UNIVERSIDAD PANAMERICANA
CAMPUS GUADALAJARA

POSGRADOS DE INGENIERÍA

A multi-objective sustainable closed-loop supply chain network problem solved by an exact method and heuristic algorithms

A dissertation submitted by:
Joel Novi Rodríguez Escoto

For the degree of
Doctor in Engineering

Advisor
Dr. Elías Olivares Benítez

Co-Advisor
Dr. Samuel Moisés Nucamendi Guillén

Zapopan, Jalisco, June 2026

The thesis of Joel Novi Rodríguez Escoto was reviewed and approved by the following:

Dr. Elías Olivares Benítez
Professor of the Faculty of Engineering
Universidad Panamericana Campus Guadalajara
Thesis Advisor, Chair of Committee

Dr. Samuel Moisés Nucamendi Guillén
Professor & Head of Operations Management department
Universidad Panamericana Campus Guadalajara
Thesis Co-Advisor

Dr. Julie Drzymalski
Professor & Director of Industrial & Systems Engineering Program
College of Engineering Temple University

Dr. Alfredo Segismundo Ramos Jiménez
Professor & Associate Dean for academic affairs
Universidad Panamericana Campus Guadalajara

Dr. Iris Abril Martínez Salazar
Professor of the Faculty of Mechanical and Electrical Engineering
Universidad Autónoma de Nuevo León

Abstract

In this thesis, we present a multi-objective model for a sustainable closed-loop supply chain network featuring hybrid facilities. The model incorporates circularity principles, sustainability factors, vehicle emissions, and the obnoxiousness of hybrid facilities across five forward echelons, two reverse echelons, and three location decisions. This structure leads to exponential growth in complexity across decision levels.

The literature review reveals a key gap: classical exact and matheuristic algorithms remain underexplored for multi-objective sustainable closed-loop supply chain problems. In this context, we pioneer the application of the well-known methods to this problem, an improved augmented ϵ-constraint method, a kernel search matheuristic, and a hybrid non-dominated sorting genetic algorithm II. Using a synthetically generated dataset based on supply chain parameters, we evaluate these methods with standard Pareto front quality metrics.

Sensitivity analysis reveals the model's behavior under varying conditions. A comparison of the solution methods demonstrates their performance. Crucially, the kernel search matheuristic outperforms both the classical exact method (72% faster) and the metaheuristic (35% faster) in CPU time, delivering high-quality solutions in the shortest time.

Table of Contents

List of Figures

List of Tables

List of Algorithms

Acknowledgement

I'm profoundly grateful to God, my beloved wife, my fearless daughter, my family, my neighbors, this house of study, and my advisors for the opportunity and support to complete the doctorate.

Chapter 1

Introduction

1.1 Research description

The sustainable closed-loop supply chain network problem has attracted increasing attention in supply chain management research. Designing such a network involves strategic decisions, such as location selection and supplier choice, tactical planning, including inventory management and routing, and operational activities, such as scheduling (Olivares-Benitez et al., 2013).

This design must carefully balance the interests of multiple stakeholders by effectively and sustainably addressing economic feasibility, environmental impact, and social responsibility. Moreover, integrating forward and reverse logistics enhances efficiency and supports sustainability objectives (Barbosa-Póvoa et al., 2018), making the problem more representative of real-world conditions. However, this integration significantly increases the problem's complexity. To effectively manage this complexity, innovative methods and approaches are essential for delivering faster, more effective solutions.

This challenge presents an excellent opportunity to develop strategies for navigating the increasing complexity of dynamic market pressures and enabling companies to maintain a competitive edge while ensuring sustainability. Ultimately, exploring practical applications is crucial for finding more effective solutions to the multi-objective sustainable closed-loop supply chain network problem (Govindan et al., 2015; Tavana et al., 2022).

1.2 Applications and motivation

Research on sustainable supply chains—particularly closed-loop supply chains (CLSCs)—encompasses a broad range of real-world applications. These applications are especially prominent in industries that involve reverse logistics, remanufacturing, and recycling, such as the tire industry, household appliances and electronics, pharmaceuticals, garments and textiles, beverages (e.g., returnable containers), wood-plastic composites, and automotive sectors. These industries are increasingly adopting sustainability practices to enhance competitiveness while meeting environmental regulations, addressing resource scarcity, and responding to consumer expectations (Momeni et al., 2024; Zengin et al., 2024).

Sustainability in supply chains encompasses three key pillars: economic, environmental, and social aspects (often referred to as the triple bottom line). In multi-objective CLSC models, the economic dimension focuses on minimizing total costs (e.g., procurement, production, transportation, and facility setup). The environmental dimension emphasizes reducing negative impacts, such as lowering CO_2 emissions from transportation and operations, minimizing energy use, and promoting resource recovery.

The social dimension addresses community well-being, for example, by mitigating exposure to obnoxious facilities (e.g., noisy, odorous, or polluting recovery/recycling centers) through careful site selection to reduce health and quality-of-life risks. By integrating these conflicting objectives, companies in the listed industries can achieve more balanced, resilient, and circular networks that support long-term profitability alongside genuine sustainability gains (Joshi, 2022; Tavana et al., 2022).

The obnoxiousness characteristic is relevant to consider when locating potential hybrid facilities (defined as hybrid by adding a function, such as collection, recycling, or remanufacturing) that can be points of sale, distribution centers, or factories. These facilities will allow customers to recycle their used products. The facility will be open based on the customer's proximity to it and the potential facility's obnoxiousness; that is, to be close to the customer for convenience, but also far due to their pollution (Church and Drezner, 2022).

Also, the use of electric vehicles and internal combustion in the supply chain to reduce CO_2 emissions is relevant, so that the model can assign the best combination of usage. The latter are added to a multi-objective model based on the single-objective version (Pazhani et al., 2021), which involves economic, environmental, and social factors. Therefore, this proposal aims to advance Sustainable Development Goals 12 (responsible consumption and production) and 13 (climate action) through circular-economy practices and emission-reduction strategies, respectively (United Nations, 2026).

This results in a multi-objective, sustainable, closed-loop supply chain with a hybrid facilities problem, represented as a mixed-integer linear program and a generalization of a combinatorial optimization problem (encompassing both discrete and continuous solution spaces). Being a combinatorial problem, it is considered NP-hard, which manifests in the difficulty of obtaining optimal solutions in instances arising from real-world problems within competitive computational times (Lenstra and Kan, 1981).

For complexity reasons, metaheuristics have emerged to overcome the difficulty to obtain solutions in competitive computational time, which generally fall into two categories: those that start with an initial solution (local search, tabu search, simulated annealing; those are also known as trajectory) and those that begin with an initial population of solutions, such as genetic algorithms, which are the most well-known group (Coello, 2025; Loaiza et al., 2017; Talbi et al., 2012). These algorithms aim to produce diverse solutions quickly.

Recently, however, matheuristics have evolved, combining the processing speed of heuristics with the quality of exact methods. In other words, they are approximation algorithms that result from combining heuristics and mathematical programming techniques. One of the main characteristics of matheuristics is the variability in algorithm implementation, which depends on the problem to be solved (Maniezzo et al., 2021).

Practical decision-making frequently demands rapid evaluation of numerous parameter configurations, sensitivity analyses, or adaptive re-optimization in response to changing conditions (e.g., fluctuating demand, cost variations, or input uncertainty). Although exact methods guarantee optimality, their often prohibitive computation times make them impractical for iterative or time-sensitive applications. This drawback highlights the importance of effective matheuristics, which combine mathematical programming with heuristic principles to produce high-quality (near-optimal) solutions in practically acceptable time frames. Such approaches enable fast what-if analyses, interactive decision support, and timely deployment in operational settings (Govindan et al., 2015; Jebreili et al., 2025).

In this context, this research proposes new solution methods for the multi-objective sustainable closed-loop supply chain network with hybrid facilities, using an exact method, a matheuristic, and a metaheuristic to overcome complexity and deliver practical solutions within competitive computational time.

1.3 Contributions

The contributions of this work are:

- A mathematical model that integrates hybrid facilities, considering their obnoxiousness and CO_2 emissions, into a multi-objective sustainable closed-loop supply chain network.
- An analysis of the model behavior of a multi-objective sustainable closed-loop supply chain network with hybrid facilities through an ANOVA sensitivity analysis.
- An exact method application to a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- A matheuristic approach that combines the strengths of both heuristics and exact methods to efficiently solve the complex problem of designing a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- A metaheuristic approach that achieves diversity to solve the complex problem of designing a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- A comparison analysis of different solution methods to evaluate their effectiveness in solving multi-objective sustainable closed-loop supply chain problems.

1.4 Research objectives

The general objective of the research is to develop and implement exact, matheuristic, and metaheuristic methods for optimizing a multi-objective sustainable closed-loop supply chain network with hybrid facilities.

The specific objectives are presented below:

- To review relevant literature on solution methods to solve a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- To develop a mathematical model for the design of a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- To develop and implement an exact method for optimizing a multi-objective, sustainable closed-loop supply chain network design with hybrid facilities.
- To develop and implement a matheuristic for optimizing a multi-objective sustainable closed-loop supply chain network design with hybrid facilities.
- To develop and implement a metaheuristic for optimizing a multi-objective sustainable closed-loop supply chain network design with hybrid facilities.
- To perform a comparative analysis of quality and computational time of the solution methods in solving multi-objective sustainable closed-loop supply chain problems.

1.5 Research questions

The questions that this research aims to answer are:

- Can an exact method obtain optimal solutions for a multi-objective sustainable closed-loop supply chain network with hybrid facilities in a reasonable time?
- Can a matheuristic obtain high-quality solutions for a multi-objective sustainable closed-loop supply chain network with hybrid facilities in a reasonable time?
- Can a metaheuristic obtain high-quality solutions for a multi-objective sustainable closed-loop supply chain network with hybrid facilities in a reasonable time?
- Can one of the proposed approaches consistently perform the others differently in terms of quality of the solutions and computational time?

1.6 Hypotheses

The hypotheses of the proposed work are as follows:

- Null H1: The proposed exact method cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Alternative H1: The proposed exact method can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Null H2: The proposed exact method cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.
- Alternative H2: The proposed exact method can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.
- Null H3: The proposed matheuristic cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Alternative H3: The proposed matheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Null H4: The proposed matheuristic cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.
- Alternative H4: The proposed matheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.
- Null H5: The proposed metaheuristic cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Alternative H5: The proposed metaheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.

- Null H6: The proposed metaheuristic cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.
- Alternative H6: The proposed metaheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.
- Null H7: There is no significant difference in solution quality and computational efficiency among the exact method, matheuristic, and metaheuristic for the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Alternative H7: There is a significant difference in solution quality and computational efficiency among the exact method, matheuristic, and metaheuristic for the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.

1.7 Thesis structure

The remainder of the document structure consists of Chapter 2, where a literature review is presented, denoting the contribution of this research. Chapter 3 describes the problem definition and the mathematical model, as well as the solution methods: the exact method, the matheuristic, and the metaheuristic algorithms. It also outlines the dataset characteristics and performance metrics used to evaluate these methods. Chapter 4 discusses the results obtained with the exact, metaheuristic, and matheuristic approaches. Finally, Chapter 5 summarizes the conclusions and the insights for future research. Additionally, the extensions of tables, graphics, and evidence of the thesis are in appendices A and B.

Bibliography

Barbosa-Póvoa, A. P., da Silva, C., and Carvalho, A. (2018). Opportunities and challenges in sustainable supply chain: An operations research perspective. *European Journal of Operational Research*, 268:399–431.

Church, R. L. and Drezner, Z. (2022). Review of obnoxious facilities location problems. *Computers & Operations Research*, 138.

Coello, C. A. C. (2025). *Multiobjective Optimization*, pages 231–257. Springer Nature Switzerland, Cham.

Govindan, K., Soleimani, H., and Kannan, D. (2015). Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. *European Journal of Operational Research*, 240:603–626.

Jebreili, S., Babazadeh, R., Fazayeli, S., A. Kamran, M., and Gharibi, A. R. (2025). A multi-objective milp model for sustainable closed-loop supply chain network design: Evidence from the wood–plastic composite industry. *Mathematics*, 13(21).

Joshi, S. (2022). A review on sustainable supply chain network design: Dimensions, paradigms, concepts, framework and future directions. *Sustainable Operations and Computers*, 3:136–148.

Lenstra, J. K. and Kan, A. H. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11:221–227.

Loaiza, R. E. P., Olivares-Benitez, E., Gonzalez, P. A. M., Campanur, A. G., and Flores, J. L. M. (2017). Supply chain network design with efficiency, location, and inventory policy using a multiobjective evolutionary algorithm. *International Transactions in Operational Research*, 24:251–275.

Maniezzo, V., Boschetti, M. A., and Stützle, T. (2021). *Matheuristics*. Springer International Publishing.

Momeni, M. A., Jain, V., and Bagheri, M. (2024). A multi-objective model for designing a sustainable closed-loop supply chain logistics network. *Logistics 2024, Vol. 8, Page 29*, 8:29.

Olivares-Benitez, E., Ríos-Mercado, R. Z., and González-Velarde, J. L. (2013). A metaheuristic algorithm to solve the selection of transportation channels in supply chain design. *International Journal of Production Economics*, 145:161–172.

Pazhani, S., Mendoza, A., Nambirajan, R., Narendran, T. T., Ganesh, K., and Olivares-Benitez, E. (2021). Multi-period multi-product closed loop supply chain network design: A relaxation approach. *Computers & Industrial Engineering*, 155:107191.

Talbi, E.-G., Basseur, M., Nebro, A. J., and Alba, E. (2012). Multi-objective optimization using metaheuristics: non-standard algorithms. *International Transactions in Operational Research*, 19(1-2):283–305.

Tavana, M., Kian, H., Nasr, A. K., Govindan, K., and Mina, H. (2022). A comprehensive framework for sustainable closed-loop supply chain network design. *Journal of Cleaner Production*, 332:129777.

United Nations (2026). Goal 12: Ensure sustainable consumption and production patterns. https://sdgs.un.org/goals/goal12. Online; accessed february 3rd 2026.

Zengin, M., Amin, S. H., and Zhang, G. (2024). Closing the gap: A comprehensive review of the literature on closed-loop supply chains. *Logistics*, 8(2):54.

Chapter 2

Literature review

To establish the theoretical foundation for the contributions of this thesis, this chapter proceeds as follows. First, it provides a formal definition of a multi-objective sustainable closed-loop supply chain network. Second, it reviews and critically discusses the solution methods proposed in the recent literature. Third, it introduces the kernel search (KS) matheuristic. Finally, it presents an explanation of the Non-dominated Sorting Genetic Algorithm II (NSGAII) metaheuristic.

2.1 Sustainable closed-loop supply chain network problem

The closed-loop supply chain network problem (CLSCN) occurs when the reverse product flow is integrated into the leading supply chain, closing the loop rather than discarding material (Asif et al., 2012). The independent operations of forward and reverse supply in a traditional supply chain result in inefficiency within the system; hence, the best way to optimize a supply chain is to close the loop (Pazhani and Ravindran, 2018). The first work on a CLSCN problem was carried out by Fleischmann et al. (2001), which introduced the problem and proposed a recovery network that analyzed recovery product costs using a general facility location model. This foundational study has inspired numerous subsequent research efforts.

There are different ways to tackle reverse supply based on the type of product returned. According to Ferguson and Souza (2010), three types of returns could be considered: (i) consumer returns: the product is usually new; (ii) end-of-use returns: the product is used but still functional; and (iii) end-of-life returns: the product reached its useful life. This implies various dispositions of the returns, including landfilling, incineration, recycling, part harvesting, reselling, internal reuse, remanufacturing, or refurbishing.

The CLSCN problem is closely related to the circular economy concept, which encompasses practices such as reduction, reuse, recycling, and recovery. It is also called reverse logistics, effectively closing the loop in traditional supply chains. Unlike conventional supply chains that operate independently and inefficiently, the circular economy promotes sustainable solutions and economic savings for businesses (Sarkis and Dou, 2017). Integrating sustainability into the CLSCN introduces three key dimensions: environmental (e.g., pollution and recycling), economic (e.g., profit), and social (e.g., working conditions, job creation, and human rights).

The application and modeling of such SCLSCNs can be implemented within two- or three-dimensional frameworks. In the two-dimensional approach, the model usually optimizes two of the three sustainability pillars (most commonly economic and environmental objectives). In contrast, the three-dimensional framework simultaneously considers all three pillars—economic viability, environmental performance, and social responsibility. This multidimensional perspective allows decision-makers to capture the inherent trade-offs among conflicting objectives and to design more balanced, truly sustainable networks (Joshi, 2022).

Since then, the Multi-Objective Sustainable Closed-Loop Supply Chain Network (MOSCLSCN) problem has involved the interplay among forward and reverse logistics decisions, sustainability considerations, and objective interactions. In addition to general supply chain decisions, specific choices are made at three levels of organizational planning. Strategic decisions are long-term and capital-intensive, focusing on facility location, capacity, and inventory management. Tactical, medium-term decisions involve allocation, assignment, material flow, and product transportation. Operational-level decisions pertain to logistics scheduling. These decisions and objectives significantly increase the complexity of the models involved (Jayarathna et al., 2021).

2.1.1 Multi-objective sustainable closed-loop supply chain network problem

Multi-objective optimization accounts for interactions among these decisions, even when conflicts arise. Many studies emphasize the significance of adopting multi-objective models, which is why most practical applications of the SCLSCN problem utilize a multi-objective approach rather than a single-objective one. Given their complexity, it is crucial to identify robust and effective solution methods for decision analysis (Amin et al., 2020; Govindan et al., 2015; Joshi, 2022). Numerous theories, mechanisms, and concepts have been developed to tackle the MOSCLSCN problem.

Table 2.1 summarizes the recent literature on multi-objective deterministic sustainable closed-loop supply chain network (SCLSCN) design problems. The table includes the studies reviewed and the solution approaches proposed in the present thesis. Unless otherwise noted, all models consider exactly three objectives. The table reports, for each study, the forward and reverse supply chain echelons considered, the solution method employed, the location decisions involved, and the specific algorithm(s) applied. Consistent with the reviewed literature, the economic objective is formulated as the minimization of the total cost. Environmental performance is measured through emissions and pollution-related indicators, while the social dimension captures aspects such as job creation, accident rates, and broader corporate social responsibilities.

