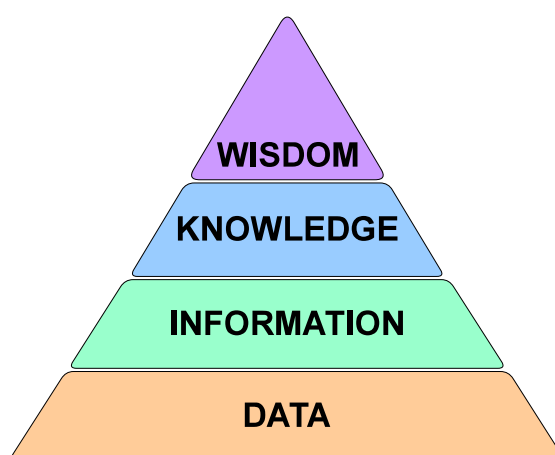


Figure 2: DIKW pyramid

Following this hierarchy will permit to discuss over the main concepts important for the constructions of a DSS system within a data driven methodology.

3.1 Production Data Collection

Data Collection is a critical step in data driven applications as it constructs the base from which all of the following phases up until the decision making are constructed.

The current era is characterized by the Big Data, which emerged with the appearance of Web 2.0 where users began to generate content through blogs, social media, wikis, etc. Although this new paradigm increases the availability of data, it doesn't necessarily

generate useful data, in fact, much of it may be irrelevant or noisy, complicating the task of extracting meaningful insights (Jing Han et al. 2011; Saxenna, Ali, and Singh 2014).

The complexities of Big Data are often described through the so-called 5V's of Big Data: Volume, Variety, Veracity, Velocity, and Value, as illustrated in Figure 3. These dimensions highlight the complexities and scope of managing large datasets (Hariri, Fredericks, and Bowers 2019).

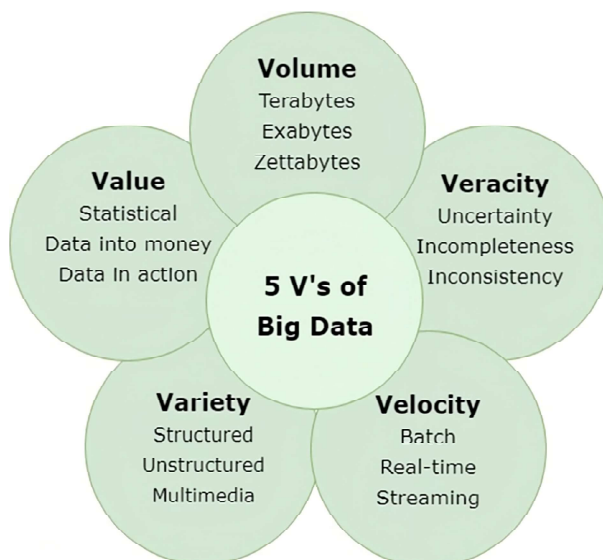


Figure 3: 5 V's of Big Data (Hariri, Fredericks, and Bowers 2019)

In that context the application of FAIR data principles of Open Science, first purposed by Wilkinson et al. (2016) and endorsed by European Commission (2020) facilitates the navigation over this large amount of data. According to them, data should be:

- **Findable**: Easy to find for both humans and computers.
- **Accessible**: Retrievable through well-defined, standardized protocols.
- **Interoperable**: Compatible with other datasets and systems.
- **Reusable**: Well-documented so that they can be reused in different setting.

In computer science this principles have translated into the Semantic Web initiative, often referred to as Web 3.0 (Rasmussen et al. 2020). The Semantic Web aims to provide a structured framework for data on the internet, enabling machines to understand and process information by using standardized vocabularies and ontologies. This vision is supported by the World Wide Web Consortium (W3C) and leads to the development of interconnected knowledge systems, such as the Linked Open Data Cloud (see Figure 4b).

One of the key frameworks supporting this vision is the 5-Star Open Data Scheme (Figure 4a), proposed by Berners-Lee (2010), which encourages to incrementally enhance the openness and usability of the data:

- ★ Available on the Web (whatever format) under an open license.
- ★★ Available as structured data (e.g., Excel).
- ★★★ Available in a non-proprietary open format (e.g., CSV instead of Excel).
- ★★★★ Use URIs to denote things, so that people can point to your data.
- ★★★★★ Link your data to other data to provide context.

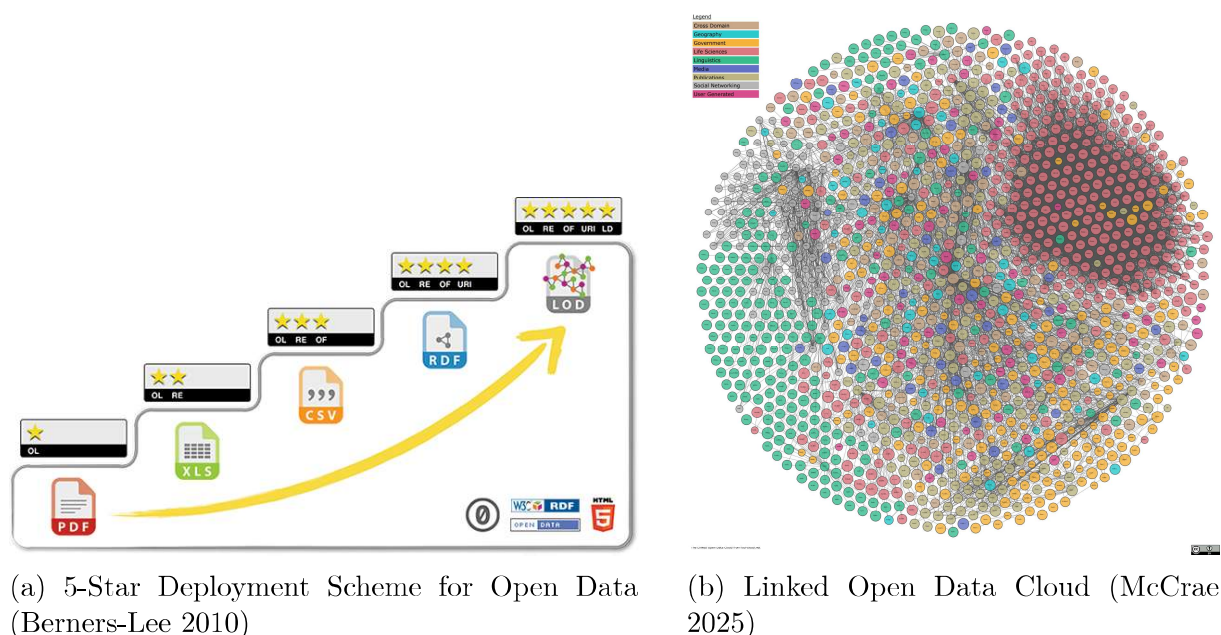


Figure 4: Overview 5-Star Deployment Scheme and of the Linked Open Data

However, despite the numerous ongoing efforts, many domains still face significant challenges in accessing standardized and open datasets that fully adhere to the FAIR principles and the 5-Star Deployment Scheme. A notable example is the field of Process Engineering, where data is often non-standardized, incomplete, uncertain, and subject to economic restrictions. Additionally, domain-specific constraints frequently limit the applicability of collected data outside its original context, reducing its potential for reuse and broader integration.

In that context, many studies have been performed to retrieve information from Scientific Literature. Lechtenberg et al. (2022) developed Information Retrieval and Extraction (IRenE) procedure which permits to access the scientific abstract and citation database Scopus[®] to retrieve new documents based on a set of initial relevant papers ("seed corpus") permitting to automate the literature review and consequently reaching a higher number of useful studies and research. Kato and Kano (2023) reported another method which permitted to retrieve mathematical equations from documents in order to build physical models.

3.2 Information & Knowledge Management

It is crucial to organize data to efficiently facilitate querying and data usage. Effective data organization not only supports better decision-making but also enhances the accessibility and reliability of the information.

Organized data is typically stored in databases, which are managed by Data Base Management System (DBMS). One of the most widely used models is the relational (or traditional) database first proposed in the seminal work by Codd (1970), being still the most popular with a 90% of the use cases (Khazaei et al. 2016). Relational databases store information in relations which essentially consist of tables with rows (tuples) and columns (attributes) as shown in Figure 5a.

A key feature in relational databases is the use of Structured Query Language (SQL), a standardized (*ISO/IEC 9075* (2023)) and declarative language in which users specify what data they want to retrieve, rather than how to retrieve it, first purposed by Chamberlin and Boyce (1974). This abstraction is possible thanks to the separation between the logical level, which defines the structure and relationships of the data, and the physical level, which determines how data is actually stored and accessed on disk.

Relational databases operations are constructed around the concept of transaction, which is a fundamental unit of work composed of a sequence of database actions, reads and writes. Transactions are designed to follow the ACID properties (Vossen 2009):

- **Atomicity:** All actions are completed successfully or none are applied at all.
- **Consistency:** The database remains in a valid state before and after a transaction.
- **Isolation:** Concurrently executed transactions do not interfere with each other.
- **Durability:** Committed transactions persist even after a system failure.

However, the Big Data era, as discussed in previous subsection, brought with it a massive increase in the volume, variety and velocity of data. These characteristics challenged the performance and scalability of traditional relational databases and the strict adherence to ACID properties (Banothu, Bhukya, and Sharma 2016).

In response to these limitations, Not Only SQL (NoSQL) systems emerged, introducing alternative data management solutions that are better suited to the demands of large-scale, fast-evolving, and heterogeneous datasets. These systems prioritize scalability and availability, relaxing ACID properties into BASE properties.

- **Basically Available:** System always available.
- **Soft State:** System's data may be temporarily not synchronized.
- **Eventual Consistency:** Database will eventually converge to a consistent state.

NoSQL systems are broadly categorized based in their data model in Aggregate-Oriented stores: Key-Value, Column-Family and Document-Oriented, as well as Graph-Based databases: Property Graph (PG) and Resource Description Framework (RDF) stores. In Table 1 the taxonomy along with representative examples of the different types of NoSQL systems is collected.

Table 1: Taxonomy of NoSQL Databases

Category	Type	Example
Aggregate-Oriented	Key-Value Store	Riak
	Column-Family Store	Cassandra
	Document Store	MongoDB
Graph-Based	Property Graph (PG)	Neo4j
	RDF / Triple Store	GraphDB

Each category of NoSQL databases addresses specific application requirements, allowing users to choose the most suitable system based on factors such as data complexity, schema flexibility, and scalability needs. Key-value stores are optimized for rapid data retrieval using keys and are ideal for simple, high-performance applications that do not require complex querying. Column-family stores support flexible schemas, allowing rows to have varying columns without requiring placeholder (e.g., NULL) values. They can also embed complex data types, making them well-suited for semi-structured data with high write throughput. Document stores support nested, hierarchical JSON-like structures and allow each document to have its own schema, providing high flexibility in how data is modeled. Graph databases, including both PG and RDF/Triple stores, are designed to efficiently model and query highly interconnected data. Following, in Figure 5b a representation of the different data models can be visualized.

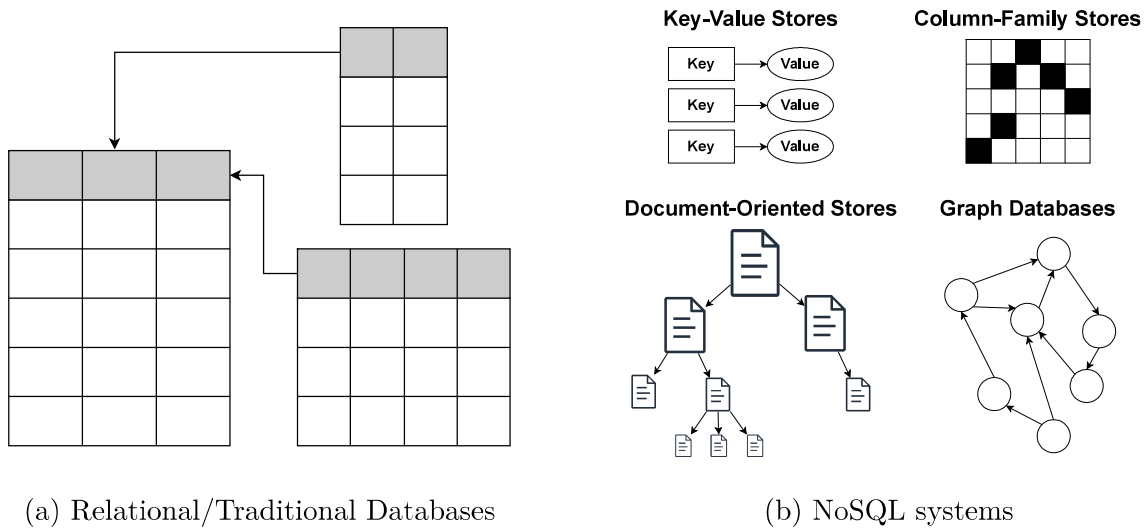


Figure 5: Databases data model schema

As previously discussed, Key-Value stores are best suited for representing simple data structures, while graph-based databases are ideal for handling highly complex and interconnected data. Column-Family and Document-Oriented stores occupy an intermediate position in terms of both data complexity and flexibility. This positioning is illustrated

in Figure 6, which maps various data store technologies according to their capacity to manage increasing data size and complexity. Notably, all NoSQL solutions demonstrate superior scalability compared to traditional relational databases, making them well-suited for Big Data applications (Khazaei et al. 2016).