The complexity of the studies presented in Table 2.1 is analyzed by examining the number and type of facility location decisions, the objectives pursued, and the forward and reverse echelons considered in each work. Since facility location problems are NP-hard (Cornuéjols et al., 1983; Krarup and Pruzan, 1983; Soleimani and Kannan, 2015), the inclusion of multiple facility layers significantly increases problem complexity. This discussion provides supporting evidence for the relative difficulty of the models reviewed.

Hajiaghaei-Keshteli and Fard (2019) presented a nonlinear problem of the MOSCLSCN with three objective functions: total cost, environmental impacts, and job creations, including transportation discount based on shipment volume, which considered nine types of facilities to be established: manufacturing and

Table 2.1: Summary of recent deterministic MOSCLSCN applications

References	Echelons FWD	Echelons REV	Method	FLP	Algorithms
Hajiaghaei-Keshteli and Fard (2019)	7	3	META	7	Hybridization GA
Mehrjerdi and Shafiee (2020)	3	3	EXME	4	ϵC
Yun et al. (2020)	5	5	META	5	GA and hybrid GA
Khorshidvand et al. (2021)	5	7	MATH	6	LR-based WS
Akbari-Kasgari et al. (2022)	3	3	EXME	3	WS and ϵC
Irawan et al. (2022)	2	4	MATH	2	Hybrid CP and META
Mogale et al. (2022)	4	5	META	4	ϵC & Hybrid NSGAII
Seydanlou et al. (2022)	6	7	META	4	VCS-SA and EMA-GA
Soleimani et al. (2022)	4	5	MATH	1	LR-heuristics
Tirkolaee et al. (2022)	4	4	META	6	MOGWO and NSGAII
Abbasi et al. (2023)	3	4	EXME	3	Weighted Tchebysheff
Becerra et al. (2024)	4	4	EXME	1	LM
Gholipour et al. (2023)	6	7	META	3	NSGAII and MOPSO
Goodarzian et al. (2023)	4	5	META	2	SPEA-II and PESA-II
Mirzaei et al. (2023)	7	5	META	2	MOSA, MOPSO, MOGWO,and MOWOA
Yamchi et al. (2023)	6	6	META	2	NSGAII, NRGA, and NSGAIII
Momeni et al. (2024)	4	7	EXME	4	ϵC
Rodríguez-Escoto et al. (2025) *	5	2	EXME	3	LPR-based AUG2
Jebreili et al. (2025)	3	6	EXME	4	Lp-metric
Restrepo Diaz and Amin (2025)	3	6	EXME	5	WS, ϵC , and Hybrid WS/ϵC
Jauhari and Wakhid (2026)	4	4	EXME	4	Transformation function
Rodríguez-Escoto et al. (2026) **	5	2	MATH	3	KS-based AUG2
Solution method 3 ***	5	2	META	3	Hybrid NSGAII

Nomenclature: AUG2 = Augmented ϵ-constraint 2, CP = Compromise programming, EMA = Electromagnetism-like algorithm, EXME = Exact method, GA = Genetic algorithm, GP = Goal programming, KS = Kernel search matheuristic, LM = Lexicographic method, LPR = Linear Programming Relaxation, LR = Lagrangian relaxation, LS = Local search, MATH = Matheuristic, META = Metaheuristic, MOGWO = Multi-objective Grey Wolf Optimizer, MOPSO = Multi-objective particle swarm optimization, MOSA = Multi-objective Simulated Annealing, MOWOA = Multi-objective whale optimization algorithm, NRGA = Non-dominated ranked genetic algorithm, NSGAII = Non-dominated sorting genetic algorithm II, PESA-II = Pareto Envelope-based Selection Algorithm II, SA = Simulated annealing, SEO = Social Engineering Optimizer, SPEA-II = Strength Pareto Evolutionary Algorithm II, VCS = Virus colony search algorithm, WS = Weighted-sums, ϵ-C = ϵ-constraint. * = The proposed exact method, ** = The proposed matheuristic, and *** = The proposed metaheuristic.

distribution centers, inspection centers for recovering, remanufacturing, recycling, and disposal centers. They proposed several hybridizations of Genetic Algorithms (GA), particularly the Keshtel and Genetic Algorithm (KAGA). The latter yielded better-quality results than other hybrid algorithms on Pareto-optimal solutions, though at the expense of longer computation times.

Mehrjerdi and Shafiee (2020) addressed an MOSCLSCN model with three objective functions: reducing total cost, energy consumption, and pollution, increasing job opportunities and multiple sourcing, and establishing four types of facilities: manufacturing, distribution, collection, and recycling centers. They approached the problem using the ϵ-constraint method, solving a case study from the tire industry, and demonstrated that multiple sourcing improves robustness.

Yun et al. (2020) worked on the SCLSCN model, which involved five decisions: facility locations, manufacturer, distribution, retailer, collection, and recovery facilities, considering different distribution channels and applying a genetic algorithm (GA) and a hybrid of cuckoo search (CS) and GA. Their approach outperformed traditional approaches in terms of solutions obtained, but fell short in execution time.

Khorshidvand et al. (2021) implemented the Lagrangian relaxation based on the weighted-sum method in an MOSCLSCN model in which six facilities are established: manufacturer, distributor, retailer, collection, recycling, and recovery centers. Their model maximizes total profit and employee safety while simultaneously minimizing CO_2 emissions. The matheuristic achieved a 50% improvement against the GUROBI solver in elapsed CPU time.

Akbari-Kasgari et al. (2022) developed a MOSCLSCN application with three objective functions: maximize profit and social desirability, and minimize water consumption for the copper distribution industry with facility decisions as factory, distribution, and collection centers, using exact methods such as ϵ-constraint and weighted sum. Their contribution is validated with real industry data, offering practical insights for sustainable and resilient copper supply chain design.

Irawan et al. (2022) developed a bi-objective SCLSCN involving economic and environmental objectives that establishes two types of facilities: distribution and recycling centers, utilizing matheuristic algorithms to enhance solution quality and computational efficiency compared to exact methods.

Mogale et al. (2022) designed a bi-objective MOSCLCSN model with total cost and carbon emissions minimization that establishes four types of facilities: production, distribution, retail, and collection centers, applying the ϵ-constraint and two stages of hybridization of Non-dominated sorting genetic algorithm II (NSGAII). The approach yields robust and computationally efficient results compared to ϵ-constraint methods, with a gap of less than 2%.

Seydanlou et al. (2022) presented an MOSCLSCN model for the olive agriculture industry, considering four facility location decisions regarding the construction of distribution, composting, and manufacturing centers. They proposed a model based on the triple bottom line, minimizing total cost and CO_2 emissions, and maximizing job opportunities. They used both metaheuristics, GA and Simulated Annealing (SA), and the hybridization of the electromagnetism-like algorithm (EMA)-GA and the virus colony search algorithm (VCS)-SA, achieving better performance with the hybrid algorithms.

Soleimani et al. (2022) presented an MOSCLSCN model applying Lagrangian relaxation and heuristics that involve a decision on the location of the distributor center. The authors maximize total profit and the number of job opportunities, and minimize energy consumption. They found that the developed matheuristic algorithm performs efficiently, with a computational time improvement of at least 66% compared with the CPLEX solver.

Tirkolaee et al. (2022) designed an MOSCLSCN model that minimizes the total cost, total pollution, and total human risk, which establishes six types of potential facilities: a factory, distribution center, collection center, quarantine center, recycling center, and disposal center, in the context of COVID-19 and face masks. They used MOGWO and NSGAII and found better performance with MOGWO than with NSGAII.

Abbasi et al. (2023) designed an MOSCLSCN model to determine the location of the manufacturing, collection, and recycling centers in a COVID-19 context, minimizing the total cost, the negative environmental impact, and the negative social impact. The weighted Tchebyshev method was used to generate cost scenarios that account for the pandemic.

Becerra et al. (2024) proposed three objective functions: minimize the economic and carbon emissions, and maximize social impact in a nonlinear MOSCLSCN model with the production plant's location, inventory, and routing decisions. To solve the problem, they applied linearization to their model, utilizing the lexicographic method. Their work primarily involves modeling behavior in the context of economic and environmental trade-offs.

Gholipour et al. (2023) developed an MOSCLSCN model with three objective functions: minimizing the supply chain's costs and risk, and maximizing profit for pomegranate, with three location facilities to be established: Distribution, factory, and recycling sites. They utilized NSGAII and MOPSO. The NSGAII achieved better results in total and supply risk reduction than MOPSO.

Goodarzian et al. (2023) applied two metaheuristics, the strength Pareto evolutionary algorithm II (SPEA-II) and the Pareto envelope-based selection algorithm (PESA-II), in a case study of the citrus industry, using an MOSCLSCN, locating two facilities: distribution and recycling centers. They minimized three objectives: total cost, environmental impact, and social responsibility, as measured by pollution.

Mirzaei et al. (2023) proposed a dual-channel MOSCLSCN with objective functions as minimizing total costs and pollutants/emissions, and maximizing created jobs for the tea industry, with two location decisions: processing and recycling centers. They applied an LP-metric method to small instances and four multi-objective metaheuristics to large instances: multi-objective simulated annealing (MOSA), multi-objective particle swarm optimization (MOPSO), MOGWO, and the whale optimization algorithm (MOWOA). The MOSA algorithm outperformed the others in terms of CPU time.

Yamchi et al. (2023) worked on three objectives where it minimizes cost and CO_2 emissions, and it maximizes demand satisfaction and justice-based jobs on the MOSCLSCN model in agricultural products that established distribution and vermicomposting centers, applying metaheuristics where non-dominated ranked genetic algorithm (NRGA) outperformed NSGAII and NSGAIII in generating high-quality Pareto-optimal solutions.

Momeni et al. (2024) implemented a MOSCLSCN establishing four types of facilities: distribution, remanufacturing, recycling, and recycling centers with four objectives that minimize economic costs associated with the supply chain, the negative environmental aspects (such as CO_2 emissions, water usage, and energy consumption), and number of facilities, while maximizing positive social impacts (such as job creation and self-sufficiency). Using an ϵ-constraint method, they conducted a sensitivity analysis that accounted for demand variations.

Jebreili et al. (2025) designed a MOSCLSCN in the wood-plastic composite industry, applied to a real case in Iran. The model minimizes overall cost, maximizes employment rate, maximizes green impact, and establishes manufacturing, distribution, collection, inspection, reuse, and re-production centers. The multi-objective MILP is solved using the Lp-metric method to aggregate objectives into a single scalar for optimization; model verification confirms its real-world applicability.

Restrepo Diaz and Amin (2025) presented a MOSCLSCN model focused on the sustainable management of computer e-waste, establishing plant, retailer, collection, supplier, and disassembly centers. Three objectives: maximizing profits and social innovations while minimizing gas emissions. Three methods are used to generate Pareto-optimal solutions: weighted-sum, ϵ-constraint (providing the most efficient solutions for trade-offs), and a hybrid approach.

Jauhari and Wakhid (2026) developed a bi-objective SCLSCN model that minimizes total operational cost and maximizes job opportunities across four facility types: processing, sorting, distribution, and recycling centers. A transformation function serves as a normalization scheme for the solution approach.

The works cited in this section consider three to seven forward echelons, two to seven reverse echelons, and one to seven location decisions. This work considers seven echelons and three location decisions in the chain. In addition, ten of them were solved by metaheuristics, nine by exact methods, and three by matheuristics. At a glance, this fact highlights the complexity of the problem to be solved. It underscores the need to explore new approaches, methods, and solutions for the MOSCLSCN problem.

This work defines a model for the MOSCLSCN problem that establishes three location decisions: manufacturers, warehouses, and warehouse/recycling centers. Three objectives are minimized: total cost, CO_2 emissions, and the population vulnerable to obnoxious facilities. This objective, which has not been studied before, is inspired by the known minimum-impact location problem cited by Church and Drezner (2022). This impact could be considered obnoxious due to pollution (e.g., air pollution, noise, thermal pollution, and soil pollution). Additionally, a methodological approach is proposed that can simultaneously analyze economic, social, and environmental costs. This approach enables decision-makers to understand the holistic impact of their operations, allowing them to make informed decisions that balance economic efficiency with social and environmental responsibility. The proposed model has not been previously presented as a multi-objective problem.

2.2 Matheuristic algorithm

A key research opportunity remains in the hybridization of exact and metaheuristic methods. To date, only a few studies have applied hybrid approaches or matheuristics to the MOSCLSCN (Govindan et al., 2015; Jayarathna et al., 2021). Matheuristic algorithms emerged by combining the faster heuristic processing with the quality of the exact method, and they result from integrating heuristics with mathematical computation (Boschetti et al., 2009). Additionally, matheuristics exhibit implementation variability, depending on the specific problem being solved. Some examples of original matheuristic algorithms are relaxation methods, local branching methods, Lagrangian methods, decomposition methods, corridor methods, kernel search, and fore-and-back (Maniezzo et al., 2021). However, matheuristic algorithms have not been widely explored or tested in the multi-objective setting; according to work (Jayarathna et al., 2021), which presented a multi-objective optimization in sustainable supply chain and logistics review, only 1% of the reviewed work involves matheuristic applications, which supports the effort to delve into this type of algorithm for multi-objective optimization.

2.2.1 Kernel search matheuristic

The kernel search matheuristic was selected as one of the methods to solve the MOSCLSCN problem addressed in this research. Kernel search matheuristic (Angelelli et al., 2007; Maniezzo et al., 2021) are ideal for problems involving binary variables, as demonstrated in recent implementations by Archetti et al. (2021); Borčinová (2022); Zhang et al. (2024). A kernel search algorithm divides the solution space into a set of "promising" variables, known as the kernel, and other sets of variables, called buckets. The algorithm iteratively updates the kernel with variables from the buckets to diversify the search until a stopping criterion is met.

Zhang et al. (2022) presents a work similar to ours, in the context of perishable food, where a bi-objective closed-loop supply chain is modeled. A kernel search-based ϵ-constraint method was implemented as a solution, yielding at least a 50% improvement against the CPLEX solver in CPU time. In addition to its primary applications, the kernel search matheuristic has been successfully applied across various fields. For instance, it has been used to solve a fair facility location problem using a bi-objective model, resulting in improved solutions within the time limit and reduced computational time compared to CPLEX (Filippi et al., 2021). Similarly, in a multi-vehicle inventory routing problem, this approach achieved superior solution quality compared to other algorithms (Archetti et al., 2021). Furthermore, it demonstrated a remarkable 50% reduction in average computational time for a capacitated vehicle routing problem (Borčinová, 2022). Lastly, in a closed-loop inventory routing problem, the matheuristic achieved computational times that were 1.6% of those of the CPLEX solver, on average (Zhang et al., 2024). To our knowledge, no work has been presented on solving MOSCLSCN using the kernel search matheuristic.

2.3 Metaheuristic algorithm

The metaheuristic algorithm is a guided, approximate, and randomized exploration of the solution space, with local and global searches as its two categories. Genetic algorithms are the preferred solution method among evolutionary algorithms, which are heuristic search and optimization algorithms that emulate the principles of evolution through natural selection and genetics. According to the work (Jayarathna et al., 2021), which presented a multi-objective optimization in sustainable supply chain and logistics review, 33% of the reviewed work involves metaheuristic applications. The literature review of this thesis denotes the use of the genetic algorithm as a benchmark. Then, an NSGAII is selected to explore its performance in MOSCLSCN.

2.3.1 Non-dominated sorting genetic algorithm II

The non-dominated sorting genetic algorithm II (NSGAII) (Deb et al., 2002) is a well-known and widely used evolutionary algorithm for multi-objective optimization. It is an improvement over the previous proposal (Srinivas and Deb, 1994). It is a benchmark algorithm. The strategies in NSGAII include elitism to select the best solutions by rank, non-dominated sorting for convergence, and the use of crowding distance to maintain diversity.

NSGAII is used as a benchmark or a hybridization method, and it produces competitive results. As a hybridization approach, Mogale et al. (2022) applied a hybrid NSGAII to obtain an initial solution and a surrogate model for an SCLSCN problem with four facility decisions. As a benchmark, Tirkolaee et al. (2022) compared NSGAII with MOGWO, and MOGWO performed better in an SCLSCN with six potential facilities. Also, Gholipour et al. (2023) used NSGAII and MOPSO. The NSGAII achieved better results than MOPSO in a model with three location facilities. Yamchi et al. (2023) worked with NSGAII, NRGA, and NSGAIII. NRGA outperformed with two location decisions.

The following section details the mathematical model, solution methods, data description, and performance metrics.

Bibliography

Abbasi, S., Daneshmand-Mehr, M., and Kanafi, A. G. (2023). Designing a tri-objective, sustainable, closed-loop, and multi-echelon supply chain during the covid-19 and lockdowns. *Foundations of Computing and Decision Sciences*, 48:269–312.

Akbari-Kasgari, M., Khademi-Zare, H., Mohammad, Fakhrzad, B., Hajiaghaei-Keshteli, M., and Honarvar, M. (2022). Designing a resilient and sustainable closed-loop supply chain network in copper industry. *Clean Technologies and Environmental Policy*, 24:1553–1580.

Amin, S. H., Zhang, G., and Eldali, M. N. (2020). A review of closed-loop supply chain models. *Journal of Data, Information and Management*, 2:279–307.

Angelelli, E., Mansini, R., Speranza, M. G., et al. (2007). Kernel search: a heuristic framework for milp problems with binary variables. In *Technical Report RT 2007-04-56, Department of Electronics for Automation, University of Brescia*. Brescia.

Archetti, C., Guastaroba, G., Huerta-Muñoz, D. L., and Speranza, M. G. (2021). A kernel search heuristic for the multivehicle inventory routing problem. *International Transactions in Operational Research*, 28:2984–3013.

Asif, F. M., Bianchi, C., Rashid, A., and Nicolescu, C. M. (2012). Performance analysis of the closed loop supply chain. *Journal of Remanufacturing*, 2:1–21.

Becerra, P., Mula, J., and and, R. S. (2024). Optimising location, inventory and transportation in a sustainable closed-loop supply chain. *International Journal of Production Research*, 62(5):1609–1632.

Borčinová, Z. (2022). Kernel search for the capacitated vehicle routing problem. *Applied Sciences 2022, Vol. 12, Page 11421*, 12:11421.

Boschetti, M. A., Maniezzo, V., Roffilli, M., and Röhler, A. B. (2009). Matheuristics: Optimization, simulation and control. *Lecture Notes in Computer Science* *(including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5818 LNCS:171–177.

Church, R. L. and Drezner, Z. (2022). 105468 survey in operations research and management science. *Computers & Operations Research*, 138.

Cornuéjols, G., Nemhauser, G., and Wolsey, L. (1983). The uncapicitated facility location problem. Technical report, Cornell University Operations Research and Industrial Engineering.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197.

Ferguson, M. E. and Souza, G. C. (2010). Closed-loop supply chains: New developments to improve the sustainability of business practices. *Closed-Loop Supply Chains: New Developments to Improve the Sustainability of Business Practices*, pages 1–241.

Filippi, C., Guastaroba, G., Huerta-Muñoz, D. L., and Speranza, M. G. (2021). A kernel search heuristic for a fair facility location problem. *Computers and Operations Research*, 132.

Fleischmann, M., Beullens, P., Bloemhof-Ruwaard, J. M., and Wassenhove, L. N. V. (2001). The impact of product recovery on logistics network design. *Production and Operations Management*, 10:156–173.

Gholipour, A., Sadegheih, A., Mostafaeipour, A., Mohammad, Fakhrzad, B., Mostafaeipour, A., and Bagher, M. (2023). Designing an optimal multi-objective model for a sustainable closed-loop supply chain: a case study of pomegranate in iran. *Environment, Development and Sustainability.*

Goodarzian, F., Ghasemi, P., Gonzalez, E. D. S., and Tirkolaee, E. B. (2023). A sustainable-circular citrus closed-loop supply chain configuration: Pareto-based algorithms. *Journal of Environmental Management*, 328:116892.

Govindan, K., Soleimani, H., and Kannan, D. (2015). Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. *European Journal of Operational Research*, 240:603–626.

Hajiaghaei-Keshteli, M. and Fard, A. M. F. (2019). Sustainable closed-loop supply chain network design with discount supposition. *Neural Computing and Applications*, 31:5343–5377.

Irawan, C. A., Abdulrahman, M. D.-A., Salhi, S., and Luis, M. (2022). An efficient matheuristic algorithm for bi-objective sustainable closed-loop supply chain networks. *IMA Journal of Management Mathematics*, 33:603–636.

Jauhari, W. A. and Wakhid, P. A. (2026). A multi-echelon closed-loop agricultural supply chain network problem for corn industry with waste recycling and carbon regulation. *Modeling Earth Systems and Environment*, 12(1):6.

Jayarathna, C. P., Agdas, D., Dawes, L., and Yigitcanlar, T. (2021). Multi-objective optimization for sustainable supply chain and logistics: A review. *Sustainability (Switzerland)*, 13:13617.

Jebreili, S., Babazadeh, R., Fazayeli, S., A. Kamran, M., and Gharibi, A. R. (2025). A multi-objective milp model for sustainable closed-loop supply chain network design: Evidence from the wood–plastic composite industry. *Mathematics*, 13(21).

Joshi, S. (2022). A review on sustainable supply chain network design: Dimensions, paradigms, concepts, framework and future directions. *Sustainable Operations and Computers*, 3:136–148.

Khorshidvand, B., Soleimnai, H., Mehdi, M., Esfahani, S., and Sibdari, S. (2021). Sustainable closed-loop supply chain network: Mathematical modeling and lagrangian relaxation. *Journal of Industrial Engineering and Management Studies*, 8:240–260.

Krarup, J. and Pruzan, P. M. (1983). The simple plant location problem: Survey and synthesis. *European Journal of Operational Research*, 12(1):36–81.

Maniezzo, V., Boschetti, M. A., Stützle, T., Editors, S., Speranza, M. G., and Fernando, J. (2021). *Matheuristics*. Springer International Publishing.

Mehrjerdi, Y. Z. and Shafiee, M. (2020). Multiple-sourcing in sustainable closed-loop supply chain network design: Tire industry case study. *International Journal of Supply and Operations Management*, 7:202–221.

Mirzaei, M. G., Goodarzian, F., Mokhtari, K., Yazdani, M., and Shokri, A. (2023). Designing a dual-channel closed loop supply chain network using advertising rate and price-dependent demand: Case study in tea industry. *Expert Systems with Applications*, 233:120936.

Mogale, D. G., De, A., Ghadge, A., and Aktas, E. (2022). Multi-objective modelling of sustainable closed-loop supply chain network with price-sensitive demand and consumer's incentives. *Computers & Industrial Engineering*.

Momeni, M. A., Jain, V., and Bagheri, M. (2024). A multi-objective model for designing a sustainable closed-loop supply chain logistics network. *Logistics 2024, Vol. 8, Page 29*, 8:29.

Pazhani, S. and Ravindran, A. R. (2018). A bi-criteria model for closed loop supply chain network design. *International Journal of Operational Research*, 31:330–356.

Restrepo Diaz, J. D. and Amin, S. H. (2025). A multi-objective optimization approach for sustainable management of computers e-waste in a closed-loop supply chain network. *Journal of Cleaner Production*, 506:145494.

Rodríguez-Escoto, J.-N., Nucamendi-Guillén, S., Olivares-Benitez, E., and Drzymalski, J. (2026). Circular economy modeling: A multiobjective closed-loop sustainable supply chain problem solved by kernel search. *Mathematics*, 14(5).

Rodríguez-Escoto, J.-N., Olivares-Benitez, E., Nucamendi-Guillén, S., and Drzymalski, J. (2025). A multi-objective sustainable closed-loop supply chain network problem with hybrid facilities. *International Transactions in Operational Research*, 32(6):3497–3527.

Sarkis, J. and Dou, Y. (2017). *Green Supply Chain Management : A Concise Introduction*. Routledge.

Seydanlou, P., Jolai, F., Tavakkoli-Moghaddam, R., and Fathollahi-Fard, A. M. (2022). A multi-objective optimization framework for a sustainable closed-loop supply chain network in the olive industry: Hybrid meta-heuristic algorithms sustainable closed-loop supply chain multi-objective perishable products olive industry hybrid optimization algorithms. *Expert Systems With Applications*, 203:117566.

Soleimani, H., Chhetri, P., Amir, Fathollahi-Fard, M., Al-E-Hashem, S. M. J. M., and Shahparvari, S. (2022). Sustainable closed-loop supply chain with energy efficiency: Lagrangian relaxation, reformulations and heuristics. *Annals of Operations Research*, 318:531–556.

Soleimani, H. and Kannan, G. (2015). A hybrid particle swarm optimization and genetic algorithm for closed-loop supply chain network design in large-scale networks. *Applied Mathematical Modelling*, 39:3990–4012.

Srinivas, N. and Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, 2(3):221–248.

Tirkolaee, E. B., Goli, A., Ghasemi, P., and Goodarzian, F. (2022). Designing a sustainable closed-loop supply chain network of face masks during the covid-19 pandemic: Pareto-based algorithms. *Journal of Cleaner Production*, 333:130056.

Yamchi, H. R., Jabarzadeh, Y., Govindan, K., and Mahdiraji, H. A. (2023). A triple bottom line approach for designing a sustainable closed-loop supply chain network in fruit industry: A metaheuristic solution approach. *Journal of the Operational Research Society*.

Yun, Y., Chuluunsukh, A., and Gen, M. (2020). Sustainable closed-loop supply chain design problem: A hybrid genetic algorithm approach. *Mathematics 2020, Vol. 8, Page 84*, 8:84.

Zhang, Y., Che, A., and Chu, F. (2022). Improved model and efficient method for bi-objective closed-loop food supply chain problem with returnable transport items. *International Journal of Production Research*, 60:1051–1068.

Zhang, Y., Chu, F., Che, A., and Li, Y. (2024). Closed-loop inventory routing problem for perishable food with returnable transport items selection. *International Journal of Production Research*, 62:501–521.

Chapter 3

Methodology

This chapter explains a multi-objective sustainable closed-loop supply chain network using a hybrid facilities model. Its network structure, notation, and formulation are described. Then, the solution methods, the dataset characteristics used for testing, and the performance metrics are detailed.

3.1 Formal problem definition

A multi-objective sustainable closed-loop supply chain network (MOSCLSCN) with a hybrid facility model is proposed. The MOSCLSCN takes into account the economic, environmental, and social factors, aiming to minimize (1) the overall economic cost of the supply chain, (2) the total CO_2 emissions produced by the vehicles involved, and (3) the total population affected by the obnoxiousness.

In the forward flow, a number of suppliers provide raw materials to a number of manufacturers. From there, finished products are distributed to a number of retailers, and retailer demand must be met using a required combination of two types of intermediate nodes: warehouses and hybrid facilities, where hybrid facilities serve as storage point and also as disassembly operation for recycling/remanufaturing/disposal.

In the reverse flow, this is where we "close the loop." Used products are collected from retailers and sent to hybrid facilities, where they are processed for recycling and reintegrated into manufacturing plants, accounting for return and disposal rates. Figure 3.1 shows the network structure. This structured network is represented mathematically, taking the following assumptions:

- Retailer demand is assumed to be deterministic, and the costs associated with travel between locations are calculated considering Euclidean distances and volume transported [Distance $\rightarrow$ cost].
- The emission factor converts the cost associated to transported distance into an estimated CO_2 per origin-destination pair [Distance $\rightarrow$ cost $\rightarrow$ CO_2].
- For every product, a standard deviation is computed considering the variability of total demand over periods.
- To avoid stockout, a safety stock is considered based on a determined service level.
- The model incorporates facility assignments but does not address the specifics of routing logistics.
- Hybrid facilities are designed for storage and recycling/remanufacturing/disposal.
- To fulfill retailer demand, a combination of warehouses and hybrid facilities is required, as neither type alone is sufficient.
- The available vehicle fleet has limited capacity, and the model does not account for constraints specific to electric vehicles, such as recharging requirements or range limitations.

3.2 Mathematical formulation

3.2.1 Mathematical notation

The model indices, problem parameters, and decision variables are stated as follows. Table 3.1 displays the indices that run over their sets, along with the sums, parameters, and variables. Table 3.2 describes the parameters of the model. Also, the computation of the parameters for the different instances is described in section 3.3.5. Table 3.3 denotes the variables of the model.

3.2.2 Mathematical model

The model objective functions are presented in the following equations:

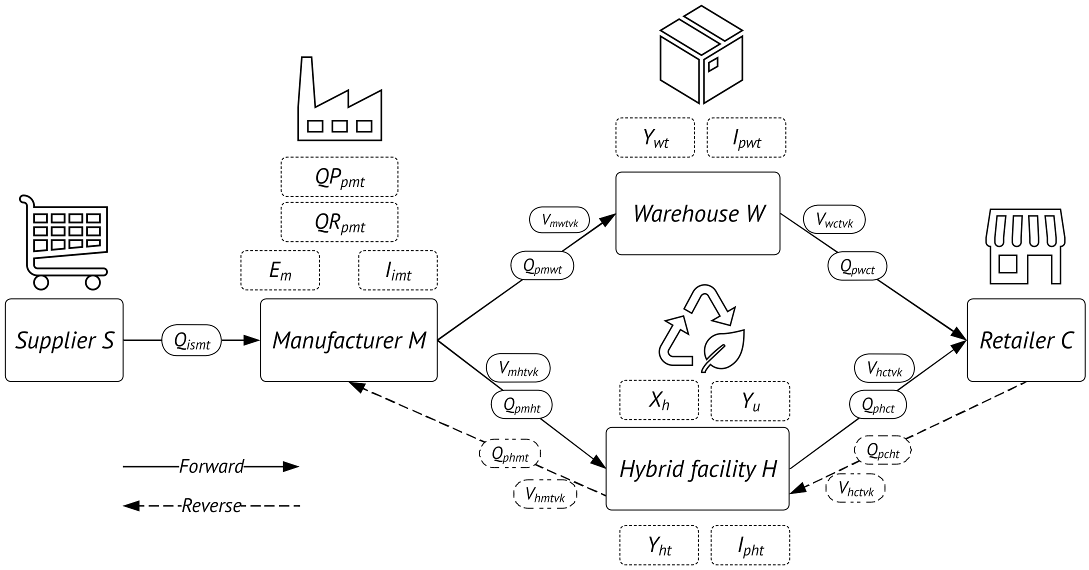


Figure 3.1: SCLSCN network

Table 3.1: Model indexed sets

Indices	**Sets**	**Description**
s	Supplier	$s = \{1, ..., S\}$
m	Manufacturer	$m = \{1, ..., M\}$
w	Warehouse	$w = \{1, ..., W\}$
c	Retailer	$c = \{1, ..., C\}$
p	Products	$p = \{1, ..., P\}$
i	Raw Materials	$i = \{1, ..., I\}$
t	Time period	$t = \{1, ..., T\}$
h	Hybrid facility	$h = \{1, ..., H\}$
v	Type of vehicle	$v = \{1, ..., V\}$
k	Size of vehicle	$k = \{1, ..., K\}$
u	Urban centers	$u = \{1, ..., U\}$

Table 3.2: Model parameters

Notation	Description
IC_{im}	Cost of inventory per unit of raw material i at manufacturer m
IC_{pw}	Cost of inventory per unit of product p at warehouse w
IC_{ph}	Cost of inventory per unit of product p at hybrid facility h
TC_{ism}	Cost per unit of raw material i when supplied by s to m
TC_{mw}	Cost of transport per unit weight from manufacturer m to warehouse w
TC_{mh}	Cost of transport per unit weight from manufacturer m to hybrid h
TC_{wc}	Cost of transport per unit weight from warehouse w to retailer c
TC_{hc}	Cost of transport per unit weight from hybrid h to retailer c
TC_{ch}	Cost of transport per unit weight from retailer c to hybrid h
TC_{hm}	Cost of transport per unit weight from hybrid h to manufacturer m
PC_{pm}	The production expense of product p at manufacturer m
RC_{pm}	The recycling expense of product p at manufacturer m
MC_{pm}	Manufacturing capacity limit of product p per manufacturer m
FC_w	Opening cost for warehouse w
FC_h	Opening cost for hybrid facility h
FC_m	Opening cost for manufacturer m
SC_w	Capacity for storage at warehouse w
SC_h	Capacity for storage at hybrid facility h
SC_m	Capacity for storage at manufacturer m
D_{pct}	Retailer c demand of product p in period t
A_{pct}	Return rate of product p available in retailer c at period t
W_p	Weight of product p
B_p	Disposal rate of product p
SDS_p	Sales standard deviation of product p
L_p	Average lead time for product p at retailers
SL	Service level for the inventory safety stock
MB	Big number for the opening manufacturer m
e	Number near to 1
OFF_{ip}	Off-take of raw material i for product p
VC_{vk}	Vehicle cost of type v and size k
VQ_{vk}	Vehicle capacity of type v and size k
CO_{vk}	CO_2 emissions factor of vehicle v and size k
DUC_{uh}	Distance from urban centers u to hybrid facilities h
POP_u	The population of urban centers u
SR	A zone of negative influence surrounding a facility

Table 3.3: Model variables

Notation	Description
Q_{ismt}	Amount of raw material i acquired from supplier s by manufacturer m in period t
P_{pmt}	Amount of product p produced by manufacturer m in period t
R_{pmt}	Amount of product p recycled at manufacturing plant m in period t
Q_{pmwt}	Amount of product p shipped from manufacturer m to warehouse w in period t
Q_{pmht}	Amount of product p shipped from manufacturer m to hybrid facility h in period t
Q_{pwct}	Amount of product p shipped from warehouse w to retailer c in period t
Q_{phct}	Amount of product p shipped from hybrid facility h to retailer c in period t
Q_{pcht}	Amount of product p shipped from retailer c to hybrid facility h in period t
Q_{phmt}	Amount of product p shipped from hybrid facility h to manufacturer m in period t
DQ_{phmt}	An auxiliary variable to linearize where Q_{phmt} is involved
I_{imt}	Inventory of raw material i available at manufacturer m in period t
I_{pwt}	Product inventory p available in warehouse w in period t
I_{pht}	Product inventory p available at hybrid facility h in period t
V_{mwtvk}	Number of vehicles of type v and size k chosen for transportation between manufacturer m, warehouse w in period t
V_{wctvk}	Number of vehicles of type v and size k chosen for transportation between warehouse w to retailer c in period t
V_{mhtvk}	Number of vehicles of type v and size k chosen for transportation between manufacturer m to hybrid facility h in period t
V_{hctvk}	Number of vehicles of type v and size k chosen for transportation between hybrid facility h to retailer c in period t
V_{chtvk}	Number of vehicles of type v and size k chosen for transportation between retailer c to hybrid facility h in period t
V_{hmtvk}	Number of vehicles of type v and size k chosen for transportation between hybrid facility h to manufacturer m in period t
Y_{wt}	1 if warehouse w is active during period t, and 0 if it is inactive
Y_{ht}	1 if hybrid facility h is active during period t, and 0 if it is inactive
E_m	1 if manufacturer m is opened, and 0 if it is closed
X_h	1 if hybrid facility h is closed and unoccupied, and 0 if site h is operational
Z_u	1 if urban center u is within the negative impact zone of any hybrid facility, and 0 otherwise
SH	Number of hybrid facilities opened

$$
\begin{aligned}
&\mathit{Minimize}\ F_1 = \\
&\sum_i^I \sum_s^S \sum_m^M \sum_t^T TC_{ism} * Q_{ismt} + \sum_p^P \sum_m^M \sum_t^T PC_{pm} * P_{pmt} + \\
&\sum_p^P \sum_m^M \sum_t^T RC_{pm} * R_{pmt} + \sum_p^P \sum_t^T W_p * \Bigg(\sum_m^M \sum_w^W TC_{mw} * Q_{pmwt} + \\
&\sum_m^M \sum_h^H TC_{mh} * Q_{pmht} + \sum_w^W \sum_c^C TC_{wc} * Q_{pwct} + \sum_h^H \sum_c^C TC_{hc} * Q_{phct} + \\
&\sum_c^C \sum_h^H TC_{ch} * Q_{pcht} + \sum_h^H \sum_m^M TC_{hm} * Q_{phmt} \Bigg) + \\
&\sum_i^I \sum_m^M \sum_t^T IC_{im} * I_{imt} + \sum_p^P \sum_w^W \sum_t^T IC_{pw} * I_{pwt} + \sum_p^P \sum_h^H \sum_t^T IC_{ph} * I_{pht} + \\
&\sum_m^M FC_m * E_m + \sum_w^W \sum_t^T FC_w * Y_{wt} + \sum_h^H \sum_t^T FC_h * Y_{ht} + \\
&\sum_t^T \sum_v^V \sum_k^K VC_{vk} * \Bigg(\sum_m^M \sum_w^W V_{mwtvk} + \sum_m^M \sum_h^H V_{mhtvk} + \sum_w^W \sum_c^C V_{wctvk} + \\
&\sum_h^H \sum_c^C V_{hctvk} + \sum_c^C \sum_h^H V_{chtvk} + \sum_h^H \sum_m^M V_{hmtvk} \Bigg),
\end{aligned}
\tag{3.1}
$$

$$
\begin{aligned}
&\mathit{Minimize}\ F_2 = \\
&\sum_t^T \sum_v^V \sum_k^K CO_{vk} * \Bigg(\sum_m^M \sum_w^W TC_{mw} * V_{mwtvk} + \sum_m^M \sum_h^H TC_{mh} * V_{mhtvk} + \\
&\sum_w^W \sum_c^C TC_{wc} * V_{wctvk} + \sum_h^H \sum_c^C TC_{hc} * V_{hctvk} + \sum_c^C \sum_h^H TC_{ch} * V_{chtvk} + \\
&\sum_h^H \sum_m^M TC_{hm} * V_{hmtvk} \Bigg),
\end{aligned}
\tag{3.2}
$$

$$
\begin{aligned}
&\mathit{Minimize}\ F_3 = \\
&\sum_u^U POP_u * Z_u,
\end{aligned}
\tag{3.3}
$$