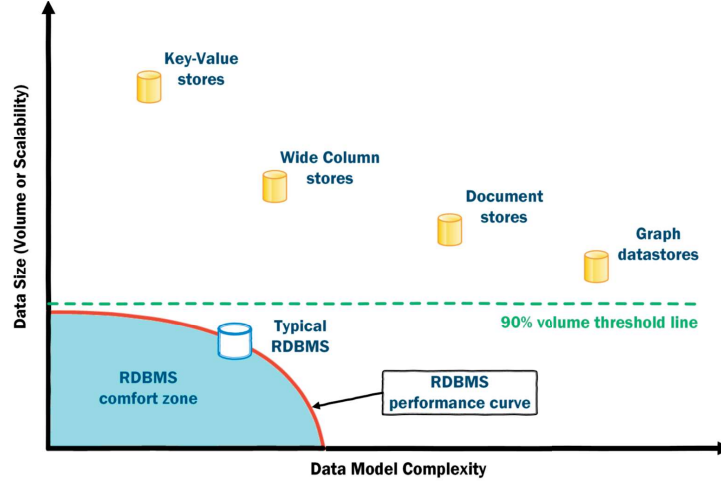


Figure 6: Positioning of Data Stores by Size and Complexity (Khazaei et al. 2016)

Graph databases suitability for the representation of complex data is mainly because they are constructed following Graph Theory, being physically represented in memory as a Graph $G(V, E)$ constructed by a set of V vertex (nodes) and connected by a set of E edges (relationships), what permits to cope with highly interconnected data structures (Angles and Gutierrez 2008).

As previously shown in Table 1, graph-based databases can be further categorized into 2 distinct types: Property Graph (PG) databases such as Neo4J, and also Resource Description Framework (RDF) / Triple Store databases such as GraphDB. However, many more graph-based databases exist in both categories, as well as intermediate or hybrid models, as shown in Table 2.

Property Graph databases emerged with other types of NoSQL databases during the rise of Web 2.0. In contrast, RDF Triple Stores were introduced by the W3C as part of the Semantic Web initiative, often referred to as Web 3.0 (Rasmussen et al. 2020).

Graph databases, as well as traditional databases have developed their own query languages being Cypher and Gremlin the most widely adopted by PG, and SPARQL in RDF stores. It is important to note the recent standardization performed on PG Query Languages under *ISO/IEC 39075* (2024), which introduced the Graph Query Language (GQL), the first ISO standard for a query language not based on SQL. This initiative, largely driven by the Cypher language originally developed by Neo4j, aims to unify the ecosystem, and be gradually adopted by all the systems.

Table 2: Overview of Graph Database Systems

Name	Logo	Model	License *	Query Language	Key Features
Neo4j		PG	Hybrid	Cypher	Mature ecosystem
MemGraph		PG	Hybrid	Cypher	Real-time analytics
FalkorDB		PG	Open	Cypher	Linear algebra backend
JanusGraph		PG	Open	Gremlin	Distributed & scalable
HugeGraph		PG	Open	Gremlin	Large Scale Graphs
NebulaGraph		PG	Open	nGQL	Cloud native
TigerGraph		PG	Commercial	GSQL	Real-time analytics
GraphDB		RDF	Open	SPARQL	Linked data integration
Cayley	 Cayley	RDF	Open	Gizmo, MQL & GraphQL	Layer over any DB
DGraph	 Dgraph by Hypermode	RDF	Open	DQL	GraphQL API
AllegroGraph	 AllegroGraph Franz Inc.	RDF	Commercial	SPARQL	AI integration
Stardog		PG-RDF	Closed	SPARQL	Dual graph model
Amazon Neptune		PG-RDF	Closed	Cypher, Gremlin & SPARQL	Dual graph model
PuppyGraph		Virtual Graph	Hybrid	Cypher & Gremlin	No data transfer
OrientDB		Graph-Doc	Open	O-SQL	Document-rich nodes
ArangoDB		Graph-Doc-KV	Hybrid	AQL	All-in-one

Name	Logo	Model	License	Query Language	Key Features
Hyper-GraphDB		Hyper-graph	Open	Java API	Granular modelling

* Hybrid license corresponds to intermediate licenses, offering open-source Community Edition with the essential functionalities, as well as a Commercial Edition with the full set of features.

3.3 Modeling & Optimizing Process Networks

The use of graph databases efficiently enables connecting data and producing information. However, knowledge refers to the abstraction beyond the tool and requires models, conceptual and mathematical, linking this information with a simplified and useful representation of the reality of the problems to be addressed. Thus, this section addresses both the conceptual models supporting the graph design and the mathematical models (mathematical programming) representing the material flows of processing networks. The detailed modeling and simulation of the processes involved is assessed in the following sub-section 3.4.

In that context the publication performed by Umeda, Hirai, and Ichikawa (1972) was the first to purpose the concept of superstructure as a form of data representation in process engineering. The superstructure concept is not just useful for containing the data, but it also facilitates the construction of mathematical programming formulations that permit to design the system and optimize the system according to the defined objective(s), such as cost minimization, benefit maximization or energy efficiency.

A superstructure is a network design to represent multiple possible systems by integrating them into a single scheme. That scheme can be later translated into a mathematical programming problem whose resolution will provide the optimal configuration according to the defined objective. However, in traditional superstructures where processes are represented as nodes and materials as edges, ambiguities can arise especially when dealing with Multiple Input Multiple Output (MIMO) structures.

To address these challenge, alternative graph representations that distinguish two distinct types of nodes for materials and processes have been proposed. Examples of that include the State Task Network (STN) representation, first introduced by Kondili, Pantelides, and R.W.H. Sargent (1993) for the scheduling of batch operation and later extended to process flowsheeting by Yeomans and Grossmann (1999), as well as the Process Graph (P-graph) framework introduced by Friedler et al. (1992). Both STN and P-graph use a bipartite graph structure with separate node types for states/materials and tasks/operating units, eliminating all the previous commented ambiguities.

In Figure 7, an example provided by Kondili, Pantelides, and R.W.H. Sargent (1993), of the utility of representing a superstructure using two distinct types of nodes is illustrated.

In this representation, rounded nodes correspond to materials, while rectangular nodes represent processes. From the initial Process Network shown in 7a, four different network configurations can be derived, including the alternatives depicted in 7b and 7c.

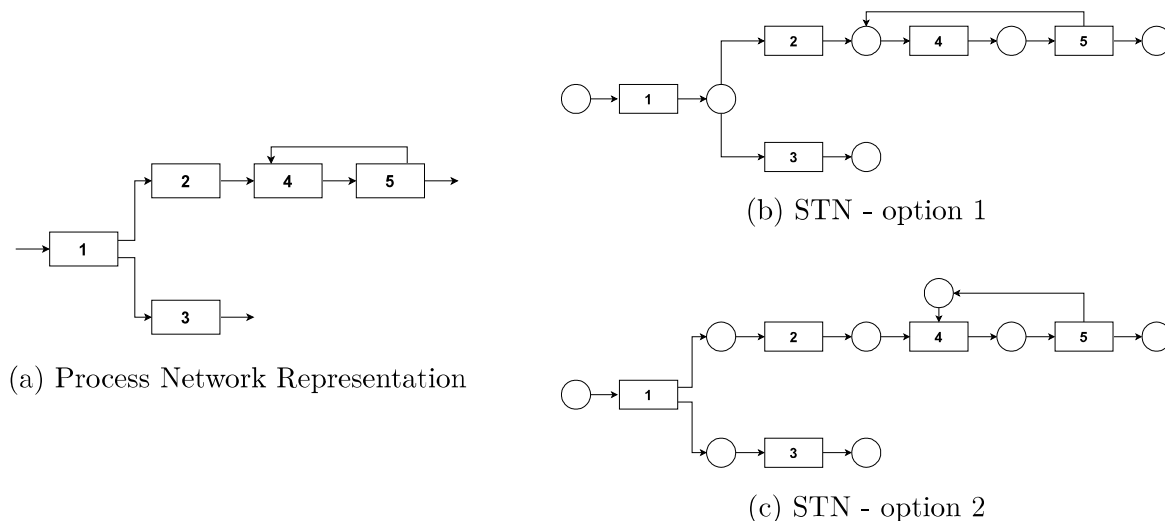


Figure 7: Comparison of process network and corresponding STN representations (Kondili, Pantelides, and R.W.H. Sargent 1993)

A graph database with a coherent data model is also known as a Knowledge Graph, what is an information structure from which knowledge can be extracted. There are several ways to extract that knowledge from the graph, one of them is by the use of queries permit to reduce the graph, obtaining the subgraph of interest and so answering to some questions, constructing in that way knowledge.

However, most of the questions that need to be answered require a more advanced judgment to retrieve among the available possibilities the more convenient according to quantitative criteria. In order to do so mathematical programming/optimization problems should be resolved.

Mathematical programming/optimization originates from the studies performed during World War II, in order to address the logistic challenges associated to it, leading to the development of the field now known as Operations Research (INFORMS 2025). Nowadays, optimization can be found in almost all the areas of science and engineering, in particular in chemical and process engineering the more typical problems are in process selection, design, operation and control (Biegler 2010).

Mathematical programming is a branch of applied mathematics used to formulate optimization problems. This problems involves the maximization or minimization of an objective function, quantification of the decision criteria, subject to some constraints and defined by variables (unknown) and parameters (known).

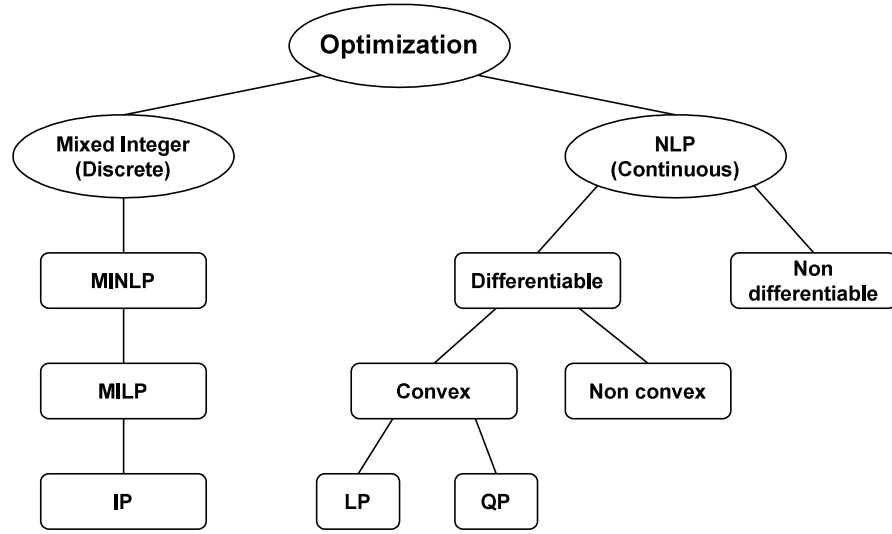


Figure 8: Optimization topology (Biegler 2010)

As shown in Figure 8, optimization procedures can be classified into continuous and discrete problems. The most general form of continuous optimization is known as Non-Linear Programming (NLP), where there exist non-linearities in either the objective function or the constraints. Within differentiable and convex NLP two important types of problems exist: Linear Programming (LP) and Quadratic Programming (QP). LP are the problems in which both the objective function and all constraints are linear. QP are the problems in which the constraints are linear but the objective is quadratic.

In discrete optimization, integer variables, often binary, are introduced to model decision-making, such as activating or deactivating system components. The most general form in this category is Mixed Integer Non-Linear Programming (MINLP), which combines nonlinear functions with integer variables. A special case of MINLP is Mixed Integer Linear Programming (MILP), where all nonlinear expressions are replaced with linear ones. Finally, when all variables are required to be integers and the problem remains linear, it becomes an Integer Programming (IP) problem.

In Figure 9, a general mathematical programming formulation is shown, consisting on an scalar objective function $f(\mathbf{x})$ to be maximized or minimized subject to a vector of decision variables $\mathbf{x}$ within the set X that represents the set of feasible solutions and is determined by the constraints of the model.

$$\begin{array}{ll}
 \text{(MP)} & \max_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \\
 & \text{subject to } \mathbf{x} \in X \subseteq \mathbb{R}^n
 \end{array}$$

Figure 9: General mathematical programming formulation

In order to solve those problems numerous algorithms exist. The selection of the most

appropriate algorithm depends on the morphology of the problem, and most of the times, more than one algorithm may be appropriate for an specific problem. This algorithms are implemented within solvers, which can be either commercial or open-sourced. To interface with solvers, Algebraic Modeling Language (AML) (see Table 3) are often employed as they permit to formulate the problem as a human-readable format that closely resembles the mathematical notation itself (Jusevičius and Paulavičius 2021).

Table 3: Overview of AML Systems

Name	Logo	Language/ Platform	License
GAMS	 G A M S	Proprietary Language	Commercial
AIMMS	 AIMMS	Proprietary Language	Commercial
AMPL	 AMPL	Proprietary Language	Commercial
Pyomo	 PYOMO	Python	Open (solvers may not be)
JuMP	 JUMP	Julia	Open (solvers may not be)

3.4 Detailed Process Modelling

The previous subsections have outlined the progression from Data to Information and finally Knowledge. To advance to the stage of wisdom, enabling decision-making within the context of a DSS, the involvement of expert supervision is required.

In both academic and industrial settings, expert decision-making is often facilitated by structured representations of chemical processes, what is flowsheets. These representations, typically developed using Chemical Process Simulation Software, model physical and chemical transformations through mathematical formulations that include mass and energy balances, phase equilibrium, and transport and reaction kinetics equations (Gil Chaves et al. 2016).

An example of such a process representation is shown in Figure 10, which depicts a flow-sheet developed using DWSIM chemical process simulator. This visual model illustrates how complex transformations and operational logic are encapsulated within a digital environment, facilitating both understanding and decision-making.