Subject to:

$$\sum_{i}^{I}\sum_{s}^{S} Q_{ismt} + \sum_{i}^{I} I_{imt-1} + \sum_{p}^{P}\sum_{h}^{H} Q_{phmt} \leq SC_m, \qquad \forall\ m, t \quad (3.4)$$

$$\sum_{s}^{S} Q_{ismt} + I_{imt-1} = I_{imt} + \sum_{p}^{P}(P_{pmt} * OFF_{ip}), \qquad \forall\ i, m, t \quad (3.5)$$

$$R_{pmt} = \sum_{h}^{H} Q_{phmt}, \qquad \forall\ p, m, t \quad (3.6)$$

$$P_{pmt} \leq MC_{pm}, \qquad \forall\ p, m, t \quad (3.7)$$

$$\sum_{t}^{T} P_{pmt} + \sum_{t}^{T} R_{pmt} \leq MB * E_m, \qquad \forall\ p, m \quad (3.8)$$

$$P_{pmt} + R_{pmt} = \sum_{w}^{W} Q_{pmwt} + \sum_{h}^{H} Q_{pmht}, \qquad \forall\ p, m, t \quad (3.9)$$

$$\sum_{p}^{P} I_{pwt-1} + \sum_{p}^{P}\sum_{m}^{M} Q_{pmwt} \leq SC_w * Y_{wt}, \qquad \forall\ w, t \quad (3.10)$$

$$\sum_{p}^{P} I_{pht-1} + \sum_{p}^{P}\sum_{m}^{M} Q_{pmht} + \sum_{p}^{P}\sum_{c}^{C} Q_{pcht} \leq SC_h * Y_{ht}, \qquad \forall\ h, t \quad (3.11)$$

$$I_{pwt-1} + \sum_{m}^{M} Q_{pmwt} = I_{pwt} + \sum_{c}^{C} Q_{pwct}, \qquad \forall\ p, w, t \quad (3.12)$$

$$I_{pht-1} + \sum_{m}^{M} Q_{pmht} = I_{pht} + \sum_{c}^{C} Q_{phct}, \qquad \forall\ p, h, t \quad (3.13)$$

$$\sum_{w}^{W} I_{pwt} \geq SL * SDS_p * \sqrt{L_p}, \qquad \forall\ p, t \in T-1, \quad (3.14)$$

$$\sum_{h}^{H} I_{pht} \geq SL * SDS_p * \sqrt{L_p}, \qquad \forall\ p, t \in T-1, \quad (3.15)$$

$$\sum_{w}^{W} Q_{pwct} + \sum_{h}^{H} Q_{phct} >= D_{pct}, \qquad \forall\ p, c, t \quad (3.16)$$

$$\sum_{c}^{C}(1 - B_p) * Q_{pcht} - \sum_{m}^{M} DQ_{phmt} \geq 0, \qquad \forall\ p, h, t \quad (3.17)$$

$$\sum_{c}^{C}(1 - B_p) * Q_{pcht} - \sum_{m}^{M} DQ_{phmt} - e \leq 0, \qquad \forall\ p, h, t \quad (3.18)$$

$$Q_{phmt} = DQ_{phmt}, \qquad \forall\ p,h,m,t \quad (3.19)$$

$$\sum_{h}^{H} Q_{pcht} = \lceil (A_{pct} * D_{pct}) \rceil, \qquad \forall\ p,c,t \quad (3.20)$$

$$\sum_{v}^{V} \sum_{k}^{K} VQ_{vk} * V_{mwtvk} \geq Q_{pmwt}, \qquad \forall\ p,m,w,t \quad (3.21)$$

$$\sum_{v}^{V} \sum_{k}^{K} VQ_{vk} * V_{mhtvk} \geq Q_{pmht}, \qquad \forall\ p,m,h,t \quad (3.22)$$

$$\sum_{v}^{V} \sum_{k}^{K} VQ_{vk} * V_{wctvk} \geq Q_{pwct}, \qquad \forall\ p,w,c,t \quad (3.23)$$

$$\sum_{v}^{V} \sum_{k}^{K} VQ_{vk} * V_{hctvk} \geq Q_{phct}, \qquad \forall\ p,h,c,t \quad (3.24)$$

$$\sum_{v}^{V} \sum_{k}^{K} VQ_{vk} * V_{chtvk} \geq Q_{pcht}, \qquad \forall\ p,c,h,t \quad (3.25)$$

$$\sum_{v}^{V} \sum_{k}^{K} VQ_{vk} * V_{hmtvk} \geq Q_{phmt}, \qquad \forall\ p,h,m,t \quad (3.26)$$

$$\sum_{h}^{H} X_h = H - SH, \qquad (3.27)$$

$$X_h + Z_u \geq 1, \qquad \forall\ u,h | DUC_{uh} \leq SR \quad (3.28)$$

$$X_h = 1 - Y_{ht}, \qquad \forall\ h,t \quad (3.29)$$

$$I_{imt} = 0, \qquad \forall\ i,m,t\{0\} \quad (3.30)$$

$$I_{pwt} = 0, \qquad \forall\ p,w,t\{0\} \quad (3.31)$$

$$I_{pht} = 0, \qquad \forall\ p,h,t\{0\} \quad (3.32)$$

$$Q_{ismt}, P_{pmt}, R_{pmt}, Q_{pmwt}, Q_{pmht}, Q_{pwct}, Q_{phct} \in \mathbb{Z}_0^+, \qquad (3.33)$$

$$Q_{pcht}, Q_{phmt}, DQ_{phmt}, I_{imt}, I_{pwt}, I_{pht} \in \mathbb{Z}_0^+, \qquad (3.34)$$

$$V_{mwtvk}, V_{wctvk}, V_{mhtvk}, V_{hctvk}, V_{chtvk}, V_{hmtvk} \in \mathbb{Z}_0^+, \qquad (3.35)$$

$$Y_{wt}, Y_{ht}, E_m, X_h, Z_u \in \{0,1\}, \qquad (3.36)$$

$$0 \leq SH \leq H \in \mathbb{Z}^+ \qquad (3.37)$$

Objective functions: The first objective function, (3.1), minimizes the total economic cost of the supply chain, encompassing variable costs (purchasing from suppliers, production, and recycling expenses at the plant), transportation costs (calculated by unit weight and Euclidean distance between all nodes), fixed costs (opening manufacturers, warehouses, and hybrid facilities), inventory management, and vehicle fleet usage costs.

The second objective function (3.2) aims to reduce total CO_2 emissions generated by vehicles transporting goods across supply chain echelons. Here, the model makes a critical tactical decision: fleet selection. It must choose between internal combustion and electric vehicles, each with different costs (VC_{vk}), capacities (VQ_{vk}), and emission factors (CO_{vk}), to find the combination that minimizes the carbon footprint across all echelons. Specifically, CO_{vk} helps convert transportation costs into CO_2 emission units.

The social innovation of this model lies in the function F_3, which minimizes the obnoxiousness caused by the hybrid facilities' pollution. We define a negative influence zone (SR) around hybrid facilities, driven by pollution (noise, emission, odor, biological, etc.). The goal is to ensure that urban centers (u) with high population densities (POP_u) are not affected by being located within this impact radius.

Manufacturer, flow material, and material constraints: Manufacturer storage constraints (3.4) limit total raw material i inventory (s, h), and carryover stock from period $t-1$ based on manufacturer capacity m. The manufacturer's material flow constraints (3.5) balance raw material purchases, accounting for off-take (OFF_{ip}) and manufacturer m's production. The recycling constraints (3.6) ensure that all product p shipped from manufacturer m to the hybrid facility h at period t is recycled. The manufacturer's production capacity, as constrained by (3.7), limits the manufacturer's m capacity. Constraints (3.8) open the corresponding manufacturer m under the Big M approach (the estimation of MB is provided in table 3.5). The product flow constraints (3.9) balance the amount of product p at the manufacturer m, which is sent to the hybrid facility h and the warehouse w.

Storage and inventory constraints: The warehouse and hybrid facility storage constraints (3.10) - (3.11) limit the sum of initial inventory and the product manufactured to the capacity of the warehouse w and the hybrid facility h, respectively.

The inventory flow constraints (3.12) - (3.13) ensure the flow of product p in the warehouse w and the hybrid facility h at period t. Constraints (3.14)-(3.15) determine the same safety stock for warehouse w and the hybrid facility h separately. Where a standard deviation is computed considering the variability of total demand over periods for every product p. Also, safety stock is determined based on a specified service level and lead time for every product p (see table 3.5).

Demand and returns constraints: The demand constraints (3.16) guarantee that the demand of product p for retailer c at period t is met. Constraints (3.17)-(3.19) ensure the flow of the returned products, accounting for the disposal rate (B_p). The product return flow constraints (3.20) are determined by a return rate (A_{pct}).

Operational constraints: The constraints (3.21) - (3.26) calculate the vehicles needed by the echelon. The constraint (3.27) ensures that all but SH (selected hybrid facility) is left vacant. The constraints (3.28) encourage that the hybrid facility h is not chosen if the urban center u is located within the negative impact zone ($DUC_{uh} \leq SR$). The constraints (3.29) specify the relation between the variable not chosen, X_h, and the opening Y_{ht} of the hybrid facility h.

Initial inventory constraints and variable's domains: The initial inventory is declared in (3.30) - (3.32). The variable's domains are settled in (3.33) - (3.37).

The model addresses a MOSCLSCN encompassing facility location, inventory levels, and vehicle selection decisions, by integrating economic, environmental, and social factors as objectives. The model offers a more realistic approach to supply chain design by addressing sustainability through hybrid facility incorporation, consideration of obnoxiousness, and vehicle selection based on CO_2 emissions.

The following section outlines the key aspects of the computational experiment, including the implemented solution methods, the dataset characteristics, and the metrics used to evaluate the methods' performance.

3.3 Computational experience

This section outlines solution methods used for tackling the MOSCLSCN problem, including the exact method (i.e., the improved augmented epsilon-constraint, AUG2), the matheuristic (i.e., the kernel search, KS) matheuristic applied to AUG2, and the metaheuristic (i.e., the hybrid non-dominated sorting genetic algorithm II, HNSGAII).

3.3.1 Improved Augmented Epsilon-constraint method

To solve the MOSCLSCN, the Improved Augmented Epsilon-constraint (AUG2) method is selected because it efficiently computes a high-quality Pareto front. Developed by Mavrotas and Florios in 2013 and subsequently by the authors themselves, AUG2 enhances earlier methods by using a scalarization strategy that converts all but one objective into constraints.

The objective (3.1) serves as the main objective function (3.38), which is penalized by multiplying the epsilon ($\epsilon = 10^{-6}$) parameter with the surplus variables s_k normalized with their ranges r_k ($k = 2, 3$). The remaining objectives (3.2) and (3.3) are transformed into constraints (3.39) and (3.40) by incorporating the surplus variable s_k and the constraint's right-hand side e_k derived from the scalarization process. The AUG2 application to the MOSCLSCN model can be mathematically formulated as follows:

$$\textit{Minimize } MO\Big\{F_1 + \epsilon \times \Big(\frac{s_2}{r_2} + 10^{-1}\frac{s_3}{r_3}\Big)\Big\} \tag{3.38}$$

Subject to:

$$F_2 + s_2 = e_2 \tag{3.39}$$

$$F_3 + s_3 = e_3 \tag{3.40}$$

and constraints (3.4) - (3.37).

The AUG2 algorithm starts by obtaining the lexicographic chart (see Figure 3.2 part A, solid lines). That payoff matrix will function as an input to calculate the upper bound (ub_k) and lower bound (lb_k) of the objectives F_k. After that, the calculation of the range $r_k = ub_k - lb_k$ is performed, and the size of the iteration movement $Step_k = r_k/q_k$ is obtained (q_k denotes the predefined number of points) to create the epsilon grid. The iteration starts with $t_k = 0$ and every movement is updated by $e_k = ub_k - t_k * Step_k$. If the solution is feasible, it is saved, and the bypass coefficient, b_k, is calculated to skip further iterations if the slack (s_k) is greater than the $Step_k$ value. Otherwise, it is updated with the standard increment. However, if it is infeasible, the algorithm is concluded.

3.3.2 AUG2-EXTENDED

As stated previously, in the AUG2, each objective function is minimized in the lexicographic step, leaving the others unrestricted. During the experiment, F_2 exceeded the established time limit of 7200 seconds per iteration. Therefore, to overcome this issue, AUG2-EXTENDED is applied (see Figure 3.2 part B, dashed lines), using linear programming relaxation consisting of changing the actual nature of the variable (Eq.(3.35)) that is represented as $x \in \mathbb{Z}_+^n$ into $\mathbb{R}_+^n$, which resulted in a bound lower than the integer lower bound with an average gap difference of 0.58% and a maximum value gap of 0.89%, that overcomes the time limit issue. Additionally, the gap in the solution leads to premature infeasibility in the final iterations of e_2. To deal with the infeasible solution caused by the lower bound relaxation, an extension of another cycle is used to obtain solutions between the previous iteration and the infeasible iteration of e_2. The new upper bound is calculated $e_{22} = ub_2 - (t_2 - 1) * Step_2$ to update the value of r_{22} and $Step_{22}$ with the same grid points ($q_{22} = 10$).

Four different AUG2-based variations were implemented, as follows:

- INT: AUG2 application (vehicle variables are integers).
- PR: AUG2 application with F_2 relaxed (vehicle variables become continuous).
- INTX: AUG2-EXTENDED application (vehicle variables are integers).
- PRX: AUG2-EXTENDED application with F_2 relaxed (vehicle variables become continuous).

The following section describes the kernel search matheuristic algorithm and its application to MOSCLSCN.

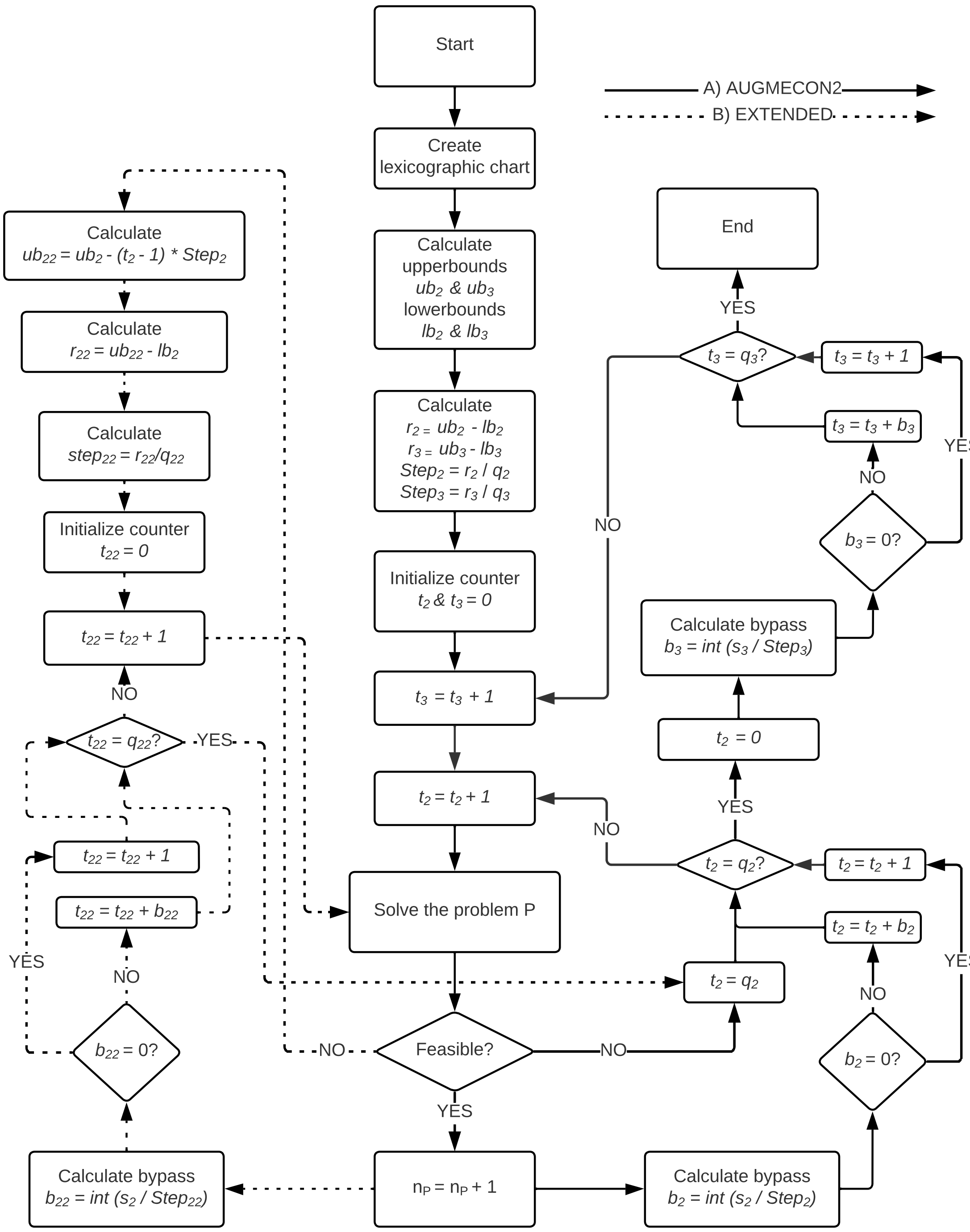


Figure 3.2: AUGMECON2 & EXTENDED diagram

3.3.3 Kernel Search matheuristic

The kernel search matheuristic is a two-phase framework that divides the main problem into sub-problems to be solved efficiently, including the initialization and improvement phases (see Figure 3.3). The kernel search algorithm is a matheuristic documented in the book by Maniezzo et al. (2021), which clearly explains its application to solve problems with binary variables, as demonstrated in Archetti et al. (2021); Borčinová (2022); Filippi et al. (2021); Zhang et al. (2022, 2024). In our work, a kernel search matheuristic for integer variables is implemented.