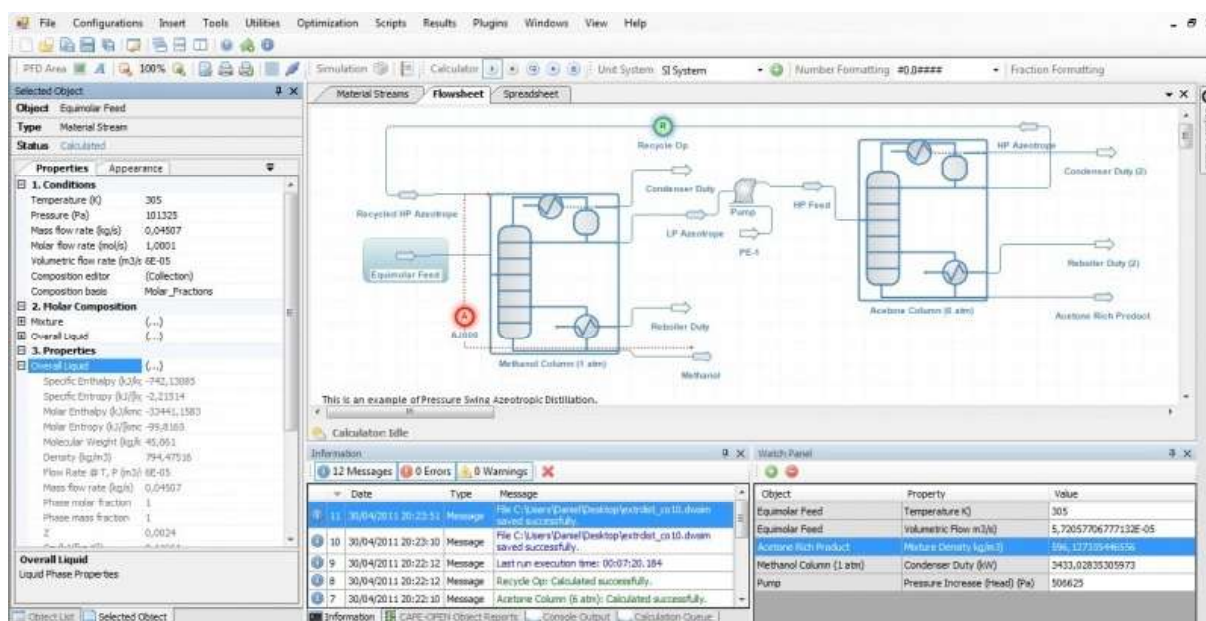


Figure 10: Process simulation in DWSIM (Wagner Oliveira de Medeiros 2011)

Chemical Process Simulation Software permits not only to predict the behavior of a process but to analyze different cases and apply optimization algorithms to improve its performance under economic, environmental and social aspects (Peters, Timmerhaus, and West 2003).

The development of Chemical Process Simulation Software initiated in the 60s-70s with the foundational studies performed by RWH Sargent and Westerberg (1964), A. Johnson (1972), and Seader, Seider, and Pauls (1974) driven by advances in computing (and the use of FORTRAN programming language). Over the following years, numerous additional studies were conducted, leading to the development of a wide range of commercial software packages, some of the most popular simulators are listed in Table 4.

However, with the independent development of the different simulators, interoperability became a significant challenge. In this context, an European Union funded project was initiated in 1997 involving participants from process industries, Chemical Process Simulation Software vendors and academic institutions that culminated with the Computer Aided Process Engineering (CAPE)-OPEN standard (Jarke et al. 1999). The CAPE-OPEN standard is regulated by the CAPE-OPEN Laboratories Network (CoLaN) non-profit organization (CoLaN 2025) and defines the rules and interfaces that allow CAPE applications to interoperate, standardizing and facilitating the implementation of software tools in process systems engineering (Belaud, Braunschweig, and White 2002).

Following the adoption of the CAPE-OPEN standard, which improved interoperability among process simulators, a new generation of tools emerged that embraced openness and accessibility. Notably, two significant developments were COCO and DWSIM simulators which were awarded by CoLaN (*CAPE-OPEN Award 2006*; *CAPE-OPEN Award 2012*). The COCO simulator consists on a freeware modular tool designed for flexibility and

extensibility, allowing users to integrate various components as needed. Instead, DWSIM is a fully open-source, code available, all-in-one process simulator platform.

Table 4: Overview of Chemical Process Simulation Software

Name	Company	Logo	License
Aspen Plus	AspenTech ^a		Commercial
Aspen Hysys ^b	AspenTech ^a		Commercial
Petro-SIM ^b	KBC Advanced Technologies		Commercial
UniSim Design ^b	Honeywell		Commercial
gPROMS	Siemens		Commercial
ChemCAD	Chemstations		Commercial
ProSimPlus	Fives		Commercial
Symmetry	Schlumberger		Commercial
AVEVA PRO/II	AVEVA		Commercial
COCO	AmsterCHEM		Free
DWSIM	Daniel Wagner		Open

^a AspenTech has very recently been acquired by the global engineering corporation Emerson Electric: (AspenTech 2025)

^b Build on the HYSYS 3.1 core, a following AspenTech's forced divestiture of its 2002 acquisition of Hyprotech to restore competition in the process engineering simulation software market (Federal Trade Commission 2004)

4 Methodology & Implementation

In this chapter, the methodology followed during the development of a Decision Support System (DSS) for Process Systems Engineering is presented. The application has been developed by means of a data driven approach and as illustrated in Figure 11, the procedure is structured into five stages, being the stage 1 the graph-based database with all the available process information and stage 5 the final simulation of the identified chemical upcycling process.

The entire methodology has been designed and implemented using open-source tools, adhering to the principles of open science. This approach ensures transparency, reproducibility, and accessibility of the research development and outcomes. By leveraging freely available technologies, the work not only promotes collaborative development but also facilitates future extensions and validation by the broader scientific community.

Although different tools will be used in the methodology, including a graph-based database (Neo4J), an AML (Pyomo) and a Chemical Process Simulation Software (DWSIM), the entire methodology implementation will be centralized from Python Programming Language for being a language largely used for the process systems community and with a high number of useful public libraries and APIs for the use of external applications.

Additionally, a simple graphical web-based user interface has been developed to support user interaction with the DSS, see Annex A for details.

In the following subsections the different stages will be discussed, the stages are as follows:

- **Stage 1:** Graph Database
- **Stage 2:** Reduced Graph
- **Stage 3:** Superstructure Optimization
- **Stage 4:** Superstructure Resolution
- **Stage 5:** Chemical Process Simulation

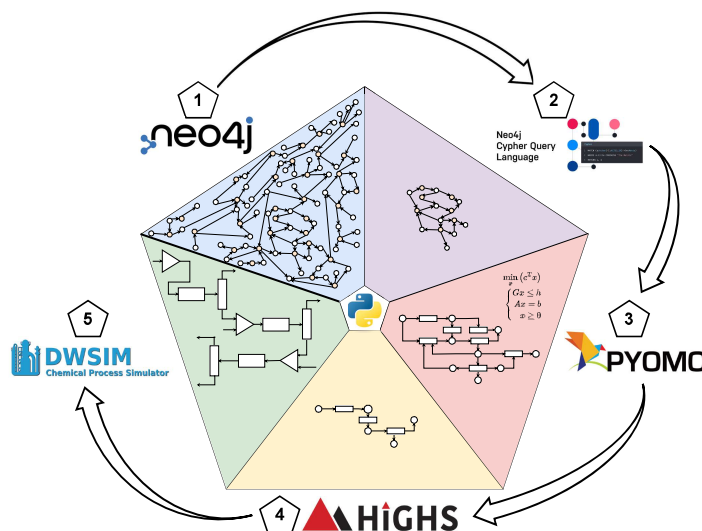


Figure 11: Methodological steps of the procedure

4.1 Graph Database

The first step for the methodology to be described is to organize the available data for the application, in that case process data. As previously discussed in the subsection 3.2 of the State of the Art, a wide range of DBMS exist for managing data, having each one of them their pros and cons. Among them, Graph-based databases are the most convenient for highly interconnected data, as it is the case for process data. Two different typologies of graph-based databases have been as well presented previously: Property Graph (PG) and Resource Description Framework (RDF)/ Triple Stores.

In this study, a Property Graph (PG) database will be used, as it is not aimed to connect to the Web 3.0 Linked Open Data Cloud, but rather to handle internal data derived from various chemical processes. A PG consists on a set of nodes and edges, where each node and edge can store attributes and is associated with one or more labels that define the class (or classes) to which pertain. This structure of information permits to efficiently represent process data.

Among the different PG DBMS, Neo4j is chosen for its strong performance, usability, and mature ecosystem. Although not being fully open-source, the Community Edition is sufficiently robust for this project. Neo4j supports efficient modeling and querying via Cypher, and its active community offers extensive resources, making it a practical and scalable solution for managing chemical process data.

The data model is designed with two core node types: Material and Process, following a similar structure to the one found in STN and P-graph methodologies presented previously in the subsection 3.3 of the State of the Art. This dual-node approach enables an unambiguous representation of production systems, ensuring a clear distinction between substances and the operations that transform them. Two types of relationships connect these nodes: Consumed_by and Produced_by, which respectively capture the consumption and production of materials in processes.

Figure 12 illustrates the structure of the PG used in this study, including the node and edges types, their attributes, and the directed relationships between them. It is important to state that the attributes included in the data model consist just on the required for the following steps but potentially they could include additional information such as physical and chemical properties, descriptions, etc.

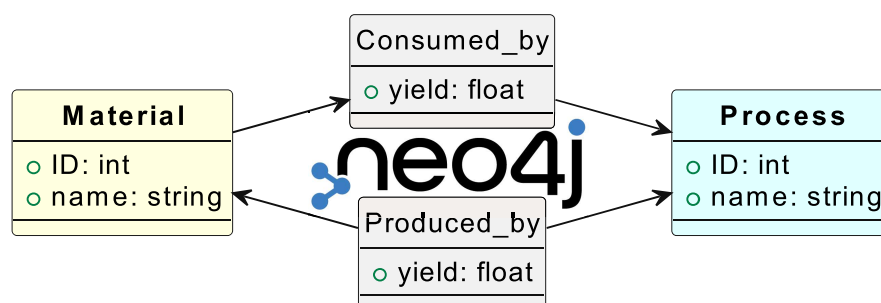


Figure 12: Graph Data Model

4.2 Reduced Graph

With the information organized in the Neo4J graph database as described in the previous subsection, the goal is to identify the most favorable pathway to synthesize a specific product of interest. Given that the database may contain a large number of interconnected processes, it is essential to reduce the graph by extracting only the necessary information required to proceed with the methodology.

This graph reduction, illustrated in Figure 13, constitutes the second step of the methodology. It is carried out using Cypher, the query language of Neo4J, which enables selective extraction from this highly interconnected data structure. Specifically, the query aims to identify all possible pathways that lead from the set of available materials to the target product defined by the user, depending on the specific case study.

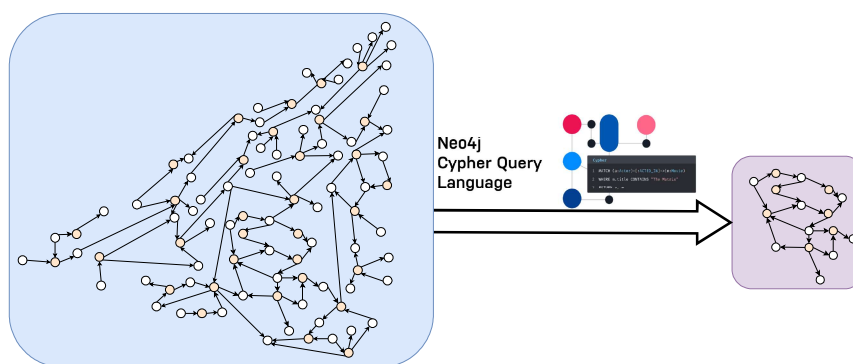


Figure 13: Graph reduction step

The Cypher query, shown in Figure 14, follows a multi-step procedure.

First, it identifies all possible paths connecting the user-defined available_materials to the main_products, also specified by the user. This search is constrained by a maximum number of hops, determined by the parameter path_length.

The path_length acts as a heuristic, set by the user, to limit the traversal depth. Longer paths are generally less favorable, as they may correspond to inefficient or impractical synthesis routes. Given the alternating structure of the graph, where materials and processes are represented as separate node types, the number of hops is twice the number of actual transformation steps.

Since the processes in the graph follow a Multiple Input Multiple Output (MIMO) structure, it is essential to apply a filtering step to ensure the feasibility of each path. Specifically, the goal is to exclude any paths in which a process node consumes input materials that are neither part of the path itself nor included in the initial set of available materials. This validation step ensures the chemical and logistical feasibility of the pathways by enforcing material balance constraints and avoiding reliance on undefined or external resources.

Finally, the valid pathways are returned, including the material and process nodes and the edges that represent the flow of materials. All associated metadata stored in the database is preserved for subsequent analysis.

It is important to note, that the purposed query can be adapted by the user to realize the selection with a different criteria if needed for his specific use case.

```

1  // Step 1: Select all paths up to path_length steps from
   available_materials to main_products
2  // Note: path_length corresponds to 2*n steps due to
   alternating material and processes nodes
3  MATCH path = (startMat:Material)
4    -[*..path_length]->(endMat:Material)
5  WHERE startMat.id IN available_materials
6    AND endMat.id IN main_products
7  // Step 2: Filter out paths where any process uses unavailable
   or undefined inputs
8  WITH nodes(path) AS pathNodes
9  WITH [n IN pathNodes WHERE n:Material] AS pathMaterials,
10     [n IN pathNodes WHERE n:Process] AS pathProcesses
11  UNWIND pathProcesses AS proc
12  MATCH (proc)-[:Produced_by]->(secondaryOutput:Material)
13  WHERE NOT secondaryOutput IN pathMaterials
14  WITH pathProcesses, pathMaterials, collect(DISTINCT
   secondaryOutput) AS secondaryOutputs
15  WITH pathProcesses, pathMaterials + secondaryOutputs AS
   expandedMaterials
16  WHERE ALL(proc IN pathProcesses WHERE
17     ALL(inMat IN [(inMat)-[:Consumed_by]->(proc) | inMat]
18       WHERE inMat.id IN available_materials
19       OR inMat IN expandedMaterials))
20  // Step 3: Return valid process steps with their input/output
   materials and connections
21  UNWIND pathProcesses AS processInPath
22  MATCH (in:Material)-[r_in:Consumed_by]->(processInPath)
23    -[r_out:Produced_by]->(out:Material)
24  RETURN DISTINCT processInPath,
25     collect(DISTINCT in) AS inputs,
26     collect(DISTINCT out) AS outputs,
27     collect(DISTINCT r_in) AS connections_in,
28     collect(DISTINCT r_out) AS connections_out

```

Figure 14: Cypher query for the graph reduction

The underlined placeholders denote user-defined parameters

4.3 Superstructure Formalization

The query presented in the previous subsection effectively reduces the solution space by extracting the subgraph of interest. This subgraph constitutes a superstructure, a representation of all plausible routes extracted from the database, however within this superstructure, multiple feasible pathways may still exist. Among these, the objective is to identify the most favorable one based on well-defined criteria.