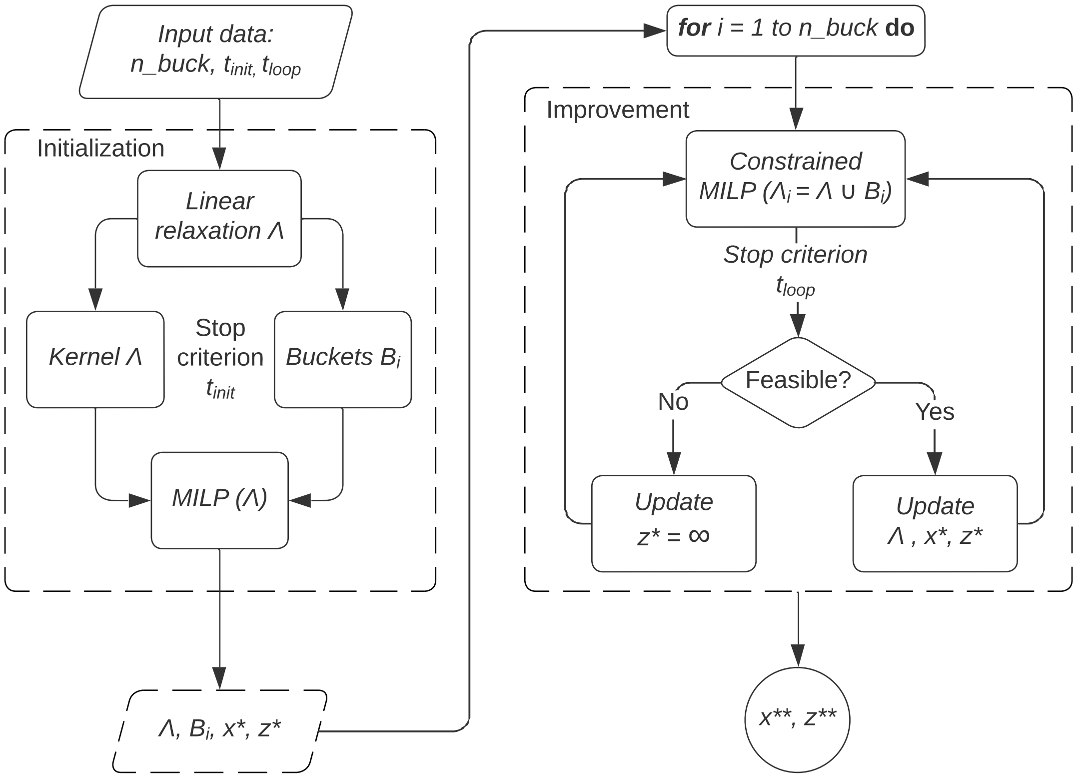


Figure 3.3: Kernel search matheuristic

The input parameters t_{init}_t_{loop}, *n_buck* are stated to be initialized. The initialization phase starts with linear relaxation, with time t_{init} applied to obtain the promising variables ($x > 0$) and the kernel (Λ). The remaining variables are sorted by decreasing reduced cost and divided into *n_buck* sets, B_i. Then, the mixed integer linear programming (MILP) is solved using Λ to obtain the solution x^* and its corresponding cost z^*. The improvement phase applies a heuristic that loops over the constrained MILP, adding the constraint $\sum_{j \in B_i} x_j \geq 1$, which encourages the use of at least one variable x_j from B_i. It also includes a cutoff constraint $z < z^*$ and a time limit t_{loop}. If the solution is feasible, it updates the kernel Λ, the solution x^*, and its cost z^*. If not, the cost $z^* = \infty$, so the process can continue. Finally, the algorithm reports the best solution, x^{**}, and its corresponding cost, z^{**}.

Algorithm 1 KS matheuristic for MOSCLSCN

Input: n_buck, t_{init}, t_{loop}
Output: x^{**}, z^{**}
▷ **1. Initialization**
Let F = MOMILP formulation, where the vector x storage the values of V_{**tvk} variables, and z the value of F_2;
Solve LP-Relaxation of F with a time limit t_{init};
Sort the variables x results by decreasing reduced costs;
Build the initial kernel Λ and a sequence of n_buck buckets from x obtained ;
Solve $F(\Lambda)$ with a time limit t_{init}, let x^* and z^* be the best solution and its cost;
▷ **2. Improvement**
for $i = 1$ to n_buck **do**
    Construct the set $\Lambda_i = \Lambda \cup B_i$;
    Add to $F(\Lambda_i)$ the constraint $\sum_{j \in B_i} x_j \geq 1$;
    Add to $F(\Lambda_i)$ the constraint $z < z^*$;
    Solve $F(\Lambda_i)$ with time limit t_{loop}
    if feasible solution found **then**
        Let x^* and z^* be the best solution and its cost;
        Add to Λ which belong to B_i and have been selected in x^*;
    end if
    if No feasible solution found **then**
        Let $z^* = \infty$
    end if
end for
Let x^{**}, z^{**} be the best solution from the storage solutions x^*, z^*.

The kernel search matheuristic adaptations are applied to the payoff matrix, and also to each iteration of the MOSCLSCN problem. In the case of the lexicographic chart phase, where the objective functions are solved independently (see

Figure 3.2 part A, solid lines), the objective function F_2 (3.2) requires the application of kernel search because it takes more time to solve, also, in each iteration of the AUG2 framework.

The kernel search matheuristic requires the following procedure (see Algorithm 1): In the initialization phase, the linear relaxation is applied by changing domain of the integer variables for the number of vehicles (V_{mwtvk}, V_{wctvk}, V_{mhtvk}, V_{hctvk}, V_{chtvk}, V_{hmtvk}) of the equation (3.35) into continuous variables, and solving MOSCLSCN objective (3.38) with a time limit t_{init}. The obtained variables, greater than zero, are sorted in decreasing order of reduced cost to define the kernel set, using a large number of vehicles as an upper bound, and the remaining variables are treated as buckets with a value of zero. The buckets are divided into n_buck groups to obtain B_i (A large number of vehicles is set as an upper bound when it is activated). After that, it solves the multi-objective mixed integer linear programming (MOMILP) with the kernel Λ, resulting in the solution x^*, where the vehicle's value-utilized variables are stored, and its F_2 value is stored at z^*. Then, in the improvement phase, the constrained MILP (pushed to use at least one variable from bucket B_i with the constraint $\sum_{j \in B_i} x_j \geq 1$ and cutting off constraint $z < z^*$) is executed in a loop to explore the buckets obtained, updating the Λ that is selected in x^* in each iteration.

The following section describes the hybrid NSGAII and its application to MOSCLSCN.

3.3.4 Hybrid Non-dominated Sorting Genetic Algorithm II

The hybrid non-dominated sorting genetic algorithm II with MILP evaluation enables the exploitation of randomization and diversity. Keller (2019) provides a good source of explanation for the algorithm function.

Algorithm 2 Hybrid NSGAII for MOSCLSCN

```
Input: Instance parameters (S, M, W, H, C...)
Output: Pareto Front (F_1, F_2, F_3), Best solutions
▷ 1. Payoff table
Solve min F_1 → (F_1^min, F_1^max)
Solve min F_2 → (F_2^min, F_2^max)
    Set V_{**tvk} ∈ ℝ^n_+
    Sort the variables V_{**tvk} results by decreasing reduced costs;
    Build the A_lb^{**tvk} where V_{**tvk} > 0, and remaining as chromosome A_ub^{**tvk};
Solve min F_3 → (F_3^min, F_3^max)
▷ 2. Generate epsilon grid
Calculate r_k = ub_k − lb_k for k ∈ {2, 3}
Calculate e_k = ub_k − (i × r_k) for i ∈ {0.0, 0.1, . . . , 1.0}
Generate epsilons ← e_2 × e_3 = {(e_2j, e_3i) | e_2j ∈ e_2, e_3i ∈ e_3}
▷ 3. Evolutionary optimization
Initialize NSGAII (pop_size, time_ga, n_gen, prob)
Set operators: Binary sampling, Two-Point crossover, Bit-Flip mutation
Result ← MINIMIZE(Sub − problem, NSGAII)
return Pareto front objectives and decision variables
```

The hybrid NSGAII (see Algorithm 2 and Figure 3.4) starts calculating the lexicographic matrix. When F_2 is minimized, the number of vehicles V_{**tvk} adds complexity to the problem because of the different possible combinations of fleet selection. Therefore, the binary fix variable stated A_{lb}^{**tvk} where the number of vehicles is $V_{**tvk} > 0$, and the remaining number of vehicles is A_{ub}^{**tvk}, the latter set is translated into a chromosome form.

	A_{ub}^{**tvk}									
Initialization	0	0	0	0	0	0	0	0	0	0
Random sample	1	0	1	0	1	1	0	0	1	0

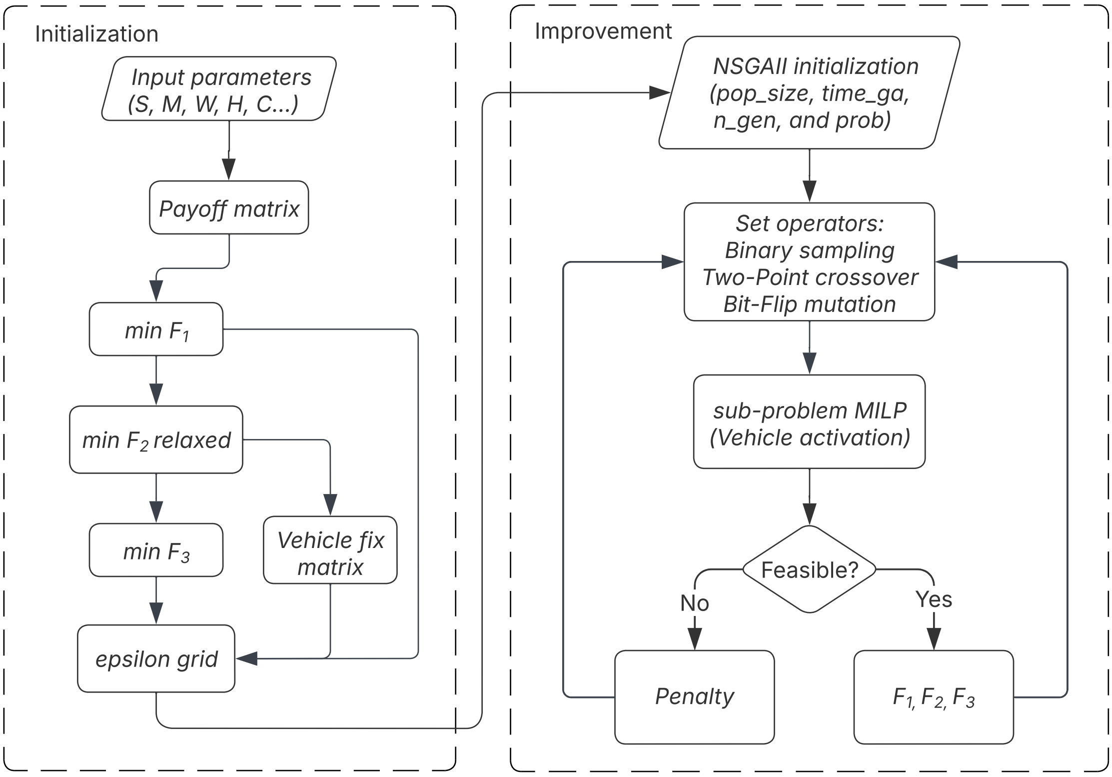


Figure 3.4: Hybrid NSGAII

Algorithm 3 Binary sampling for hybrid NSGAII

procedure BINARY SAMPLING($Problem$)
    Initialize Population $\mathcal{P} \leftarrow \emptyset$
    for $i \leftarrow 1$ **to** pop_size **do**
        for each Variable group $G \in \{A_{ub}^{**tvk}\}$ **do**
            Generate a random binary vector ($prob = 50\%$)
        end for
        Add individual X to $\mathcal{P}$
    end for
    return $\mathcal{P}$
end procedure

The algorithm continues with the epsilon grid, then proceeds to the sub-problem evaluation using the pymoo library (Blank and Deb, 2020). The generator sampling (see Algorithm 3) activates the fixed binary variables randomly with a probability (*prob*).

	A_{lb}^{**tvk}					A_{ub}^{**tvk}									
Initialization	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
Random sample	1	1	1	1	1	1	0	1	0	1	1	0	0	1	0

The chromosome is binary; hence, to achieve balanced disruption and diversity, two-point crossover with two parents (see Algorithm 4) is applied. Two-point crossover performs better than other crossover methods according to Kaya (2011).

Algorithm 4 Two-Point crossover for hybrid NSGAII

procedure TWO-POINT CROSSOVER($Parent_1, Parent_2$)
    Initialize $Offspring_1, Offspring_2$
    for each Variable group G **do**
        if $Rand() < 0.9$ **then**
            Select crossover points P_A, P_B
            Swap genetic material between P_A and P_B
        else
            Keep parents as offspring
        end if
    end for
    return $[Offspring_1, Offspring_2]$
end procedure

The bit-flip mutation (see Algorithm 5) with probability of 0.01 is applied. The bit-flip mutation is a common mutation operator in binary strings (Chicano et al. (2015)), promoting genetic diversity in the population and applied with a low probability to balance exploration and exploitation.

After generating the samples, a decoding procedure for a fixed binary variable and a random selection of epsilons are used as input for the MILP sub-problem, which is solved using *time_ga* (see Algorithm 6). The objective function values are then obtained.

Algorithm 5 Bit-Flip mutation for hybrid NSGAII

procedure BIT-FLIP MUTATION($Individual$)
  for each Variable group G **do**
    Generate a random mask based on probability (0.01)
    $Individual[G] \leftarrow Individual[G] \oplus mask$
  end for
  return $Individual$
end procedure

Algorithm 6 Sub-problem evaluation for hybrid NSGAII

procedure EVALUATE(Individual X)
  ▷ **A. Decode chromosome, map binary, and select epsilons**
  $ActiveVehicles = A^{**tvk}_{ub} \leftarrow \text{DECODE}(X)$
  Select random $[e_2, e_3]$
  ▷ **B. Solve sub-problem (MILP)**
  Pass $A^{**tvk} = A^{**tvk}_{lb} \cup A^{**tvk}_{ub}$ to sub-problem ▷ Fix binary variables
  Pass e_2, e_3 to sub-problem
  Minimize MO equation (3.38)
  Subject to:
    Constraints (3.39) - (3.40) and (3.4) - (3.37).
    Binary structure constraints A^{**tvk}
  Run sub-problem ($time\_ga$)
  ▷ **C. Return results**
  if Solver status is Optimal **or** Feasible **then**
    return $[F_1, F_2, F_3]$
  else
    return $[10^{20}, 10^{20}, 10^{20}]$ ▷ Apply penalty for infeasibility
  end if
end procedure

3.3.5 Dataset description

To ensure reproducibility of the experiments performed on a MOSCLSCN model using the proposed solution approaches, we generated a test dataset that matches the structural and parametric characteristics of the supply chain network introduced by Pazhani et al. (2021). Table 3.4 presents the network parameters for each instance, arranged from left to right. The parameters include the number of suppliers (S), manufacturers (M), warehouses (W), hybrid facilities (H), retailers (C), products (P), raw materials (I), periods (T), type (V) and size (K) of vehicles, urban centers (U), return rate A_{pct}, disposal rate B_p, number of nodes ($\#N$), and the number of paths ($\#P$) of the network.

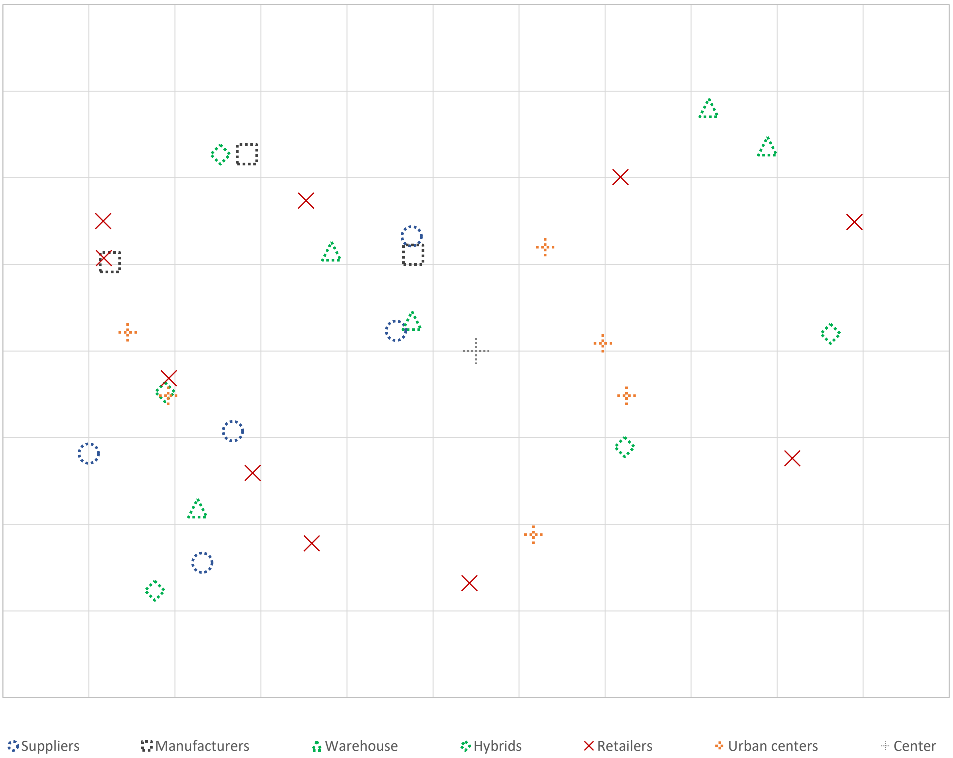


Figure 3.5: Potential locations of INS_1

Figure 3.5 illustrates the network coordinates of the INS_1 instance. The circle represents the suppliers, the manufacturers are denoted by a square, the warehouses by a triangle, the hybrid facilities by a diamond, the retailers by a cross sign, the urban centers by a small plus sign, and the midpoint of the space by a large plus sign.