To achieve this, the problem is further structured through a mathematical programming or optimization formulation. This allows the incorporation of constraints and objective functions that reflect the decision-making criteria, as discussed in Subsection 3.3 of the State of the Art.

The given subgraph, extracted from Neo4J graph database via the Cypher Query as described earlier comprises two types of nodes: Material nodes and Process nodes. These nodes are interconnected by directed edges of types Consumed_by and Produced_by, which represent the directional flow of materials into and out of processes.

Within the mathematical programming framework, the graph-based representation is reformulated using indexed sets. Materials are denoted by the set I and indexed by $i \in I$, while processes are denoted by the set P and indexed by $p \in P$. This abstraction provides the basis for defining the decision variables, parameters, constraints, and objective function.

Based on this structure, the optimization problem is defined using a single decision variable: the production level. Mathematically denoted as $X_p \in \mathbb{R}^+ \quad \forall p \in P$, indicates the level at which each process p is carried out.

In order to represent how materials are transformed across the network, material balances are formulated based on process-specific yields. These balances are defined using the yield parameter $\eta_{ip} \quad \forall i \in I \quad \forall p \in P$, which corresponds to the conversion rate of material i in process p . This parameter is derived from the yield attribute of the directed edges in the original graph structure. Edges of type Consumed_by are assigned negative values, indicating material consumption, while edges of type Produced_by are assigned positive values, indicating material production. This sign convention captures the directionality of material flows and allows the construction of an adjacency-like matrix that links processes and materials.

Using this convention, the net balance of material i across all processes, representing its total production (if positive) or consumption (if negative) is given by Equation 1.

$$\sum_{p \in P} X_p \cdot \eta_{ip} \quad \forall i \in I \quad (1)$$

This net balance is constrained by user-defined bounds corresponding to market constraints on offer and demand.

For materials that are intended to be produced (i.e., with a positive net output), the model imposes a demand constraint to ensure production falls within acceptable bounds.

Specifically, the total production must satisfy a minimum demand while not exceeding a maximum allowable quantity. These bounds are represented by $D_i^{\min} \quad \forall i \in I$ and $D_i^{\max} \quad \forall i \in I$, respectively, as expressed in constraint 2.

$$D_i^{\min} \leq \sum_{p \in P} X_p \cdot \eta_{ip} \leq D_i^{\max} \quad \forall i \in I \quad (2)$$

Similarly, for materials that are consumed (i.e., with negative net output), the model enforces a supply constraint that limits the material's availability. These constraints are defined by minimum and maximum supply values, denoted as $S_i^{\min}$ and $S_i^{\max}$ for all $i \in I$, and are given in constraint 3.

$$-S_i^{\max} \leq \sum_{p \in P} X_p \cdot \eta_{ip} \leq -S_i^{\min} \quad \forall i \in I \quad (3)$$

A unified constraint for all materials can be formulated by combining the conditions for production and consumption as shown in equation 4.

$$(D_i^{\min} - S_i^{\max}) \leq \sum_{p \in P} X_p \cdot \eta_{ip} \leq (D_i^{\max} - S_i^{\min}) \quad \forall i \in I \quad (4)$$

The objective function (equation 5) of the optimization model is to minimize a weighted sum of the process activity levels. Each process $p \in P$ is associated with a weighting coefficient c_p , which is specified by the user. This coefficient can be used to reflect various criteria, such as economic cost, environmental impact, or the technological readiness of the process.

$$\min_{X_p} \sum_{p \in P} c_p \cdot X_p \quad (5)$$

The resulting formulation serves as a generic optimization model, which can be adapted or extended to accommodate specific modeling requirements (constraints) or user-defined objectives.

For the formulation and implementation of the described optimization model, the Pyomo AML is selected for being open-sourced and integrated with Python programming language, the language that is used for the centralization of the whole methodology. In figure 15 the result of that step of the methodology can be visualized.

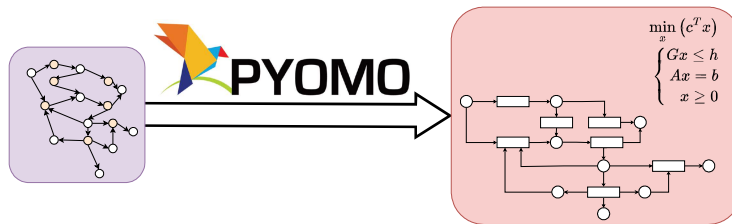


Figure 15: Superstructure formalization step

4.4 Superstructure Optimization

After the formalization of the superstructure by means of mathematical programming, the following step consists on the optimization of the same. In order to resolve it a solver, what is a software package with the implementation of mathematical optimization algorithms is required. A huge variety of solvers exist and in order to select an appropriate one for the resolution of a mathematical formulation it is important to study the typology of the same.

The mathematical formulation presented in the previous subsection corresponds to a Linear Programming (LP). LP problems are those where both the constraints (in that case equation 4) and objective function (equation 5) correspond to continuous linear functions.

Given that LP problems belong to the computationally efficient P complexity class, the choice of the solver is not critical. Therefore, the HiGHS solver is selected due to its open-source nature, ease of integration with most AML, including Pyomo, and its specific design for solving LP problems.

Being the problem an LP, the production level $X_p \quad \forall p \in P$ can continuously take any non-negative value (both positive and zero). If the production level is zero then the process won't be selected and if it is positive it is activated to the extend indicated by the same.

This, added up to the MIMO structure, where the processes may need more than one input and produce more than one output means that not necessarily a path, what is a sequence of processes from the raw material towards the final product will be found. Instead, a system of processes will be found where there can be ramifications if there is demand for more than one product and also partial or total recycling of the co-products produced by the different processes from which there is not demand.

However, with the current objective function as defined in the equation 5 a single path will be selected as the objective is to minimize the total production activity, while fulfilling the required amount.

In Figure 16 a representation of the expected output of this step can be visualized.



Figure 16: Superstructure resolution step

4.5 Chemical Process Simulation

Last but not least, it is important to perform a simulation of the process by means of a Chemical Process Simulation Software. As discussed in the subsection 3.4 of the State of the Art, expert decision-making in Chemical Process Industry is usually linked with the use of detailed flowsheets. In that context the automatic construction of a preliminary schema in a Simulation Software, from which experts would be able to evaluate the performance of the system, is a feature that can be of great utility in the context of a Decision Support System (DSS).

Among the different alternative software for its construction DWSIM simulator is selected, for being a fully open-source CAPE-OPEN simulator with available documentation and integration with Python programming language through Python.NET interface.

An ideal isobaric simulation approach is adopted for modeling each process. To represent the thermodynamic behavior of the system, Raoult's Law is used as the fluid package, a model within the simulation software that defines how properties such as phase equilibrium and mixture behavior are calculated. This choice aligns with the ideal assumptions made for this preliminary modeling stage.

The input materials are introduced as separate feed streams and, if more than one input is present, they are combined using a Mixer unit. The resulting stream enters an isothermal conversion reactor operating at 100% efficiency, where only the specified reaction occurs. Since the reactor in DWSIM produces separate liquid and gas outputs, and the distinction between phases is beyond the scope of this work, these streams are recombined using an additional Mixer unit, referred to as the "Imaginary Mixer". The unified stream, in the case of being composed by more than one material flows into a series of Ideal Isothermal Separators, each with a separation efficiency of 100%. The separator train includes $n - 1$ units, where n corresponds to the number of desired output components, ultimately producing n individual product streams. A graphical representation of a sample process constructed in DWSIM is provided in Figure 17.

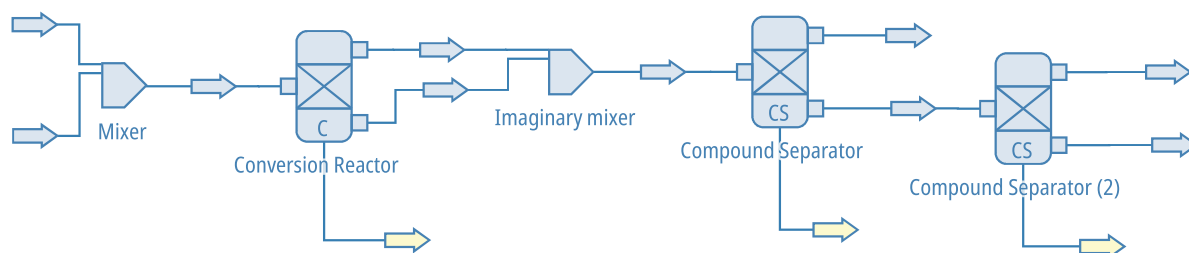


Figure 17: Representation of a process in DWSIM

The material flows inputs and outputs of the different process units as well as the market inputs and outputs obtained through the mathematical programming resolution have to be connected in order to construct the network of processes. In order to do so a structure

such as the one depicted in Figure 18 have to be build for each one of the materials involved in the system. The structure consist of a Mixer that will be present when there is more than one source for a given material (market and/or processes) and a Splitter that will be present when there is more than one sink (market and/or processes).

Additionally, Recycle Logical Operations are necessary in Sequential-Modular Simulation software like DWSIM whenever there is recirculation. Recirculation creates circular dependencies, which require iterative calculations. In such cases, Recycle Blocks are inserted at appropriate positions to ensure that the simulation can be solved with minimal computational cost. To identify where these Recycle Blocks are needed, cycles in the process network are detected using the algorithm developed by D. B. Johnson (1975), implemented via the `simple_cycles` function from the NetworkX Python library (Hagberg, Schult, and Swart 2008). For each detected cycle, one material node is randomly selected, and a Recycle Block is added at that point.

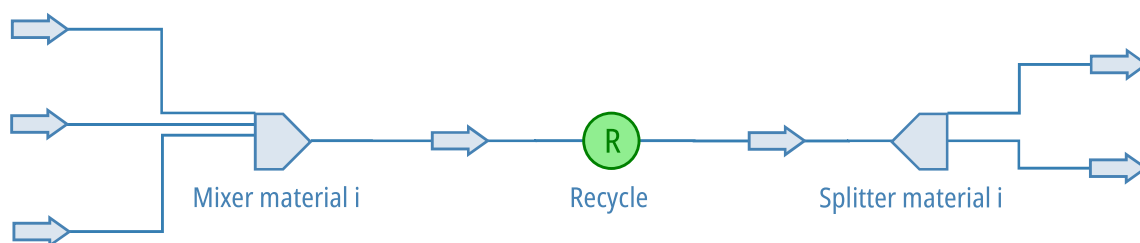


Figure 18: Representation of a material connections in DWSIM

Altogether this step builds from the Mathematical Programming result obtained in the previous step, a preliminary representation of the process, as illustrated in Figure 19. The designed tool can be further extended in order to build more rigorous an accurate simulation considering the state of matter, its thermodynamics, the kinetics, the type of separation procedure, etc.

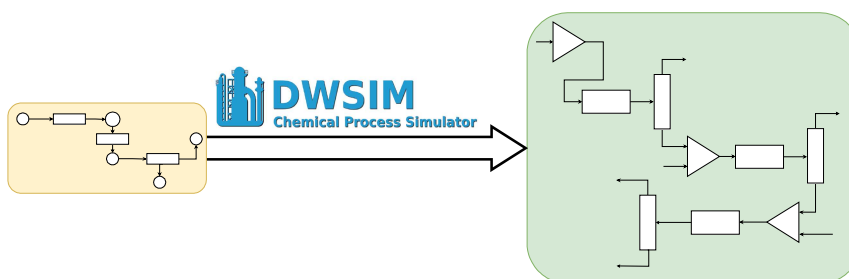


Figure 19: Chemical Process Network construction

5 Problem Definition

Given a set of available processes with input-output information, which could potentially be very large the optimal pathway/configuration is to be found to transform a set of available wastes into one or more products of interest. For the scope of this project, 40 processes will be included as illustrated in Table 5. These processes perform transformations involving 27 distinct materials as listed in Table 6.

Table 5: Processes Included in the System

Name		Reactants		Products
p01	Electrolysis	$2 \text{H}_2\text{O}$	$\longrightarrow$	$2 \text{H}_2 + \text{O}_2$
p02	Water-Gas Shift	$\text{CO} + \text{H}_2\text{O}$	$\longrightarrow$	$\text{CO}_2 + \text{H}_2$
p03	Reverse Water-Gas Shift	$\text{CO}_2 + \text{H}_2$	$\longrightarrow$	$\text{CO} + \text{H}_2\text{O}$
p04	Boudouard Reaction	2CO	$\longrightarrow$	$\text{CO}_2 + \text{C}$
p05	Reverse Boudouard Reaction	$\text{CO}_2 + \text{C}$	$\longrightarrow$	2CO
p06	Sabatier Reaction	$\text{CO}_2 + 4 \text{H}_2$	$\longrightarrow$	$\text{CH}_4 + 2 \text{H}_2\text{O}$
p07	Reverse Sabatier Reaction	$\text{CH}_4 + 2 \text{H}_2\text{O}$	$\longrightarrow$	$\text{CO}_2 + 4 \text{H}_2$
p08	Steam Reforming	$\text{CH}_4 + \text{H}_2\text{O}$	$\longrightarrow$	$\text{CO} + 3 \text{H}_2$
p09	Reverse Steam Reforming	$\text{CO} + 3 \text{H}_2$	$\longrightarrow$	$\text{CH}_4 + \text{H}_2\text{O}$
p10	Dry Methane Reforming	$\text{CH}_4 + \text{CO}_2$	$\longrightarrow$	$2 \text{CO} + 2 \text{H}_2$
p11	Methane Pyrolysis	CH_4	$\longrightarrow$	$\text{C} + 2 \text{H}_2$
p12	Partial Methane pyrolysis	2CH_4	$\longrightarrow$	$\text{C}_2\text{H}_2 + 3 \text{H}_2$
p13	Bosch Reaction	$\text{CO}_2 + 2 \text{H}_2$	$\longrightarrow$	$\text{C} + 2 \text{H}_2\text{O}$
p14	Methane Combustion	$\text{CH}_4 + 2 \text{O}_2$	$\longrightarrow$	$\text{CO}_2 + 2 \text{H}_2\text{O}$
p15	Methane Partial Combustion	$2 \text{CH}_4 + \text{O}_2$	$\longrightarrow$	$\text{CO} + 2 \text{H}_2$

	Name	Reactants		Products
p16	Methanol Synthesis 1	$\text{CO} + 2 \text{H}_2$	$\longrightarrow$	CH_3OH
p17	Methanol Synthesis 2	$\text{CO}_2 + 3 \text{H}_2$	$\longrightarrow$	$\text{CH}_3\text{OH} + \text{H}_2\text{O}$
p18	Methyl Formate Synthesis	$\text{CH}_3\text{OH} + \text{CO}$	$\longrightarrow$	HCOOCH_3
p19	Hydrolysis Methyl Formate	$\text{HCOOCH}_3 + \text{H}_2\text{O}$	$\longrightarrow$	$\text{HCOOH} + \text{CH}_3\text{OH}$
p20	Formic Acid Synthesis	$\text{CO} + \text{H}_2\text{O}$	$\longrightarrow$	HCOOH
p21	Formaldehyde Synthesis	$2 \text{CH}_3\text{OH} + \text{O}_2$	$\longrightarrow$	$2 \text{HCHO} + 2 \text{H}_2\text{O}$
p22	Methanol Dehydration	$2 \text{CH}_3\text{OH}$	$\longrightarrow$	$\text{CH}_3\text{OCH}_3 + \text{H}_2\text{O}$
p23	Methanol Dehydrogenation	CH_3OH	$\longrightarrow$	$\text{HCHO} + \text{H}_2$
p24	Ethylene Synthesis	$2 \text{CO}_2 + 6 \text{H}_2$	$\longrightarrow$	$\text{C}_2\text{H}_4 + 4 \text{H}_2\text{O}$
p25	Cloroalkali Electrolysis	$2 \text{NaCl} + 2 \text{H}_2\text{O}$	$\longrightarrow$	$\text{Cl}_2 + 2 \text{H}_2 + 2 \text{NaOH}$
p26	Direct Hydrogen Chloride Synthesis	$\text{H}_2 + \text{Cl}_2$	$\longrightarrow$	HCl
p27	Organic Hydrogen Chloride Synthesis	$\text{CH}_4 + \text{O}_2 + 2 \text{Cl}_2$	$\longrightarrow$	$\text{CO}_2 + 4 \text{HCl}$
p28	Ethylene Chlorination 1	$\text{C}_2\text{H}_4 + \text{Cl}_2$	$\longrightarrow$	CHCl_2CH_3
p29	Ethylene Chlorination 2	$\text{C}_2\text{H}_4 + \text{Cl}_2$	$\longrightarrow$	$\text{CH}_2\text{ClCH}_2\text{Cl}$
p30	Tetrachloroethylene Synthesis 1	$\text{CHCl}_2\text{CH}_3 + 3 \text{Cl}_2$	$\longrightarrow$	$4 \text{HCl} + \text{C}_2\text{Cl}_4$
p31	Tetrachloroethylene Synthesis 2	$\text{CH}_2\text{ClCH}_2\text{Cl} + 3 \text{Cl}_2$	$\longrightarrow$	$4 \text{HCl} + \text{C}_2\text{Cl}_4$
p32	Ethylene Oxychlorination 1	$2 \text{C}_2\text{H}_4 + 4 \text{HCl} + \text{O}_2$	$\longrightarrow$	$2 \text{CHCl}_2\text{CH}_3 + 2 \text{H}_2\text{O}$
p33	Ethylene Oxychlorination 2	$2 \text{C}_2\text{H}_4 + 4 \text{HCl} + \text{O}_2$	$\longrightarrow$	$2 \text{CH}_2\text{ClCH}_2\text{Cl} + 2 \text{H}_2\text{O}$
p34	Vinyl Chloride Synthesis 1	CHCl_2CH_3	$\longrightarrow$	$\text{C}_2\text{H}_3\text{Cl} + \text{HCl}$

Name		Reactants		Products
p35	Vinyl Chloride Synthesis 2	$\text{CH}_2\text{ClCH}_2\text{Cl}$	$\longrightarrow$	$\text{C}_2\text{H}_3\text{Cl} + \text{HCl}$
p36	Trichloroethylene Synthesis 1	$\text{CHCl}_2\text{CH}_3 + \text{Cl}_2$	$\longrightarrow$	$\text{C}_2\text{HCl}_3 + 3 \text{HCl}$
p37	Trichloroethylene Synthesis 2	$\text{CH}_2\text{ClCH}_2\text{Cl} + \text{Cl}_2$	$\longrightarrow$	$\text{C}_2\text{HCl}_3 + 3 \text{HCl}$
p38	Ethylene Hydration	$\text{C}_2\text{H}_4 + \text{H}_2\text{O}$	$\longrightarrow$	$\text{CH}_3\text{CH}_2\text{OH}$
p39	Ethylene Oxide Synthesis	$2 \text{C}_2\text{H}_4 + \text{O}_2$	$\longrightarrow$	$2 \text{CH}_2\text{OCH}_2$
p40	Vinyl Acetate Synthesis	$2 \text{C}_2\text{H}_4 + 2 \text{CH}_3\text{COOH} + \text{O}_2$	$\longrightarrow$	$2 \text{CH}_3\text{CO}_2\text{CHCH}_2 + 2 \text{H}_2\text{O}$

Table 6: Materials Involved in the Process Network

	CAS number	Name
H ₂ O	7732-18-5	Water
H ₂	1333-74-0	Hydrogen
O ₂	7782-44-7	Oxygen
CO	630-08-0	Carbon monoxide
CO ₂	124-38-9	Carbon dioxide
C	7440-44-0	Carbon
CH ₄	74-82-8	Methane
C ₂ H ₂	74-86-2	Acetylene
C ₂ H ₄	74-85-1	Ethylene
CH ₃ OH	67-56-1	Methanol
CH ₃ CH ₂ OH	64-17-5	Ethanol
HCOOCH ₃	107-31-3	Methyl formate
HCOOH	64-18-6	Formic acid
CH ₃ COOH	64-19-7	Acetic acid
HCHO	50-00-0	Formaldehyde
CH ₃ OCH ₃	115-10-6	Dimethyl ether
NaCl	7647-14-5	Sodium chloride
NaOH	1310-73-2	Sodium hydroxide
Cl ₂	7782-50-5	Chlorine
HCl	7647-01-0	Hydrogen chloride
CHCl ₂ CH ₃	75-34-3	1,1-dichloroethane
CH ₂ ClCH ₂ Cl	107-06-2	1,2-dichloroethane
C ₂ Cl ₄	127-18-4	Tetrachloroethylene
C ₂ H ₃ Cl	75-01-4	Vinyl chloride
C ₂ HCl ₃	79-01-6	Trichloroethylene
CH ₂ OCH ₂	75-21-8	Ethylene oxide
CH ₃ CO ₂ CHCH ₂	108-05-4	Vinyl acetate

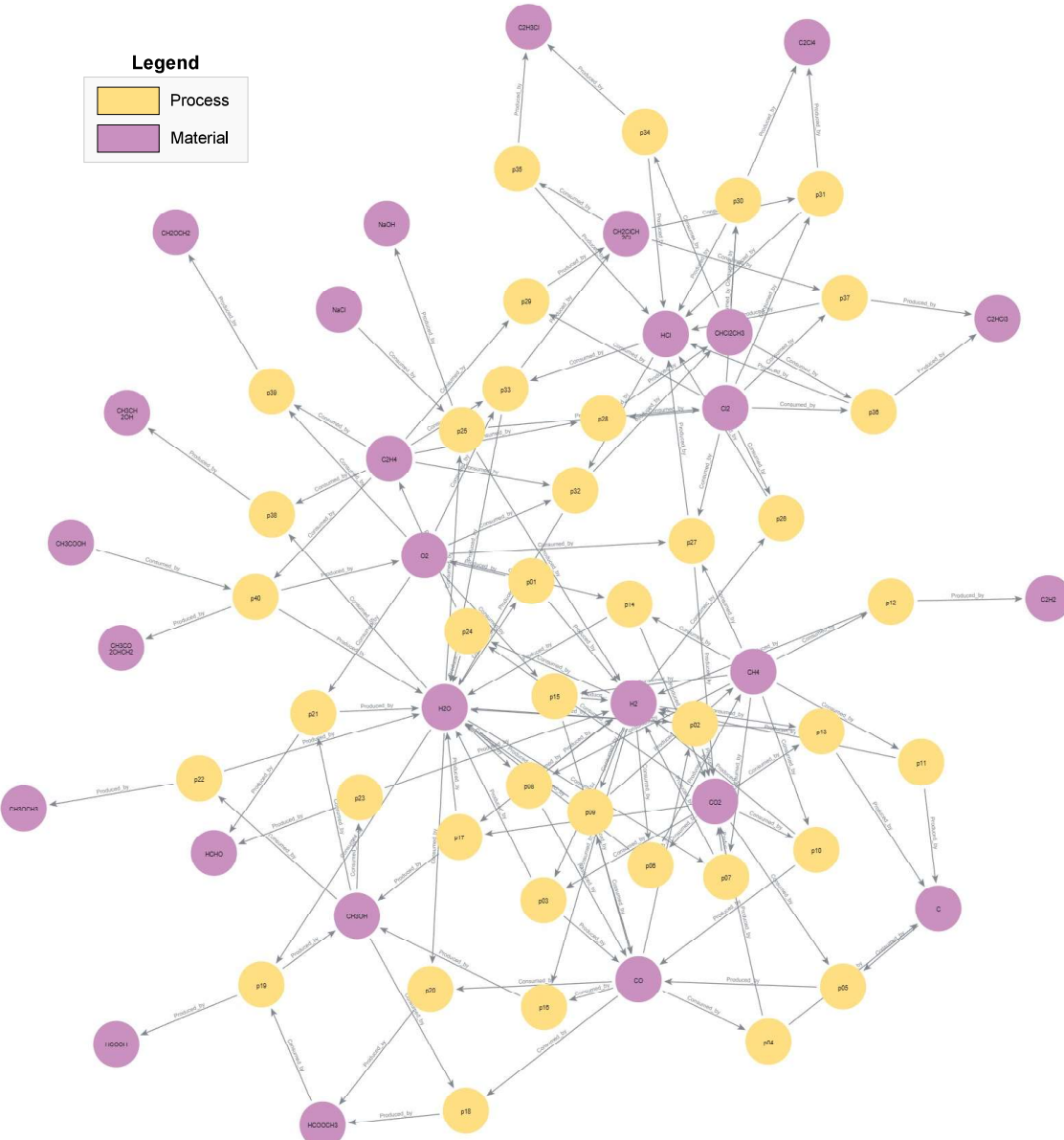


Figure 20: Graph database

6.1 Base Case

As a base case it is intended to find the best pathway to synthesize Methanol (CH_3OH), an organic compound with a great importance in industry as raw material, solvent and fuel. Carbon dioxide (CO_2), water (H_2O) and sodium chloride (NaCl) are selected as starting materials. While CO_2 serves as the carbon source and H_2O provides hydrogen (directly or indirectly), NaCl can play a supporting role in electrochemical processes.

Once the data has been introduced in the database the following step is to retrieve the relevant data and so reducing the graph as explained in Subsection 4.2 of the Methodology. In order to do so it is necessary to define the list available_materials = [" CO_2 ", " H_2O ", " NaCl "] and the list main_products = [" CH_3OH "].

It is also necessary to define the number of steps path_length which correspond to the double of the number of involved processes as material nodes are also accounted as steps in the query. The number of steps is determined as the minimum one to guarantee the production of Methanol from the available raw materials resulting in a value of path_length = 4, what is two processes of depth at most in each pathway.

With the application of the query the graph is effectively reduced to 11 nodes (3 process nodes and 8 material nodes) and 12 relationships what is a 16.4% of the size of the original graph database as can be visualized in Figure 21.



Figure 21: Reduced graph for the base case

After reducing the graph, mathematical programming is applied to identify the optimal configuration, as defined by the model presented in the Methodology section. The objective is to minimize the total cost, which requires to define the cost coefficients c_p that are multiplied by the production levels X_p of each process $p \in P$ preselected by the previously explained query.

These costs, can be fixed according to economical and/or environmental expenses as well as other possibilities such as a weighting according to the Technology Readiness Level (TRL). In a first approach they are fixed to $c_p = 1 \quad \forall p \in P$ considering that any specific technology is prioritized over others.

It is also essential to define the minimum and maximum supply and demand constraints for the materials relevant to the case study. For the main_products, in that case just Methanol (CH_3OH), a minimum demand ($D_i^{\min}$) of 500 units and a maximum demand ($D_i^{\max}$) of 2000 units are specified. For available_materials, the supply is allowed to vary between a minimum ($S_i^{\min}$) of 0 and a maximum ($S_i^{\max}$) of 2000 units. For all other products, both the minimum and maximum supply are set to 0 (i.e., they cannot be externally supplied), while their demand is allowed to range from 0 to 2000 units.

By setting a maximum demand of 2000 (a number considered big enough), the model allows for flexibility in handling coproducts or intermediates that may be produced within the network but are not entirely consumed. It is assumed that any surplus can be absorbed by the market without issue.

After defining all the necessary parameters, the resulting optimal configuration is shown in Figure 22. However, it can be observed that there are two distinct optimal configurations, illustrated in Figures 22a and 22b, both of them have the same value for the objective function, and the solver may converge in either of them depending on the initial approximation.

This behavior occurs because the solution to the defined case study is degenerate; that is, the slope of the objective function is coincident with one of the edges, planes, or hyperplanes of the feasible region.

As shown in Figure 22, in the given case study, the optimization involves selecting between processes p01 and p25 for the production of the required hydrogen in the subsequent methanol synthesis carried out by process p17.

To better visualize this situation, a representation of the constraints affecting p01 and p25 is provided in Figure 23, based on fixing the production level of p17 to 500 units, as is the case in both optimal configurations. In this figure, it can be seen that the lower bound constraint associated with H_2 production is parallel to the isocost lines of the objective function. As a result, the optimization problem has infinitely many solutions along the line defined by a total production level of 750 units, being the options visualized in 22 the two extreme ones. Any combination of p01 and p25 producing that total whether all from one process or split between the two, is optimal.