Table 3.5 describes the range of values considered to generate the parameters (see Table 3.4). A set of test instances for the MOSCLSCN was randomly generated as follows. All facility and customer locations were placed on a 450 $\times$ 300 coordinate plane. The N_coord_x and N_coord_y coordinates of nodes were independently drawn from continuous uniform distributions $U[0, 450]$ and $U[0, 300]$, respectively. Transportation costs between facilities and customers were assumed to be proportional to the Euclidean distances. A specific shipment-related cost parameter (TC_ism) was generated from a continuous uniform distribution $U[1250, 1500]$.

Production costs per unit (PC_{pm}) and recycling costs per unit (RC_{pm}) were drawn from $U[100, 110]$ and $U[75, 90]$, respectively. Fixed opening costs for warehouses (FC_w), hybrid facilities (FC_h), and manufacturers (FC_m) were generated from $U[8000, 12000]$ and $U[32000, 55000]$, respectively. Customer demand for each product p at retailer c (D_{pct}) was drawn from a discrete uniform distribution $DU[400, 600]$. The total demand value (D_TOT) was then computed as the sum of D_{pct} across all products and customers and used as the Big-number (MB) for linearizing the model constraints, as well as for manufacturing capacities (MC_{pm}). Storage capacities for warehouses, hybrid facilities, and manufacturers (SC_w, SC_h, SC_m) were subsequently generated from the interval $U[D\_TOT * 0.25, D\_TOT * 0.75]$. The standard deviation of demand for each product (SDS_p) was derived from the variability of total demand, while lead times (L_p) were drawn from $U[0.6, 1] * 4$.

Sustainability and social responsibility aspects were incorporated through obnoxiousness and proximity parameters. Euclidean distances from urban centers to a reference midpoint (DUC_U_u) and to hybrid facilities (DUC_{uh}) were calculated, along with the total and absolute deviation measures used to derive the population-weighted obnoxiousness factor (POP_u) and the average surrounding obnoxiousness score (SR). Additional technical parameters were set as follows: inventory coefficients $IC_{im} = IC_{pw} = IC_{ph} = 3$, product weight $W_p = 1$, off-take factor $OFF_{ip} = 0.5$, service level $SL = 1.64$ (corresponding to 95% under normal distribution assumptions), and a small linearization constant $e \approx 1$.

Table 3.4: Dataset description

Inst	S	M	W	H	C	P	I	T	V	K	U	A_{pct}	B_p	$\#N$	$\#P$
INS_1	5	3	5	5	10	3	3	5	2	3	6	20	2	28	3750
INS_2	5	3	5	5	10	3	3	5	2	3	6	50	2	28	3750
INS_3	8	3	5	5	15	3	4	5	2	3	6	20	4	36	9000
INS_4	8	3	5	5	15	3	4	5	2	3	6	50	4	36	9000
INS_5	10	3	10	5	25	3	5	5	2	3	6	20	2	53	37500
INS_6	10	3	10	5	25	3	5	5	2	3	6	50	2	53	37500
INS_7	20	3	10	5	30	3	10	5	2	3	6	20	4	68	90000
INS_8	20	3	10	5	30	3	10	5	2	3	6	50	4	68	90000
INS_9	10	3	15	10	40	3	5	5	2	3	6	20	2	78	180000
INS_10	10	3	15	10	40	3	5	5	2	3	6	50	2	78	180000
INS_11	20	3	15	10	45	3	10	5	2	3	6	20	4	93	405000
INS_12	20	3	15	10	45	3	10	5	2	3	6	50	4	93	405000
INS_13	20	3	18	12	50	3	10	5	2	3	6	20	4	103	648000
INS_14	25	3	20	15	55	3	15	5	2	3	6	20	4	118	1237500

This systematic parameter generation procedure produced realistic and diverse test instances while preserving the structural characteristics and relative cost–demand ratios of the original network presented by Pazhani et al. (2021).

Table 3.6 presents the values of the parameters related to the three sizes of both internal combustion (IC) and electric vehicles (EV). It shows the CO_2 emission factors, CO_{vk}, vehicle cost VC_{vk} and vehicle capacity VQ_{vk}. The factors presented assume an emissions ratio of 3:1 and a cost ratio of 1:2 for IC and EVs, respectively (Puget Sound Energy, 2024).

The datasets of this computational experience will be available at https://github.com/EliasOBoptlog/KS_MOSCLSCN (accessed on February 3rd, 2026).

3.3.6 Performance metrics

To assess the effectiveness of the solution methods, the values of non-dominated solutions (NPS), computational time (CPU time), quality metric (QM), the ratio of Pareto-optimal solutions (RPOS), the distribution and spacing metrics of the resulting Pareto front, the mean ideal distance (MID), and the spacing metric (SM), and the hypervolume indicator (HV) are used. (Audet et al., 2021; Zitzler et al., 2003). The following notations are used for the performance metrics: $i = \{1..\textit{Pareto front}\}$, $j = \{1..\textit{Solutions}\}$, PF_i = Non-dominated evaluated Pareto front i, and PFC = Non-dominated combined Pareto front.

RPOS: The ratio of Pareto-optimal solutions is computed between the non-dominated evaluated Pareto front obtained from the non-dominated combined Pareto front and the original non-dominated evaluated Pareto front. A higher ratio is preferable, as it indicates better performance (Altiparmak et al., 2006).

$$RPOS\ (PF_i) = \frac{|PF_i \bigcap PFC|}{|PF_i|}$$

QM: This quality metric evaluates the relationship between the evaluated non-dominated Pareto front and the combined non-dominated Pareto fronts. A higher

Table 3.5: Dataset settings

Parameters	Values range
N_coord_x	Continuous uniform distribution [0,450]
N_coord_y	Continuous uniform distribution [0,300]
TC_{ism}	Continuous uniform distribution [1250,1500]
TC_{mw}	Euclidean distance from manufacturer m to warehouse w
TC_{mh}	Euclidean distance from manufacturer m to hybrid facility h
TC_{hm}	Euclidean distance from hybrid facility h to manufacturer m
TC_{hc}	Euclidean distance from hybrid facility h to retailer c
TC_{wc}	Euclidean distance from warehouse w to retailer c
TC_{ch}	Euclidean distance from retailer c to hybrid facility h
PC_{pm}	Continuous uniform distribution [100,110]
RC_{pm}	Continuous uniform distribution [75,90]
D_{pct}	Discrete uniform distribution [400,600]
D_TOT MC_{pm} MB	Total sum of D_{pct}
FC_w FC_h	Continuous uniform distribution [8000,12000]
FC_m	Continuous uniform distribution [32000,55000]
SC_w SC_h SC_m	Continuous uniform distribution [D_TOT * 0.25, D_TOT * 0.75]
SDS_p	Standard deviation of D_TOT of product p
L_p	Lead time of product p continuous uniform distribution [0.6,1] * 4
e	Number near 1 for a linearized equation
DUC_U_u	Euclidean distance from urban centers u to the midpoint
DUC_U_TOT	Total sum of DUC_U_u
POP_u	ABS (DUC_U_TOT - DUC_U_u)
DUC_{uh}	Euclidean distance from urban centers u to hybrid facilities h
SR	Average of total sum of DUC_{uh} of obnoxiousness surrounding
IC_{im}	3
IC_{pw}	3
IC_{ph}	3
W_p	1
OFF_{ip}	0.5
SL	1.64 (95%)

Table 3.6: Vehicle parameters

Parameter	**1**	**2**	**3**	*Vehicle type*
CO_{vk}	0.0003	0.001	0.002	EV
	0.001	0.003	0.006	IC
VC_{vk}	100	200	1000	EV
	50	100	500	IC
VQ_{vk}	10	20	50	EV
	10	20	50	IC

value of this ratio is considered more favorable (Rayat et al., 2017).

$$QM\ (PF_i) = \frac{|PF_i \bigcap PFC|}{|PFC|}$$

MID: The mean ideal distance measures the average Euclidean distance between a solution S_j of the Pareto front and the ideal point in the objective space. The smaller the value, the better.

$$MID\ (PF_i) = \frac{\sqrt{\sum_{j=1}^{n} \left(\frac{S_j - Best\ S_j}{Max\ S_j - Min\ S_j} \right)^2}}{n}$$

SM: The spacing metric assesses the distribution of Pareto-optimal solutions in the objective space by calculating the average difference between the Euclidean distance d_j of adjacent solutions and the mean Euclidean distance $\bar{d}$. A lower value indicates a better distribution of solutions.

$$SM\ (PF_i) = \frac{\sum_{j=1}^{n-1} \left| d_j - \bar{d} \right|}{(n-1)\bar{d}}$$

HV: The hypervolume indicator measures the volume of the objective space covered by the Pareto front PF_i, bounded by a reference point $r = max(F_i) \in \mathbb{R}^m$. It is defined so that for every point $y \in PF_i$, $y \leq r$, where β_m denotes the Lebesgue m-dimensional measure (with $m = 3$). Normalizing the hypervolume value by the cube's volume facilitates comparing HV values across different HV ranges. A higher value is preferred.

$$HV\ (PF_i; r) = \beta_m \left(\bigcup_{y \in PF_i} [y, r] \right)$$

The solution approaches were coded and implemented in Python 3.11.7 (packaged by Anaconda, Inc.) using JupyterLab Version 4.0.11, and solved using Gurobi Optimizer version 11.0.3 via amplpy (Fourer et al., 2003; Optimization, 2021). The computational equipment used was a PC with a 13th Gen Intel(R) Core(TM) i9-13900 2.00 GHz processor and 32 GB of RAM, running Windows 11 Pro. A maximum time limit of 7200 seconds per iteration was set for all experimentation unless otherwise stated. The following section presents the analysis results, the computational experience, and the performance of the methods.

Bibliography

Altiparmak, F., Gen, M., Lin, L., and Paksoy, T. (2006). A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & Industrial Engineering*, 51(1):196–215.

Archetti, C., Guastaroba, G., Huerta-Muñoz, D. L., and Speranza, M. G. (2021). A kernel search heuristic for the multivehicle inventory routing problem. *International Transactions in Operational Research*, 28:2984–3013.

Audet, C., Bigeon, J., Cartier, D., Digabel, S. L., and Salomon, L. (2021). Performance indicators in multiobjective optimization. *European Journal of Operational Research*, 292:397–422.

Blank, J. and Deb, K. (2020). Pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509.

Borčinová, Z. (2022). Kernel search for the capacitated vehicle routing problem. *Applied Sciences 2022, Vol. 12, Page 11421*, 12:11421.

Chicano, F., Sutton, A. M., Whitley, L. D., and Alba, E. (2015). Fitness probability distribution of bit-flip mutation. *Evolutionary computation*, 23(2):217–248.

Filippi, C., Guastaroba, G., Huerta-Muñoz, D. L., and Speranza, M. G. (2021). A kernel search heuristic for a fair facility location problem. *Computers and Operations Research*, 132.

Fourer, R., Gay, D. M., and Kernighan, B. W. (2003). *AMPL: A Modeling Language For Mathematical Programming*. Duxbury Press, Belmont, CA, 2nd edition. Online at https://ampl.com/resources/the-ampl-book/.

Kaya, M. (2011). The effects of two new crossover operators on genetic algorithm performance. *Applied Soft Computing*, 11(1):881–890.

Keller, A. A. (2019). *Multi-objective optimization in theory and practice II: metaheuristic algorithms*. Bentham Science Publishers.

Maniezzo, V., Boschetti, M. A., and Stützle, T. (2021). *Matheuristics*. Springer International Publishing.

Mavrotas, G. and Florios, K. (2013). An improved version of the augmented ϵ-constraint method (augmecon2) for finding the exact pareto set in multi-objective integer programming problems. *Applied Mathematics and Computation*, 219:9652–9669.

Optimization, A. I. (2021). amplpy: Python API for AMPL. https://github.com/ampl/amplpy.

Pazhani, S., Mendoza, A., Nambirajan, R., Narendran, T. T., Ganesh, K., and Olivares-Benitez, E. (2021). Multi-period multi-product closed loop supply chain network design: A relaxation approach. *Computers and Industrial Engineering*, 155:107191.

Puget Sound Energy (2024). Compare co2 emissions for gas and electric cars. https://pse.chooseev.com/carbon/. Online; accessed february 7th 2026.

Rayat, F., Musavi, M., and Bozorgi-Amiri, A. (2017). Bi-objective reliable location-inventory-routing problem with partial backordering under disruption risks: A modified amosa approach. *Applied Soft Computing*, 59:622–643.

Zhang, Y., Che, A., and Chu, F. (2022). Improved model and efficient method for bi-objective closed-loop food supply chain problem with returnable transport items. *International Journal of Production Research*, 60:1051–1068.

Zhang, Y., Chu, F., Che, A., and Li, Y. (2024). Closed-loop inventory routing problem for perishable food with returnable transport items selection. *International Journal of Production Research*, 62:501–521.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and Fonseca, V. G. D. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7:117–132.

Chapter 4

Results and discussion

This chapter examines the model's performance in single- and multi-objective settings. The outcome comparison of the AUG2-based solution method is discussed. The detailed results of the kernel search matheuristic are then presented. The results of the hybrid NSGAII are also presented. And the comparison between the solution methods.

4.1 Model optimization performance

The model's behavior is analyzed based on the outcome in the lexicographic chart, where each objective function is minimized while the others are left unrestricted. Figure 4.1, 4.2, and 4.3 illustrate the solution behavior of the objective function optimized alone for the instance INS_1, where the solid line icons are selected, and the dotted line icons are not, showing the linkages between each selected facility. The supplier is represented by a circle, the manufacturer by a square, the warehouse by a triangle, the hybrid facility by a diamond, the retailer by a cross, the urban center by a small plus sign, and the midpoint of the location space by a large plus sign.

4.1.1 Single objective optimization

When the economic objective F_1 is minimized, the solution uses internal combustion vehicles and discards a supplier, two warehouses, and two hybrid facilities

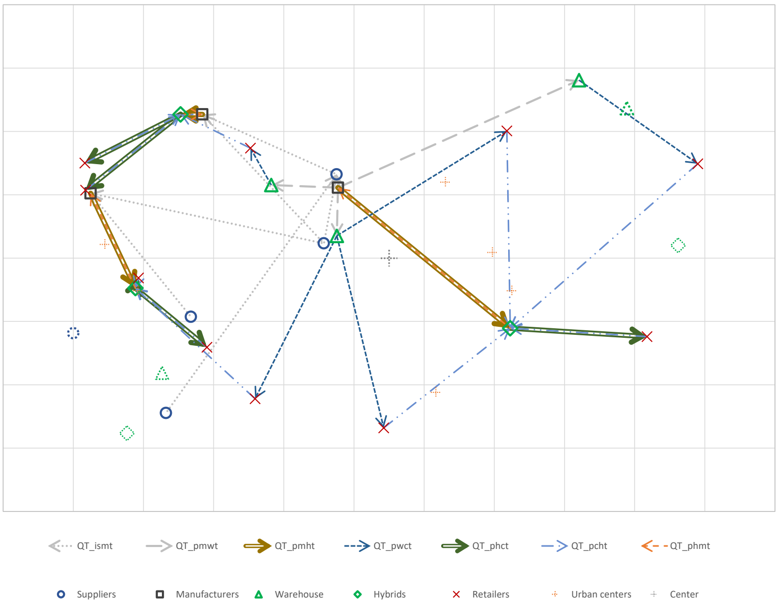


Figure 4.1: $F1_{min}$ solution of INS_1

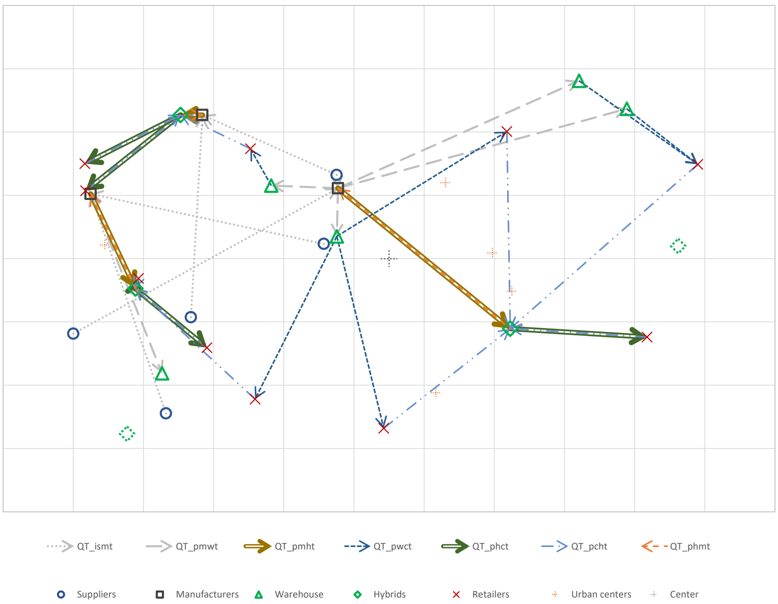


Figure 4.2: $F2_{min}$ solution of INS_1

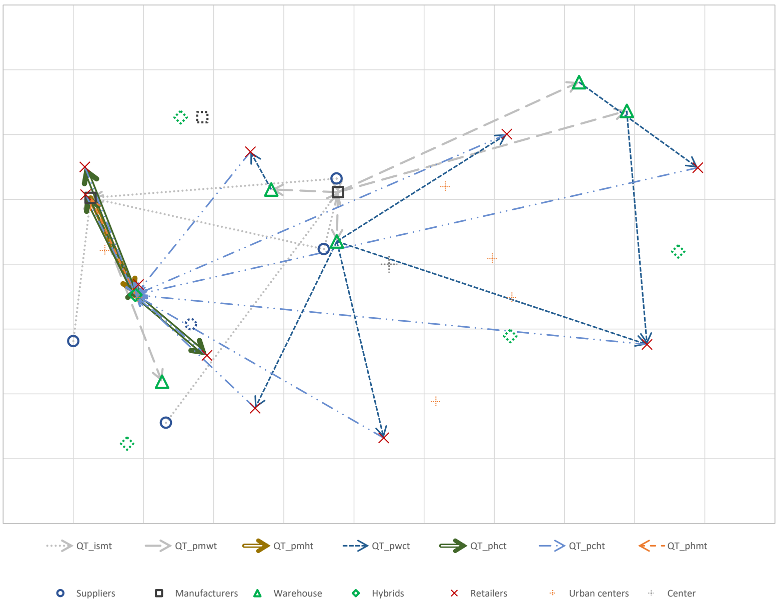


Figure 4.3: $F3_{min}$ solution of INS_1

(see Figure4.1). When the CO_2 emission function F_2 is minimized, the solution selects electric vehicles and all facilities except two hybrid facilities, minimizing both distance and CO_2 emissions (see Figure4.2). When the obnoxious function F_3 is minimized, the solution selects a single hybrid facility (see Figure4.3), thereby reducing the impact of obnoxiousness. The decision-maker can then evaluate and choose from the Pareto front solutions obtained.