However, in real-world engineering problems, it is uncommon for two different processes to have exactly the same cost. Typically, even similar technologies will differ slightly in economic or environmental impact, leading to distinct cost coefficients. As a result, this type of degeneracy, where multiple optimal configurations exist, generally does not occur. Nevertheless, it is important to highlight that even a slight difference in the cost associated with one of the processes will be sufficient for the solver to favor the alternative process with the lower cost, thus uniquely determining the optimal configuration.

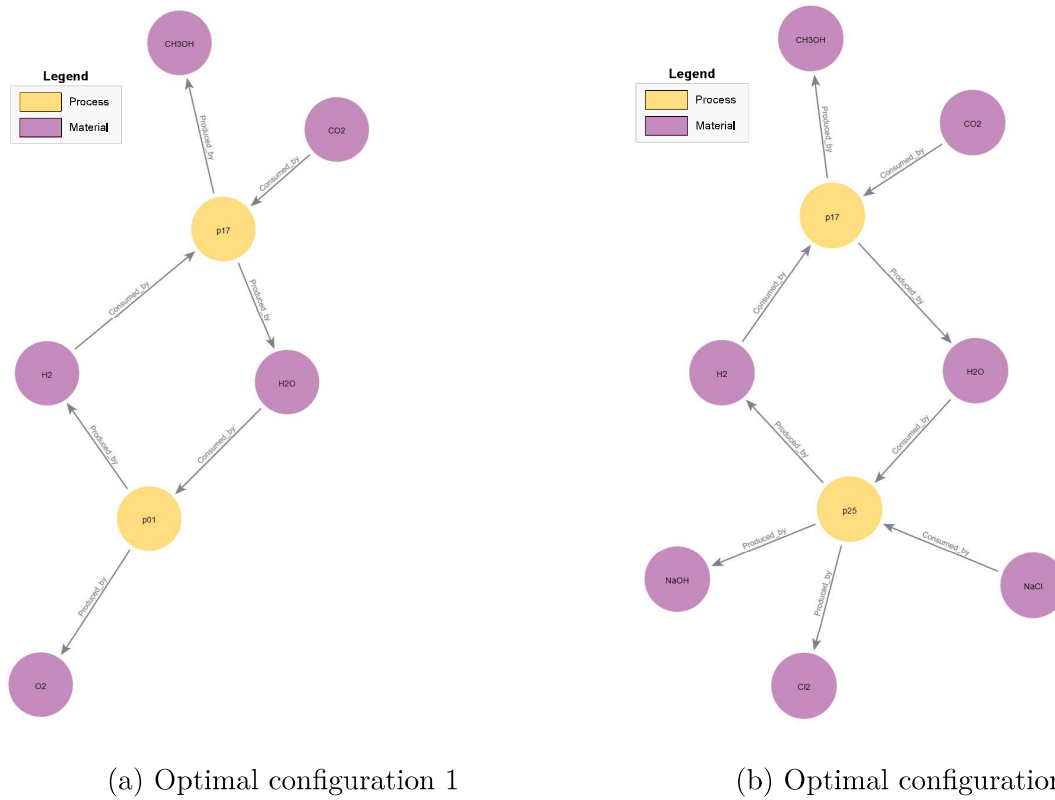
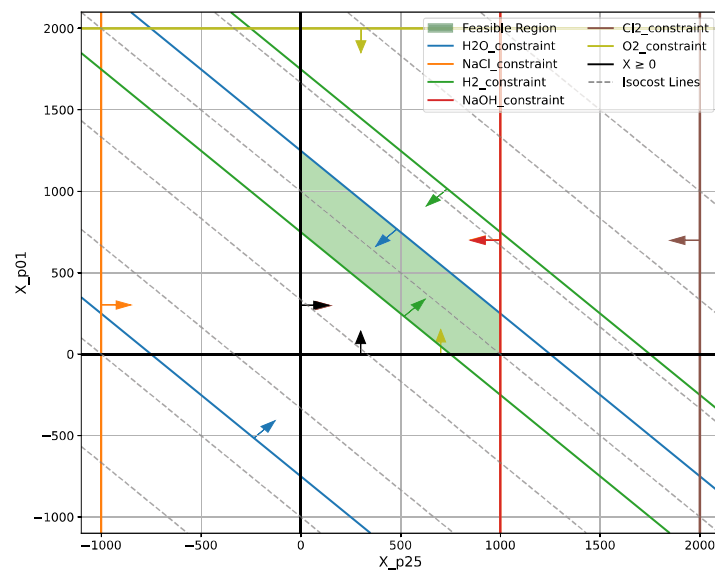


Figure 22: Optimal configurations for the base case



6.2 Alternative case

In the previous case, it has been observed that one of the available processes for the synthesis of the hydrogen necessary for the synthesis of methanol, in concrete the process p25 (Cloroalkali Electrolysis) produces chlorine (Cl_2) as a co-product. Chlorine is a resource of great utility for the market, and in this case its use for the production of hydrogen chloride (HCl) is purposed in order to extend the system purposed previously.

For that case, the parameters available_materials = ["CO₂", "H₂O", "NaCl"] and path_length=4 are maintained as defined in the previous subsection, while a new product is added in main_products = ["CH₃OH", "HCl"].

With this modification, the resulting reduced graph essentially corresponds to the one previously presented in Figure 21, with the addition of a new process (p26) for the production of hydrogen chloride (HCl) from hydrogen (H_2) and chlorine (Cl_2). This includes the corresponding connections, as well as a new node representing the material HCl , as shown in Figure 25.

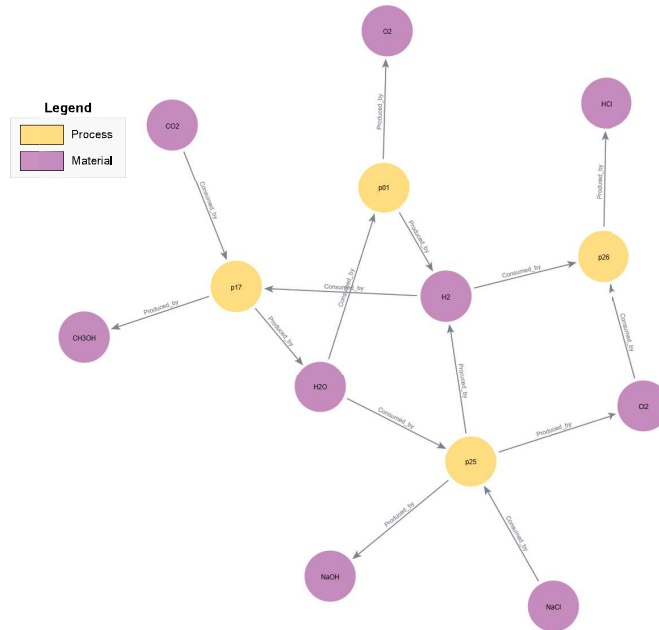


Figure 25: Reduced graph for the alternative case

To extract the optimal configuration, the offer, supply parameters, and process costs are defined as in the base case. The cost of all processes is set uniformly to $c_p = 1 \quad \forall p \in P$, so no specific technology is prioritized over others. The supply for each material in available_materials is allowed to vary between a minimum $S_i^{\min} = 0$ and a maximum $S_i^{\max} = 2000$ units. The demand for all main_products is defined to range between 500 and 2000 units, while the demand for co-products is limited to a maximum of 2000 units.

In this way, similarly to the base case, the solution is degenerate, with the two extreme configurations represented in Figure 26. In both configurations, processes p17 and p26 are included, corresponding to the production of methanol and hydrogen chloride, respectively. However, unlike the base case, process p25—used for hydrogen production, is not entirely replaced by p01 in the alternative extreme configuration. This is due to the fact that p25 also produces the chlorine required for process p26. As a result, p25 can either supply all the chlorine and hydrogen needed for processes p17 and p26 (26a), or provide the chlorine and part of the hydrogen, in combination with hydrogen supplied by p01 (26b).

This behavior can also be visualized in Figure 27, where the production levels of processes p26 and p17 are fixed and identical across all co-optimal solutions. The feasible region shows that the production level of process p25 cannot go below 250 units. Additionally, the sum of the production levels of processes p01 and p25 remains constant at 875 units across the extreme optimal solutions. This reflects the role of process p25 in supplying both hydrogen and chlorine required for the overall configuration.

The DWSIM simulation of the extreme optimal configurations 1 and 2 resulting of the methodology are constructed automatically and can be visualized in Figure 28.

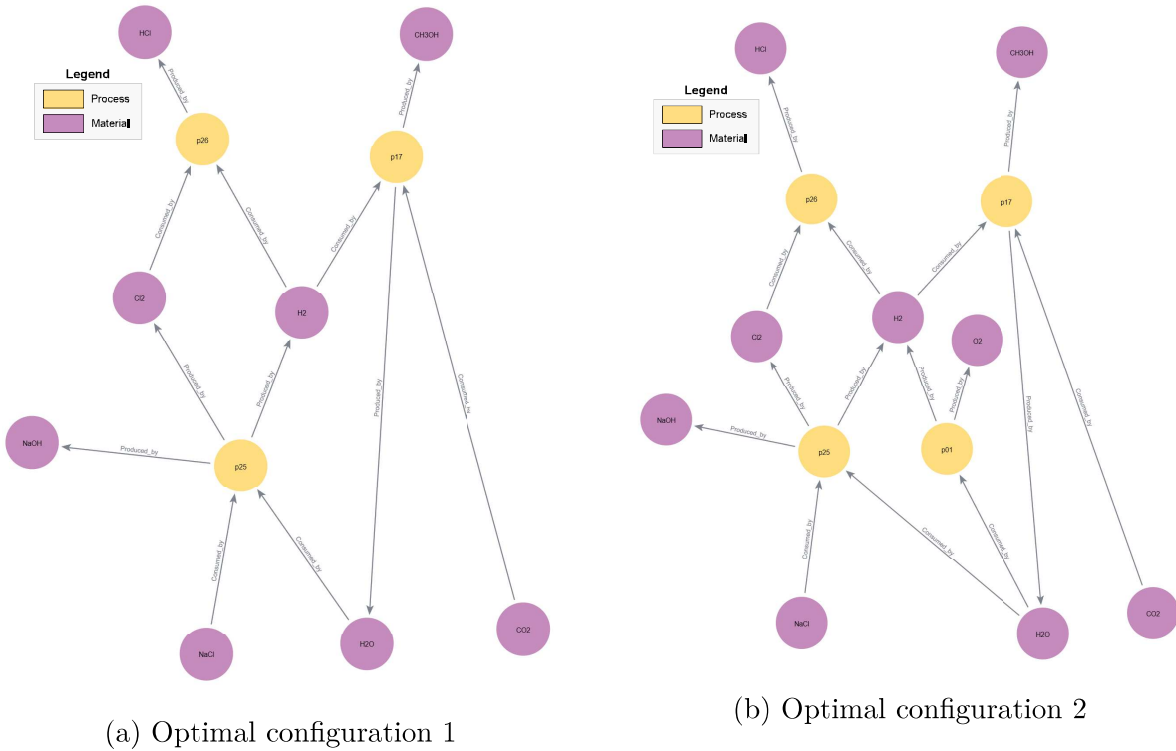


Figure 26: Optimal configurations for the alternative case

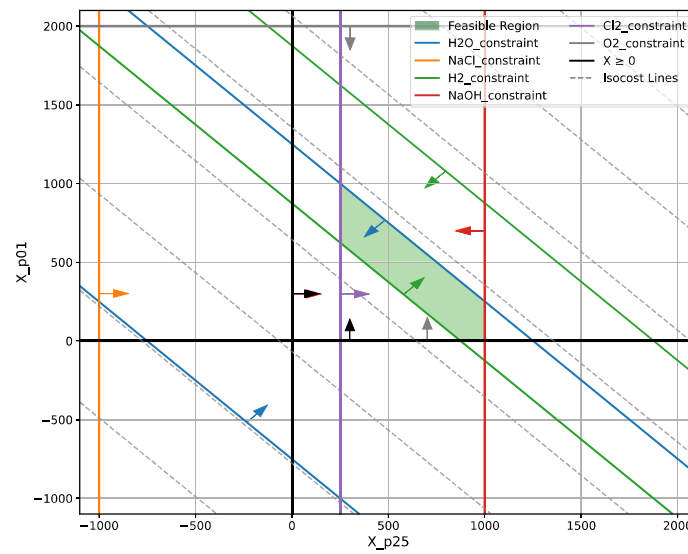
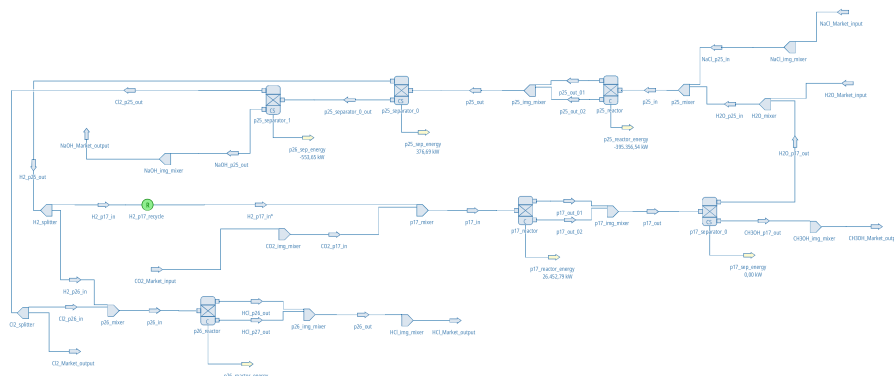
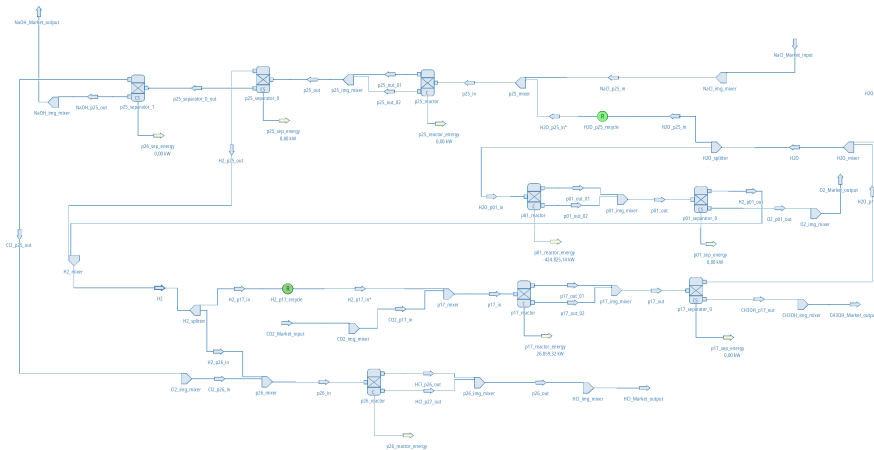


Figure 27: Feasible Region and Isocost Lines for the alternative case



(a) Flowsheet for the Optimal Configuration 1



(b) Flowsheet for the Optimal Configuration 2

Figure 28: Flowsheet for the Optimal Configurations for the base case

7 Conclusions

Summing up, this master's thesis has provided a fully open-source prototype whose modules are centralized and integrated through Python programming language. This prototype consists on a data-driven Decision Support System (DSS) for the automatic synthesis of pathways from available raw materials to specified target products.