For the INS_1 instance (see Figure4.4), the minimum $F1_{min}$ (a-dot 1) yields an increase in F_2 of 3.4 times $F2_{min}$ and in F_3 of 3.6 times $F3_{min}$. At $F2_{min}$ (a-dot 2), F_1 matches $F1_{min}$, while F_3 increases by 3.9 times $F3_{min}$. Finally, at $F3_{min}$ (a-dot 3), F_1 increases by 1.4 times $F1_{min}$ and F_2 by 1.5 times $F2_{min}$. Similarly, for the INS_14 instance (see Figure4.5), $F1_{min}$ (b-dot 1) yields an increase in F_2 of 3.5 times $F2_{min}$ and in F_3 of 5.1 times $F3_{min}$. At $F2_{min}$ (b-dot 2), F_1 matches $F1_{min}$, while F_3 increases by 5.1 times $F3_{min}$. Finally, at $F3_{min}$ (b-dot 3), F_1 increases by 1.9 times $F1_{min}$ and F_2 by 2.4 times $F2_{min}$. In summary, when F_1 is minimized, it increments more than 3 times F_2 and F_3; when F_2 is minimized, F_3 increments more than 4 times while F_1 remains unchanged because of the fleet selection, and when F_3 is minimized, it increments around 2 times F_1 and F_2.

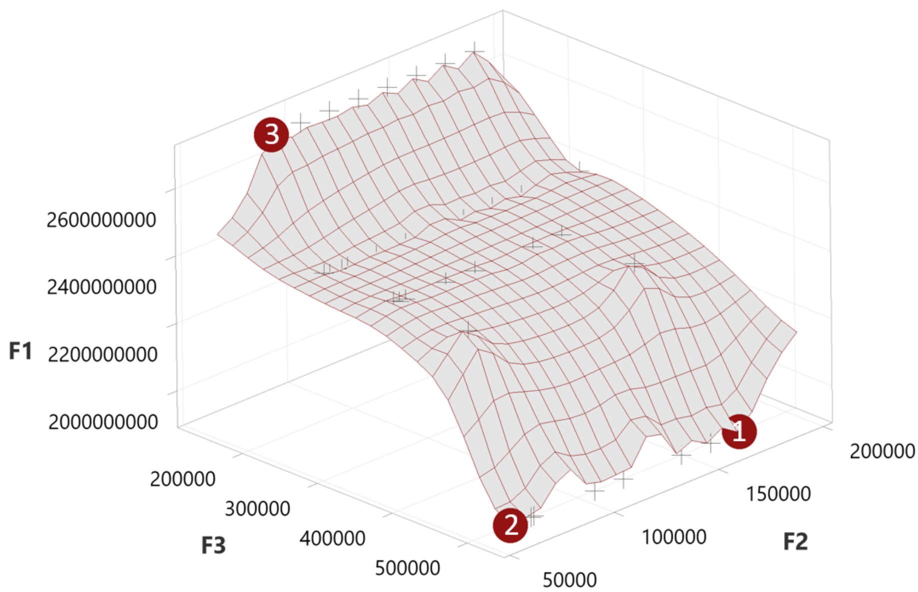


Figure 4.4: Objective function minimum points of INS_1

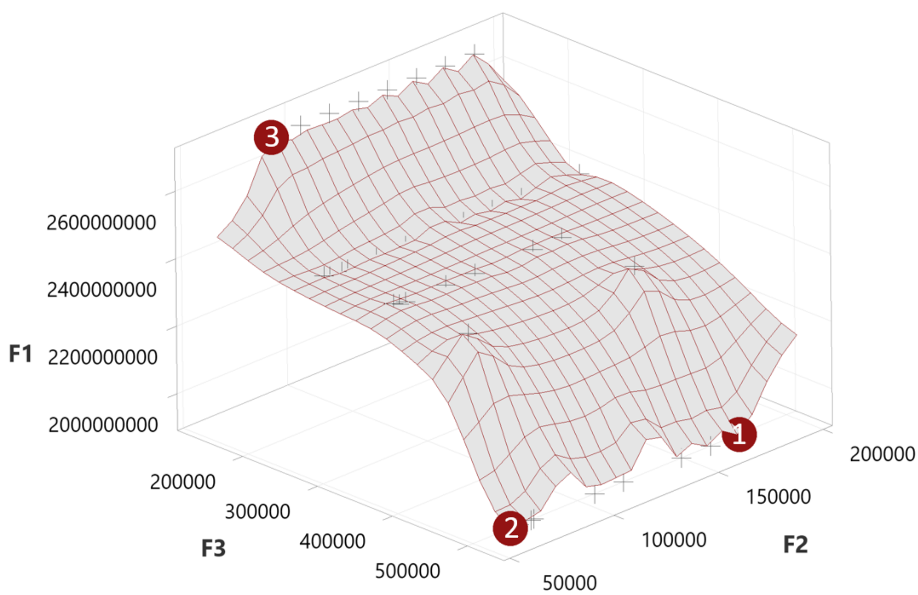


Figure 4.5: Objective function minimum points of INS_14

4.1.2 Sensitivity analysis for the multi-objective optimization

To better understand the model's response behavior, a sensitivity analysis was performed using ANOVA, comparing the Hypervolume as the output variable, similar to Mar-Ortiz et al. (2022) and Sbayti et al. (2018). A factorial design with two levels was used to test the effects of the selected factors (see Table 4.1). The selected factors are the retailer demand (DEM_{pct}) labeled as DEM, the vehicle's capacity (VQ_{vk}) labeled as CAP, and the facilities' fixed cost (FCW_w, FCH_h, FCM_m) labeled as FAC, each at a low and high level (i.e. a 2^k, $k = 3$). The experimental design was generated in Minitab. The selected instances are INS_1, INS_8, and INS_14, chosen as small, medium, and large-sized instances.

Table 4.1: Factors levels model performance

Factors	Low (-)	High (+)
Retailer demand (DEM)	-20%	20%
Vehicle's capacity (CAP)	20%	40%
Facilities' fixed cost (FAC)	-20%	20%

Since this work considers a multi-objective problem, the response of the factorial design considered is the Hypervolume indicator (a higher value is better), calculated using the PYGMO 2.19.5 engine (Biscani and Izzo, 2020) as implemented by Mohammadgholibeyki et al. (2023) and da Silva et al. (2023). The Hypervolume uses the eight Pareto fronts obtained from the factorial design runs for each instance. ANOVA requires three main assumptions (normality, homoscedasticity, and independence). For this work, the response needed to be transformed to meet these assumptions, and Minitab automatically calculated the optimal lambda.

The hypervolume values obtained for each batch are shown in Table 4.2. Both the original response and the Box-Cox (BC) transformation are shown (including the proper lambda to achieve the transformation). To evaluate the responses of the hypervolume indicator in each experiment, an ANOVA was performed to determine which factors are significant, the effects of the factors, the standardized effect charts, and the p-values for each analysis. Table 4.3 shows the effects of three

Table 4.2: HV for the sensitivity analysis

RUN	INS_1		INS_8		INS_14	
	Original	BC ($\lambda = -4$)	Original	BC ($\lambda = 1$)	Original	BC ($\lambda = 0$)
1	0.3076	-0.0003	0.0966	0.0966	0.2570	-1.3587
2	0.3186	-0.0002	0.1020	0.1020	0.2746	-1.2924
3	0.0387	-1.3013	0.0308	0.0308	0.1016	-2.2863
4	0.0387	-1.2983	0.0308	0.0308	0.1016	-2.2864
5	0.2110	-0.0013	0.0966	0.0966	0.2570	-1.3588
6	0.0413	-0.9967	0.0335	0.0335	0.1114	-2.1949
7	0.2220	-0.0011	0.1020	0.1020	0.2746	-1.2926
8	0.0413	-0.9979	0.0335	0.0335	0.1114	-2.1951

factors, their interactions, and contributions to the model regression, and the p-values per instance. The model regressions for the transformed responses deliver the following values for the standard deviation and R-squared error: INS_1 ($S = 0.0010718, R - sq = 100.00\%$), INS_8 ($S = 0.0000031, R - sq = 99.9999996871\%$), and INS_14 ($S = 0.0001169, R - sq = 100.00\%$).

In the instances studied, the p-values (assuming a confidence level of 95%) allow concluding that retailer demand (DEM), vehicle capacity (CAP), and their interaction (DEM * CAP) have a significant effect on the model, as illustrated in the graphs of Figure 4.6. The facility's fixed cost (FAC) is not significant in any of the three instances of the model. However, it plays an essential role in calculating the overall supply chain cost. In other cases, this cost component may behave differently. The optimization model can explain the effects of the factors. Specifically, the variable and fixed costs are represented in the total economic objective function F_1, which are segmented into purchasing, production, transportation, inventory, facilities, and vehicle costs.

Table 4.4 presents the proportions of economic costs at the extreme points of the Pareto front for each analyzed instance. In INS_1, INS_8, and INS_14, transportation accounts for between 67% and 95% of the economic cost, followed by purchasing costs, which range from 4% to 31% and represent the model's variable costs. Meanwhile, the facilities cost represents less than 0.03% of the economic cost. Also, the DEM contributions range from 96% to 99.59% of the model re-

Table 4.3: ANOVA for HV transformed response

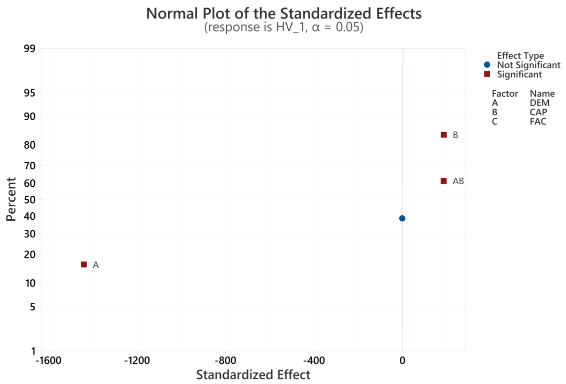
	Source	DF	Seq SS	Contribution	Adj SS	Adj MS	F-Value	P-Value
INS_1	Model	4	2.46	100.00%	2.46	0.62	5.4E+05	4.1E-09
	Linear	3	2.42	98.38%	2.42	0.81	7.0E+05	2.9E-09
	DEM	1	2.38	96.76%	2.38	2.38	2.1E+06	7.4E-10
	CAP	1	0.04	1.62%	0.04	0.04	3.5E+04	3.4E-07
	FAC	1	0.00	0.00%	0.00	0.00	1.9E+00	2.6E-01
	DEM * CAP	1	0.04	1.62%	0.04	0.04	3.5E+04	3.4E-07
	Error	3	0.00	0.00%	0.00	0.00		
	Total	7	2.46	100.00%				
INS_8	Model	4	0.01	100.00%	0.01	0.00	2.4E+08	4.4E-13
	Linear	3	0.01	99.96%	0.01	0.00	3.2E+08	3.0E-13
	DEM	1	0.01	99.59%	0.01	0.01	9.6E+08	7.5E-14
	CAP	1	0.00	0.38%	0.00	0.00	3.6E+06	3.2E-10
	FAC	1	0.00	0.00%	0.00	0.00	5.0E-02	8.4E-01
	DEM * CAP	1	0.00	0.04%	0.00	0.00	3.8E+05	9.4E-09
	Error	3	0.00	0.00%	0.00	0.00		
	Total	7	0.01	100.00%				
INS_14	Model	4	1.69	100.00%	1.69	0.42	3.1E+07	9.5E-12
	Linear	3	1.69	99.98%	1.69	0.56	4.1E+07	6.4E-12
	DEM	1	1.67	99.25%	1.67	1.67	1.2E+08	1.6E-12
	CAP	1	0.01	0.74%	0.01	0.01	9.1E+05	2.6E-09
	FAC	1	0.00	0.00%	0.00	0.00	0.0E+00	9.6E-01
	DEM*CAP	1	0.00	0.02%	0.00	0.00	2.3E+04	6.3E-07
	Error	3	0.00	0.00%	0.00	0.00		
	Total	7	1.69	100.00%				

Table 4.4: Cost proportions of the economic objective function

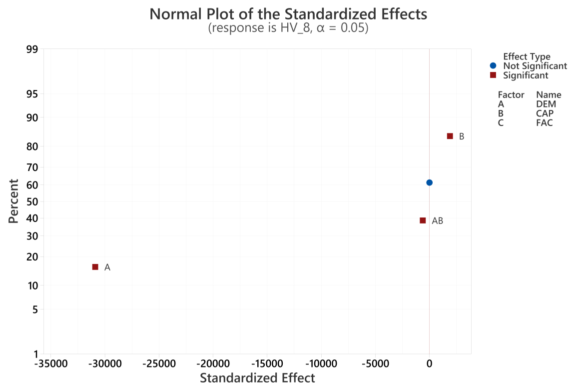
Inst	INS_1		INS_8		INS_14	
	EP min	EP max	EP min	EP max	EP min	EP max
Purchasing	6.0958%	4.6055%	13.1151%	10.9327%	31.6444%	17.7859%
Production	0.3298%	0.2374%	0.2147%	0.1636%	0.3569%	0.1834%
Transportation	93.4663%	95.0699%	86.4734%	88.7654%	67.9037%	81.9708%
Return	0.0623%	0.0454%	0.1626%	0.0976%	0.0652%	0.0341%
Inventory	0.0007%	0.0018%	0.0002%	0.0068%	0.0001%	0.0019%
Facilities	0.0282%	0.0160%	0.0126%	0.0075%	0.0118%	0.0060%
Vehicles	0.0168%	0.0239%	0.0213%	0.0263%	0.0178%	0.0180%

* EP = Extreme point of the Pareto front

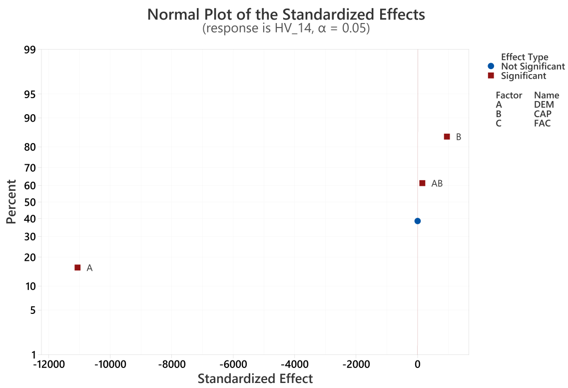
gression shown in Table 4.3, which affects purchasing and transportation costs. It means that the quantity flowing through the supply chain significantly affects the costs and CO_2 emissions, at least in the range studied. The CAP and interaction DEM * CAP contributions range from 0.02% to 1.62% in the model, which explains the correlation between retailer demand and vehicle capacity and its relation to vehicle costs. Finally, the facilities' fixed cost (FAC) factor is not significant.



(a) INS_1



(b) INS_8



(c) INS_14

Figure 4.6: Standardized effects plots for HV (Transformed data)

4.2 AUG2-EXTENDED applications

Pareto fronts from the generated set of instances were used to analyze the performance of the approaches implemented with the AUG2-based variations. As the remaining names indicate, the approaches are as follows: the INT is the AUG2 application (vehicle variables are integers), the PR is the AUG2 application with F_2 relaxed (vehicle variables become continuous), the INTX is the AUG2-EXTENDED application (vehicle variables are integers), and the PRX is the AUG2-EXTENDED application with F_2 relaxed (vehicle variables become continuous). The applications were compared by combining Pareto fronts paired with INT & PR and INTX & PRX.

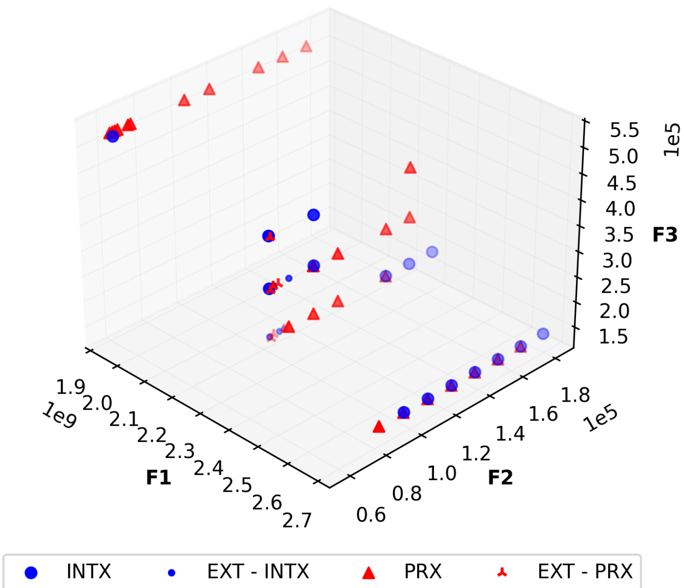


Figure 4.7: INTX & PRX Pareto front INS_1

Table 4.5 displays the NPS and the time of execution of the methods. INT, PR, INTX, and PRX are shown from left to right. Regarding the NPS and CPU time results, INT obtains, on average, one more solution than PR, requiring 15.27% of INT CPU time. Additionally, PRX obtains, on average, 5.57 solutions more than INTX, requiring 41.02% of INTX CPU time.