The prototype follows a methodology consisting of five identified key steps: graph database generation, graph reduction, superstructure formalization, superstructure optimization and chemical process simulation.

Graph databases have demonstrated to be of great utility for the representation of production systems. In production systems, each process may involve multiple input and output materials, each of them potentially interacting in many transformation steps. These data structure, converts process data in a highly interconnected data with a graph structure that can be naturally represented within a graph database. Within graph databases, the node labeling functionality, that permits to classify the nodes, have permitted to generate a data model including process and material nodes as different nodes, facilitating the construction of an explicit and unambiguous data model.

Within graph database systems, Neo4j has been used for its large community and as although being not fully open source, the community edition is enough for the project scope. Neo4J graph database, can be queried by the use of Cypher which has permitted to effectively extract the subgraph of interest from the database. The selected subgraph corresponds to the specific network of materials and processes that connects the set of available raw materials to the desired target products, constrained by a limited number of transformation steps (hops).

The reduced graph obtained from the query into the database, in most of the cases contains more than one available pathway, effectively conforming a superstructure. This superstructure is mathematically formalized within a linear programming problem. The objective is to minimize the weighted sum of the process activity levels, that can be weighted according to the user priorities, such as economical cost or environmental impacts, subject to demand and supply constraints.

The mathematical programming formulation is implemented through Pyomo open-source Python-based software. To solve the resulting problem the HiGHS solver (also open-source) is employed, effectively obtaining the optimal solution according to the user set supply and demand, and its given objective.

Finally, a simulation of the obtained pathway is performed in an automatized way in DWSIM Process Simulator via PythonNET interface. This simulation consists on the main result of the DSS. Although, the model assumes ideal conditions, such as reaction and separation phases with full efficiency, it provides a more realistic representation of the process compared to the abstract optimal graph obtained in the previous step of the methodology. Moreover, it serves as an initial template from which a Process Systems Engineer can further develop and refine the process design of a fully viable production

pathway.

By the integration of those modules, a functional prototype of an automatic pathway synthesis tool has been successfully implemented. The code relative to its implementation, together with a web-based Graphical User Interface (GUI) will be published with its appropriate documentation at Zenodo open repository following the conclusion of the thesis embargo period.

Although, the tool permits to generate pathways among arbitrary materials, as set by the user and according to the data available, the motivation behind its implementation consists on finding W2R pathways. In that context an illustrative example has been presented for the synthesis of Methanol from carbon dioxide demonstrating the practical applicability and workflow of the tool.

As said, the developed tool consists on a prototype whose modules will be further enhanced in future iterations of the project to enhance the realism and utility of the tool. Future work may focus on expanding the process database, by increasing the number of processes and adding relevant information, incorporating more detailed models for the mathematical optimization and for the flowsheet design, as well as validating the tool utility through real-world industrial case studies.

8 Economical Analysis

Being according to the Ministerio de Inclusión, Seguridad Social y Migraciones (2025), the cost of Social Security of the 23.6 % for the contractor, and considering a gross salary of 20.00 €/h for and of 35.00 €/h for the supervisors, the Human Resources cost for the Master's Thesis is 31 209.00 €, as depicted in Table 7.

Table 7: Human Resources for the Master Thesis Development

Staff	Gross Salary (€/h)	Social Security (€/h)	Hours (h)	Cost (€)
Process Engineer	20.00 €/h	4.72 €/h	1000 h	24 720.00 €
Supervisor	35.00 €/h	8.26 €/h	125 h	5 407.50 €
Co-Supervisor	35.00 €/h	8.26 €/h	25 h	1 081.50 €
TOTAL	—	—	—	31 209.00 €

Being a computational based project and having used Open Source tools for its development no extra costs are used for the machinery, licensing, etc. According to the regulation by Universitat Politècnica de Catalunya (2024), an overhead of a 21% over the total is added for generalist software needs, as well as the use of space, electricity, internet and other utilities. Being the overall cost for the project of 39 505.06 € as illustrated in Table 8.

Table 8: Total Cost of the Project Development

Concept	Cost (€)
Human Resources	31 209.00 €
Specific Software	0.00 €
Overhead (21%)	8 296.06 €
TOTAL	39 505.06 €

This investment has enabled the development of a prototype tool capable of identifying pathways for upcycling industrial waste into valuable raw materials. Such functionality can contribute significantly to the planning and optimization of Eco-Industrial Parks, which typically involve substantial capital investments from hundreds to thousands of millions. Therefore, even modest efficiency improvements facilitated by the tool can yield meaningful economic returns.

9 Sustainability Analysis

According to a Life Cycle Assessment performed by Circular Ecology (2023), an hour of office works is equivalent to an average of 0.83 kgCO₂eq. With a total of 900 hours considering all the staff involved in the project, a total of **747 kgCO₂eq** have been produced.

The given footprint consists on the environmental cost of conducting the research necessary to develop a DSS for the synthesis of Pathways among materials. Although, this methodology can be adapted to a large amount of setups the motivation behind it is the revalorization of industrial waste for the design of W2R routes that facilitate the implementation of Circular Economy through Industrial Symbiosis in Eco-Industrial Parks.

Although is challenging, to quantify the emissions that could be avoided on the application of the tool in an industrial setting, it is expected that the tool could be capable of finding uses for the given wastes, generating in that way a reduction of the emissions as illustrated through the valorization of CO₂ into methanol case study.

10 Ethical Implications

The thesis is focused on developing a computational methodology to find upcycling pathways for transforming waste into valuable products, in conformity with Circular Economy and Industrial Symbiosis (IS) paradigms. The supporting methodology relies on material balances subject to physics and chemistry laws and, in themselves, ethically neutral.

However, ethical issues of this work reveals itself when one considers the possible applications and consequences of the tool. Like any general-purpose technology, particularly one involving automation and optimization, there is always the risk of dual-use. For instance, although the primary goal of the tool is to facilitate sustainable development by upcycling waste flows, in theory, it could be used to discover synthetic pathways to controversial or poisonous materials, e.g., narcotics or ecotoxic chemicals, if used illicitly or with inappropriate datasets.

The open science approach followed in this project is the result of the ethical considerations towards equitable access to scientific data. By developing the tool as an open-source technology and making its methodology publicly available, the research encourages collective advancement and responsible innovation. Openness not only increases peer review and reproducibility but also broader public engagement and collective governance, helping to build trust and accountability in using data-driven technologies for sustainable development.

No experiments with animals or human beings, or sensitive biological materials have been carried out. No personal or sensitive data have been employed or inferred. All has been carried out following the ethical guidelines and appropriate legislative arrangements in place for academic research, including the Charter of Fundamental Rights of the European Union and the European Convention on Human Rights.

Still, broader application of such tools requires ongoing ethical consideration. As the technology evolves further and potentially gets integrated into industrial decision-support systems, further safeguards, such as usage policies, data curation practices, and perhaps restrictions on sensitive outputs, must be taken into account so that the tools are not abused and align with sustainability and societal well-being objectives.

Annex A: Graphical User Interface

The Graphical User Interface (GUI) has been developed using the Flask Python library, which enables integration between the backend Python application and a frontend built with HTML, CSS, and JavaScript. The interface consists of five screens, as shown in Figure 29: screen 1 displays the graph database visualization; screen 2 allows graph reduction; screen 3 handles graph optimization; screen 4 presents the linear programming feasible region plot; and screen 5 runs the DWSIM simulation of the selected configuration.

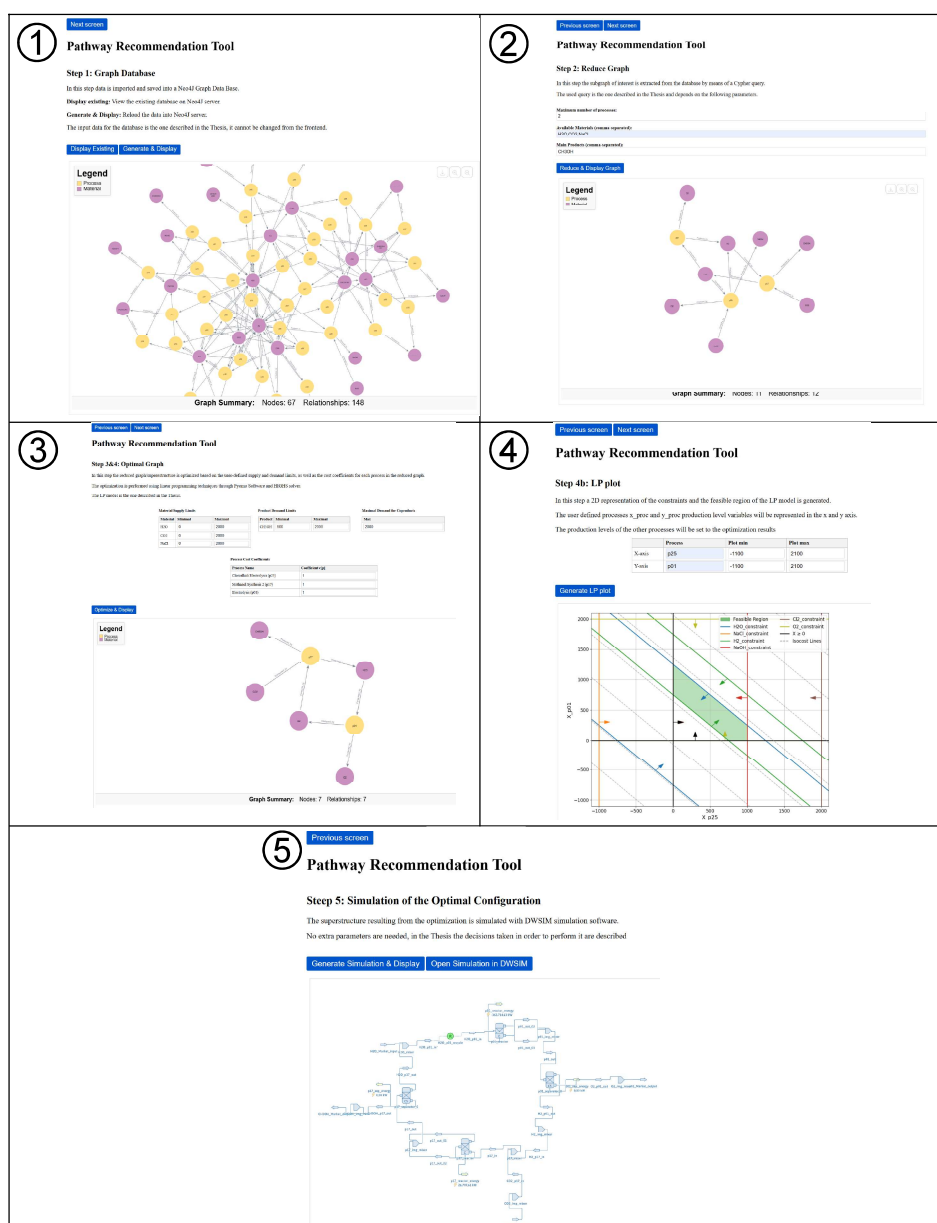


Figure 29: Graphical User Interface of the developed methodology

References

- Ackoff, Russell L (1989). “From data to wisdom”. In: *Journal of applied systems analysis* 16.1, pp. 3–9.
- Angles, Renzo and Claudio Gutierrez (2008). “Survey of graph database models”. In: *ACM Computing Surveys (CSUR)* 40.1, pp. 1–39.
<https://doi.org/10.1145/1322432.1322433>.
- AspenTech (Mar. 2025). *AspenTech has been acquired by Emerson*.
<https://www.aspentech.com/en/acquisition/emerson> (visited on 05/07/2025).
- Banothu, Narsimha, ShankarNayak Bhukya, and K Venkatesh Sharma (2016). “Big-data: Acid versus base for database transactions”. In: *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pp. 3704–3709.
<https://doi.org/10.1109/ICEEOT.2016.7755401>.
- Belaud, J.-P., B. Braunschweig, and M. White (2002). “Chapter 4.4 - The CAPE-OPEN Standard: Motivations, Development Process, Technical Architecture & Examples”. In: *Software Architectures and Tools for Computer Aided Process Engineering*. Ed. by Bertrand Braunschweig and Rafiqul Gani. Vol. 11. Computer Aided Chemical Engineering. Elsevier, pp. 303–332.
[https://doi.org/https://doi.org/10.1016/S1570-7946\(02\)80016-5](https://doi.org/https://doi.org/10.1016/S1570-7946(02)80016-5).
<https://www.sciencedirect.com/science/article/pii/S1570794602800165>.
- Berners-Lee, Tim (2010). *5-star Open Data*.
<http://5stardata.info/en/> (visited on 04/16/2025).
- Biegler, Lorenz T. (Jan. 2010). “1. Introduction to Process Optimization”. In: *Nonlinear Programming*. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, pp. 1–16.
 ISBN: 978-0-89871-702-0.
<https://doi.org/10.1137/1.9780898719383.ch1>.
<https://epubs.siam.org/doi/10.1137/1.9780898719383.ch1> (visited on 04/27/2025).
- Brundtland, Gro Harlem and Mansour Khalid (1987). *World Commission on Environment and Development: Our Common Future*.
 United Nations. New York: Oxford University Press.
<http://www.un-documents.net/ocf-ov.htm>.
- Cambridge English Dictionary (Apr. 2025). *Sustainability | Meaning in the Cambridge English Dictionary*. en.
<https://dictionary.cambridge.org/dictionary/english/sustainability> (visited on 04/25/2025).
- CAPE-OPEN Award (2006).
 Jasper van Baten of AmsterCHEM for the creation of the CAPE-OPEN based simulator COCO.
<https://www.colan.org/cape-open-awards/cape-open-2006-award/> (visited on 05/08/2025).
- CAPE-OPEN Award (2012).
 Daniel Wagner for the first implementation of CAPE-OPEN interfaces in a process

- modelling environment distributed as open source (DWSIM).
<https://www.colan.org/news/cape-open-2012-award-2/> (visited on 05/08/2025).
- Chamberlin, Donald D and Raymond F Boyce (1974). “SEQUEL: A structured English query language”. In: *Proceedings of the 1974 ACM SIGFIDET (now SIGMOD) workshop on Data description, access and control*, pp. 249–264.
- Circular Ecology (Aug. 2023). *The Carbon Emissions of Homeworking and Office Working*. en-US.
<https://circularecology.com/news/the-carbon-emissions-of-homeworking-and-office-working> (visited on 06/05/2025).
- Codd, E. F. (June 1970). “A relational model of data for large shared data banks”. In: *Commun. ACM* 13.6, pp. 377–387. ISSN: 0001-0782.
<https://doi.org/10.1145/362384.362685>.
<https://dl.acm.org/doi/10.1145/362384.362685> (visited on 04/16/2025).
- CoLaN (2025). *the CAPE-OPEN Laboratories Network*.
<https://www.colan.org/> (visited on 05/08/2025).
- European Commission (2020). *Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions: A European Strategy for Data*. en.
<https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52020DC0066> (visited on 04/30/2025).
- Federal Trade Commission (July 2004). *FTC Orders Aspen Technology, Inc. to Divest Assets from its 2002 Purchase of Hyprotech, Ltd.*
<https://web.archive.org/web/20080602232310/http://www.ftc.gov/opa/2004/07/aspen.shtm> (visited on 05/07/2025).
- Friedler, F. et al. (May 1992). “Combinatorial algorithms for process synthesis”. en. In: *Computers & Chemical Engineering* 16, S313–S320. ISSN: 00981354.
[https://doi.org/10.1016/S0098-1354\(09\)80037-9](https://doi.org/10.1016/S0098-1354(09)80037-9).
<https://linkinghub.elsevier.com/retrieve/pii/S0098135409800379> (visited on 03/06/2025).
- Gil Chaves, Iván Darío et al. (2016). *Process Analysis and Simulation in Chemical Engineering*. en. Cham: Springer International Publishing.
ISBN: 978-3-319-14812-0.
<https://doi.org/10.1007/978-3-319-14812-0>.
<https://link.springer.com/10.1007/978-3-319-14812-0> (visited on 05/07/2025).
- Hagberg, Aric A., Daniel A. Schult, and Pieter J. Swart (2008). “Exploring Network Structure, Dynamics, and Function using NetworkX”. In: *Proceedings of the 7th Python in Science Conference (SciPy2008)*. Ed. by Gäel Varoquaux, Travis Vaught, and Jarrod Millman. Pasadena, CA USA, pp. 11–15.
- Hariri, Reihaneh H., Erik M. Fredericks, and Kate M. Bowers (Dec. 2019). “Uncertainty in big data analytics: survey, opportunities, and challenges”. en. In: *Journal of Big Data* 6.1, p. 44. ISSN: 2196-1115.
<https://doi.org/10.1186/s40537-019-0206-3>.
<https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0206-3> (visited on 03/31/2025).

- ISO/IEC 39075 (2024). *Information technology - Database languages GQL*. Standard. International Organization for Standardization.
<https://www.iso.org/standard/76120.html>.
- ISO/IEC 9075 (2023). *Information technology - Database languages SQL*. Standard. International Organization for Standardization.
<https://www.iso.org/standard/76583.html>.
- INFORMS, Institute for Operations Research and the Management Sciences (Apr. 2025). *Optimization / Mathematical Programming*. en.
<https://www.informs.org/Explore/History-of-O.R.-Excellence/O.R.-Methodologies/Optimization-Mathematical-Programming> (visited on 04/29/2025).
- Jarke, M. et al. (Sept. 1999). “CAPE-OPEN: Experiences from a Standardization Effort in Chemical Industries”. In: *Standardisation and Innovation in Information Technology. Proceedings of the 1st International Conference on Standardisation and Innovation in Information Technology (SIIT '99)*. Ed. by K. Jakobs and R. Williams. Aachen, Germany: IEEE.
 ISBN: 0-7803-9935-8.
- Jing Han et al. (Oct. 2011). “Survey on NoSQL database”. en. In: *2011 6th International Conference on Pervasive Computing and Applications*. Port Elizabeth, South Africa: IEEE, pp. 363–366.
<https://doi.org/10.1109/ICPCA.2011.6106531>.
<http://ieeexplore.ieee.org/document/6106531/> (visited on 04/17/2025).
- Johnson, A.I. (1972). “Computer Aided Process Analysis and Design-A Modular Approach”. In: *Brit. Chem. Eng.* 7, p. 28.
- Johnson, Donald B. (1975). “Finding All the Elementary Circuits of a Directed Graph”. In: *SIAM Journal on Computing* 4.1, pp. 77–84.
<https://doi.org/10.1137/0204007>.
- Jusevičius, Vaidas and Remigijus Paulavičius (Jan. 2021). “Web-Based Tool for Algebraic Modeling and Mathematical Optimization”. en. In: *Mathematics* 9.21.
 Number: 21 Publisher: Multidisciplinary Digital Publishing Institute, p. 2751. ISSN: 2227-7390.
<https://doi.org/10.3390/math9212751>.
<https://www.mdpi.com/2227-7390/9/21/2751> (visited on 05/06/2025).
- Kato, Shota and Manabu Kano (2023). “Efficient physical model building algorithm using equations extracted from documents”. en. In: *Computer Aided Chemical Engineering*. Vol. 52. Elsevier, pp. 151–156.
 ISBN: 978-0-443-15274-0.
<https://doi.org/10.1016/B978-0-443-15274-0.50025-1>. (Visited on 03/31/2025).
- Khazaei, Hamzeh et al. (Sept. 2016). “How do I choose the right NoSQL solution? A comprehensive theoretical and experimental survey”. In: *Big Data & Information Analytics* 1.2&3, pp. 185–216. ISSN: 2380-6966.
<https://doi.org/10.3934/bdia.2016004>.
<https://www.aims sciences.org/article/doi/10.3934/bdia.2016004>.
- Kondili, E., C.C. Pantelides, and R.W.H. Sargent (Feb. 1993). “A general algorithm for short-term scheduling of batch operations—I. MILP formulation”. en. In: *Computers &*

- Chemical Engineering* 17.2, pp. 211–227. ISSN: 00981354.
[https://doi.org/10.1016/0098-1354\(93\)80015-F](https://doi.org/10.1016/0098-1354(93)80015-F).
<https://linkinghub.elsevier.com/retrieve/pii/009813549380015F> (visited on 11/21/2024).
- Lechtenberg, Fabian et al. (Aug. 2022). “Information retrieval from scientific abstract and citation databases: A query-by-documents approach based on Monte-Carlo sampling”. en. In: *Expert Systems with Applications* 199, p. 116967. ISSN: 09574174.
<https://doi.org/10.1016/j.eswa.2022.116967>.
<https://linkinghub.elsevier.com/retrieve/pii/S0957417422003931> (visited on 03/31/2025).
- McCrae, John Philip (2025). *The Linked Open Data Cloud*.
<https://lod-cloud.net/> (visited on 04/16/2025).
- Ministerio de Inclusión, Seguridad Social y Migraciones (2025). *Seguridad Social: Cotización / Recaudación de Trabajadores*.
<https://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/36537?changeLanguage=es> (visited on 06/05/2025).
- Peters, Max Stone, Klaus D. Timmerhaus, and Ronald E. West (2003). *Plant design and economics for chemical engineers*. en. 5th ed. McGraw-Hill higher education. New York: McGraw-Hill International Book.
ISBN: 007-124044-6.
- Rasmussen, Mads Holten et al. (2020). “BOT: The building topology ontology of the W3C linked building data group”. In: *Semantic Web* 12.1, pp. 143–161.
<https://doi.org/10.3233/SW-200385>.
- Richardson, Katherine et al. (2023). “Earth beyond six of nine planetary boundaries”. In: *Science advances* 9.37, eadh2458.
<https://doi.org/10.1126/sciadv.adh2458>.
<https://www.science.org/doi/10.1126/sciadv.adh2458>.
- Sargent, RWH and AW Westerberg (1964). “SPEED-UP (Simulation Programme for the Economic Evaluation and Design of Unsteady-State Processes) in chemical engineering design”. In: *Trans. IChemE* 42, pp. 190–97.
- Saxenna, Manoveg, Zakir Ali, and Vinod Kumar Singh (July 2014). “NoSQL Databases- Analysis, Techniques, and Classification”. In: *Journal of Advanced Database Management & Systems* 1.
- Seader, J.D., W.D. Seider, and A.C. Pauls (1974). *FLOWTRAN simulation: an introduction*. Monsanto Company.
- Umeda, Tomio, Akira Hirai, and Atsunobu Ichikawa (1972). “Synthesis of optimal processing system by an integrated approach”. In: *Chemical Engineering Science* 27.4, pp. 795–804. ISSN: 0009-2509.
[https://doi.org/10.1016/0009-2509\(72\)85013-9](https://doi.org/10.1016/0009-2509(72)85013-9).
<https://www.sciencedirect.com/science/article/pii/0009250972850139>.
- United Nations (2015). *Transforming Our World: The 2030 Agenda for Sustainable Development*.
Department of Economic and Social Affairs.
<https://sdgs.un.org/2030agenda> (visited on 04/24/2025).

- Universitat Politècnica de Catalunya (2024). *Manual de registre d'ingressos de recerca*. ca.
<https://rdi.upc.edu/ca/gestions/guies-i-directrius/manual-de-comptabilitzacio-dingressos-i-facturacio-1.pdf>.
- Vossen, Gottfried (2009). “ACID Properties”. In: *Encyclopedia of Database Systems*. Ed. by LING LIU and M. TAMER ÖZSU. Boston, MA: Springer US, pp. 19–21. ISBN: 978-0-387-39940-9.
https://doi.org/10.1007/978-0-387-39940-9_831.
- Wagner Oliveira de Medeiros, Daniel (2011). *DWSIM 2.0 running on Windows 7*. Image from Wikimedia Commons. Licensed under GNU General Public License v2 or later.
https://commons.wikimedia.org/wiki/File:Dwsim_20_windows.jpg (visited on 05/26/2025).
- Wilkinson, Mark D. et al. (Mar. 2016). “The FAIR Guiding Principles for scientific data management and stewardship”. en. In: *Scientific Data* 3.1. Publisher: Nature Publishing Group, p. 160018. ISSN: 2052-4463.
<https://doi.org/10.1038/sdata.2016.18>.
<https://www.nature.com/articles/sdata201618>.
- Yeomans, Hector and Ignacio E. Grossmann (June 1999). “A systematic modeling framework of superstructure optimization in process synthesis”. In: *Computers & Chemical Engineering* 23.6, pp. 709–731. ISSN: 0098-1354.
[https://doi.org/10.1016/S0098-1354\(99\)00003-4](https://doi.org/10.1016/S0098-1354(99)00003-4).
<https://www.sciencedirect.com/science/article/pii/S0098135499000034> (visited on 04/27/2025).
- Zeleny, Milan (1987). “Management support systems: Towards integrated knowledge management”. In: *Human systems management* 7, pp. 59–70.
<https://api.semanticscholar.org/CorpusID:167591343>.

List of Figures

1	Linear and Circular Economy Models	12
2	DIKW pyramid	14
3	5 V's of Big Data (Hariri, Fredericks, and Bowers 2019)	15
4	Overview 5-Star Deployment Scheme and of the Linked Open Data	16
5	Databases data model schema	18
6	Positioning of Data Stores by Size and Complexity (Khazaei et al. 2016)	19
7	Comparison of process network and corresponding STN representations (Kondili, Pantelides, and R.W.H. Sargent 1993)	22
8	Optimization topology (Biegler 2010)	23
9	General mathematical programming formulation	23
10	Process simulation in DWSIM (Wagner Oliveira de Medeiros 2011)	25
11	Methodological steps of the procedure	27
12	Graph Data Model	28
13	Graph reduction step	29
14	Cypher query for the graph reduction	30
15	Superstructure formalization step	32
16	Superstructure resolution step	33
17	Representation of a process in DWSIM	34
18	Representation of a material connections in DWSIM	35
19	Chemical Process Network construction	35
20	Graph database	40
21	Reduced graph for the base case	41
22	Optimal configurations for the base case	43
23	Feasible Region and Isocost Lines for the base case	43
24	Flowsheet for the Optimal Configurations for the base case	44
25	Reduced graph for the alternative case	45
26	Optimal configurations for the alternative case	46
27	Feasible Region and Isocost Lines for the alternative case	47
28	Flowsheet for the Optimal Configurations for the base case	47
29	Graphical User Interface of the developed methodology	53

List of Tables

1	Taxonomy of NoSQL Databases	18
2	Overview of Graph Database Systems	20
3	Overview of AML Systems	24
4	Overview of Chemical Process Simulation Software	26
5	Processes Included in the System	36
6	Materials Involved in the Process Network	39
7	Human Resources for the Master Thesis Development	50
8	Total Cost of the Project Development	50