Table 4.5: NPS and CPU time per AUG2-based applications

Inst	NPS				CPU time			
	INT	PR	INTX	PRX	INT	PR	INT	PRX
INS_1	27	27	42	45	459.34	84.59	663.87	1012.3
INS_2	18	19	22	26	14471.83	35.68	14474.73	45.17
INS_3	20	22	27	29	14506.33	60.9	14795.22	105.94
INS_4	36	33	47	50	21672.58	84.57	21779.49	348.61
INS_5	39	37	46	52	14586.63	176.87	15524.84	477.45
INS_6	39	36	39	52	16456.91	526.69	15023.88	838.76
INS_7	29	29	33	39	15056.64	770.28	15272.19	3816.79
INS_8	48	45	49	54	14684.33	260.21	17945.88	17005.07
INS_9	35	33	47	50	15324.58	894.3	14575.92	27861.11
INS_10	31	36	41	42	15358.89	907.5	16054.31	1113.73
INS_11	45	39	51	47	15831.73	10148.98	36949.31	9770.7
INS_12	52	52	56	68	15218.72	983.43	29878.27	2256.94
INS_13	32	35	35	48	23280.57	3806.68	32408.77	20372.99
INS_14	55	49	61	72	20755.63	14510.61	53227.41	37462.03
Avg	**36.14**	**35.14**	**42.57**	**48.14**	**15547.48**	**2375.09**	**21326.72**	**8749.11**

Table 4.6 shows the non-dominated solutions from the paired INT & PR and INTX & PRX methods, displaying their combined and individual solutions. Furthermore, it can be observed that PR & PRX methods obtain, on average, more non-dominated solutions than INT & INTX, respectively (at least 50%).

The applied method also affects the obtained solutions, so the performance of the approaches was then compared. Figure 4.4 illustrates the surface and cross mark solutions of the Pareto fronts obtained from the PRX method of INS_1 & INS_14, respectively. In addition, they represent the extreme point (red dot) obtained in the lexicographic matrix. Figure 4.7 illustrates a combined non-dominated Pareto front of INTX & PRX methods of INS_1, where the INTX solutions are represented by blue circles and their extended solutions by blue dots, and the PRX solutions are represented by red triangles and their extended solutions by red tri_up. The INTX method yields six solutions, while the PRX method yields 12 solutions in its extended parts. Additionally, PRX is denser than INTX.

Table 4.7 shows the RPOS, QM & HV metric per method, from left to right, INT, PR, INTX, and PRX are displayed. Regarding RPOS, QM & HV results, PR and PRX are the best, on average, obtaining at least 90% of the ratio RPOS, at least 60% of performance QM, and improving 5.7% in HV terms. In terms of

Table 4.6: Combined NPS per AUG2-based applications

Inst	NPS INT & PR			NPS INTX & PRX		
	MIX	INT	PR	MIX	INTX	PRX
INS_1	45	20	26	61	21	41
INS_2	25	9	18	37	13	25
INS_3	33	14	20	48	20	30
INS_4	48	19	32	75	18	38
INS_5	53	19	33	72	18	41
INS_6	45	16	32	62	14	41
INS_7	40	13	29	54	17	39
INS_8	69	32	38	79	22	51
INS_9	40	10	32	74	19	49
INS_10	48	16	34	53	19	37
INS_11	60	26	36	74	30	45
INS_12	80	29	45	77	34	63
INS_13	51	20	35	63	19	48
INS_14	68	29	47	101	39	64
Avg	**50.36**	**19.43**	**32.64**	**66.43**	**21.64**	**43.71**

CPU time, PR requires, on average, 27% of the time that PRX does.

Table 4.8 displays, from left to right, the non-dominated solution obtained in the method with the extended version (INTX & PRX), the number of the non-dominated solutions of the extended part, and the ratio of the extended solutions. The INTX method obtains, on average, 24.4% of non-dominated solutions in the extended segment, whereas the PRX method obtains, on average, 30.4% of the non-dominated solutions from the extended part.

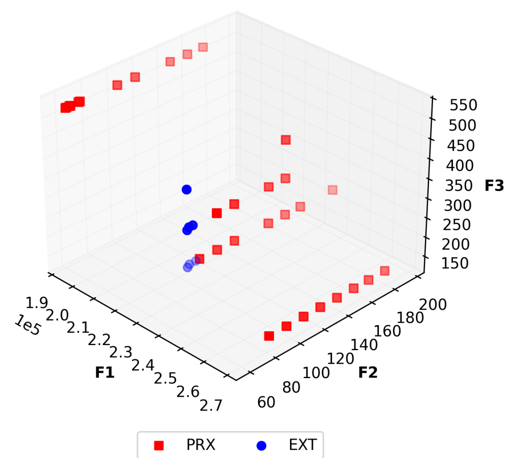
Figure 4.8 illustrates the non-dominated solutions (red squares) and their extended solutions (blue dots) obtained with PRX in the INS_1 and INS_14, obtaining 12 solutions in the extended part of 41 of the total solutions and 15 solutions in the extended part of 64 of the total solutions, respectively. That means the extended part obtains an average range of 20% to 30% of the total solution of the Pareto front. The extended implementation achieves non-dominated solutions because it allows exploration of solutions between the previous feasible iteration and the last infeasible iteration resulting from the lower bound obtained by linearly relaxing the objective function F_2. The PRX method obtains, on average, 12 non-dominated solutions more than the INT method, while requiring, on average, 56% of the CPU time.

Table 4.7: RPOS, QM & HV per AUG2-based applications

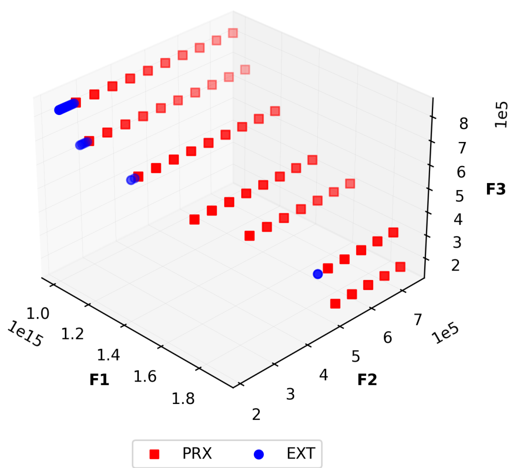
Inst	RPOS				QM				HV			
	INT	PR	INTX	PRX	INT	PR	INTX	PRX	INT	PR	INTX	PRX
INS_1	0.741	0.963	0.500	0.911	0.444	0.578	0.344	0.672	0.035	0.035	0.039	0.039
INS_2	0.500	0.947	0.591	0.962	0.360	0.720	0.351	0.676	0.054	0.055	0.056	0.055
INS_3	0.700	0.909	0.741	1.034	0.424	0.606	0.417	0.625	0.148	0.148	0.162	0.158
INS_4	0.528	0.970	0.383	0.760	0.396	0.667	0.240	0.507	0.119	0.119	0.129	0.130
INS_5	0.487	0.892	0.391	0.788	0.358	0.623	0.250	0.569	0.050	0.050	0.052	0.052
INS_6	0.410	0.889	0.359	0.788	0.356	0.711	0.226	0.661	0.039	0.037	0.039	0.039
INS_7	0.448	1.000	0.515	1.000	0.325	0.725	0.315	0.722	0.079	0.079	0.084	0.083
INS_8	0.667	0.844	0.449	0.944	0.464	0.551	0.278	0.646	0.030	0.030	0.031	0.031
INS_9	0.286	0.970	0.404	0.980	0.250	0.800	0.257	0.662	0.043	0.043	0.045	0.045
INS_10	0.516	0.944	0.463	0.881	0.333	0.708	0.358	0.698	0.046	0.046	0.050	0.050
INS_11	0.578	0.923	0.588	0.957	0.433	0.600	0.405	0.608	0.076	0.076	0.081	0.080
INS_12	0.558	0.865	0.607	0.926	0.363	0.563	0.442	0.818	0.145	0.145	0.151	0.151
INS_13	0.625	1.000	0.543	1.000	0.392	0.686	0.302	0.762	0.015	0.015	0.017	0.017
INS_14	0.527	0.959	0.639	0.889	0.426	0.691	0.386	0.634	0.097	0.097	0.102	0.102
Avg	**0.541**	**0.934**	**0.512**	**0.916**	**0.380**	**0.659**	**0.327**	**0.661**	**0.070**	**0.070**	**0.074**	**0.074**

Table 4.8: NPS ratio of AUG2-EXTENDED

Inst	NPS INTX & PRX			EXT		EXT RATIO	
	MIX	INTX	PRX	INTX	PRX	INTX	PRX
INS_1	61	21	41	6	12	0.286	0.293
INS_2	37	13	25	2	8	0.154	0.320
INS_3	48	20	30	4	8	0.200	0.267
INS_4	75	18	38	13	18	0.722	0.474
INS_5	72	18	41	2	11	0.111	0.268
INS_6	62	14	41	3	13	0.214	0.317
INS_7	54	17	39	2	10	0.118	0.256
INS_8	79	22	51	4	15	0.182	0.294
INS_9	74	19	49	8	22	0.421	0.449
INS_10	53	19	37	4	11	0.211	0.297
INS_11	74	30	45	9	15	0.300	0.333
INS_12	77	34	63	3	14	0.088	0.222
INS_13	63	19	48	3	11	0.158	0.229
INS_14	101	39	64	10	15	0.256	0.234
Avg	**66.43**	**21.64**	**43.71**	**5.21**	**13.07**	**0.244**	**0.304**



(a) INS_1



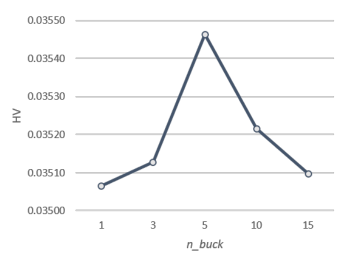
(b) INS_14

Figure 4.8: PRX Pareto fronts

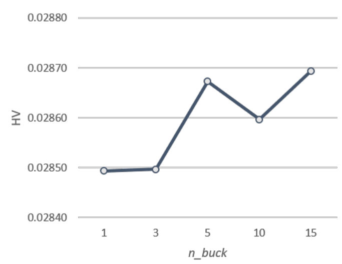
The results demonstrate that PRX outperforms the other variants. On average, it achieves at least 90% of the RPOS ratio, at least 60% of the QM performance, and improves the Hypervolume (HV) by 5.7%. Furthermore, it generates 25% more solutions than the INT application while requiring half of the CPU time.

4.3 KS application

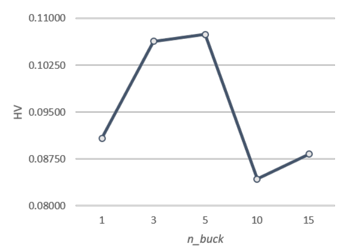
The implementation of the kernel search matheuristic, including the design of experiments for the parameters and results, is described. The parameter inputs needed to apply the kernel search matheuristic are the number of buckets n_buck and the time limits $time\_ks$ ($t_{init}\_t_{loop}$). In the reviewed kernel search matheuristic works, parameter settings were ad hoc, tailored to instance size, problem context, or preliminary experiments. For example, Archetti et al. (2021) stated the number of buckets of 10, Borčinová (2022) explored the number of 2 to 11 buckets, and Zhang et al. (2024) reported the number of 2 and 3 buckets. This research explored 1, 3, 5, 10, and 15 as the number of buckets.



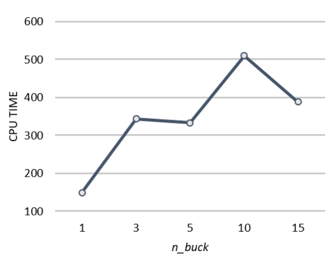
(a) INS_1



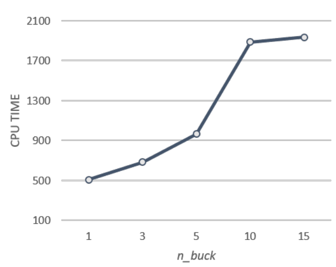
(b) INS_8



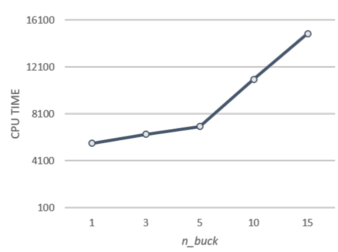
(c) INS_14



(d) INS_1



(e) INS_8

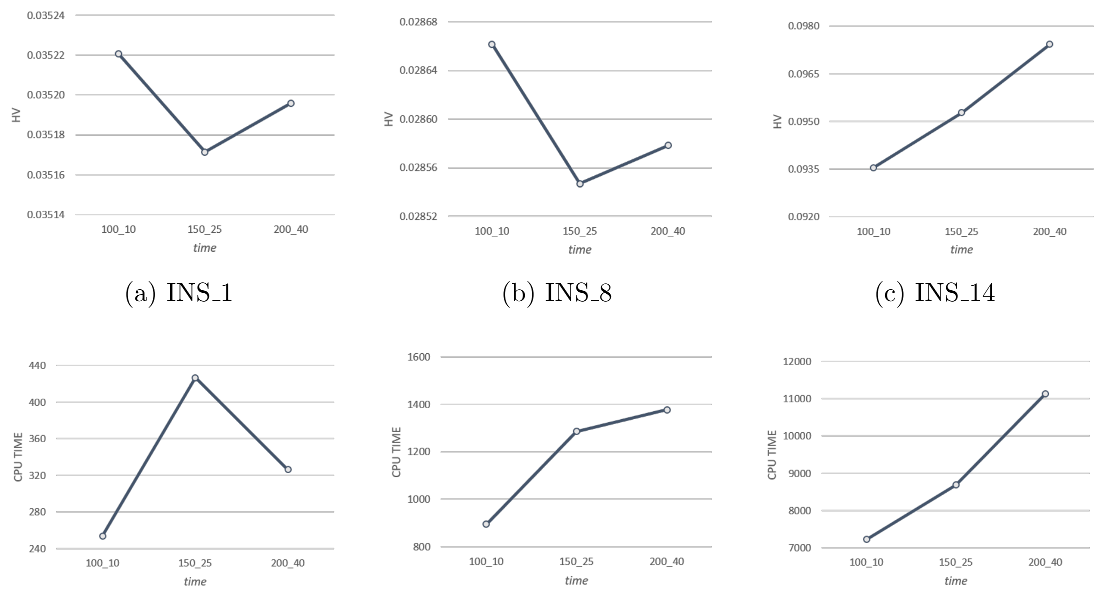


(f) INS_14

Figure 4.9: HV performance and CPU time through n_{buck} parameters

After observing the number of iterations of the AUG2 method, the 7200-second limit is divided by 50 iterations, obtaining an approximate time of 150 seconds. The 150_25 seconds is stated as parameter $time\_ks$ ($t_{init}\_t_{loop}$). To explore other parameters of $time\_ks$, 200_40 and 100_10 seconds were used. These parameters were tested with three representative sizes of instances: a small (INS_1), a medium (INS_8), and a large (INS_14). The comparison used the hypervolume indicator and the CPU time.

Figure 4.9 (a-c) exhibits the HV performance, and (d-f) illustrates the CPU time outcomes for each of the three instance sizes and numbers of buckets. The 5-bucket instance shows higher HV and reports medium CPU Time results compared to the other number of buckets.



(a) INS_1 (b) INS_8 (c) INS_14

(d) INS_1 (e) INS_8 (f) INS_14

Figure 4.10: HV performance and CPU time through $time\_ks$ parameters

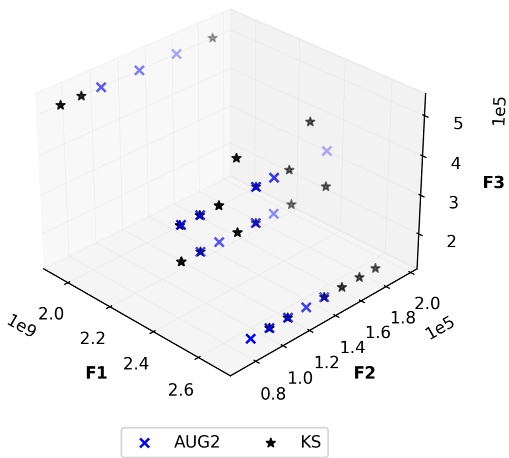
Figure 4.10 (a-c) illustrates the HV performance, and (d-f) depicts the CPU time outcomes in each parameter $time\_ks$. It can observe higher HV values in 100_10 for the small and medium instances, whereas the best performance was achieved with the 200_40 time configuration for the large instance. The experimentation was subsequently performed with $n_{buck} = 5$. For the parameter $time\_ks$, 100_10 was used for INS_1 to INS_8 and 200_40 for INS_9 to INS_14. The latter parameter is consistent with the number of paths $\#P$ of the instances. Instances 1 to 8 explore 3750 to 90000 possible paths, while instances 9 to 14 explore 180000 to 1237500 possible paths. The selection of the *time_ks* parameter is justified due to the two-fold possible number of paths in INS_9. However, the computational experiments were performed exhaustively, and the results are displayed in figures A.1-A.2 in the appendix A. The HV indicators were calculated by comparing the Pareto fronts of the AUG2 and KS applications.

The performance of the kernel search matheuristic (KS) was compared to the results of the AUG2 method (AUG2). Performance metrics compared the Pareto fronts combined in QM, RPOS, and HV. The MID and SM metrics were evaluated independently. Table 4.9 displays the combined non-dominated solutions per method and the CPU time per instance. On average, the KS matheuristic improved the AUG2 method by 72.07% in CPU time while obtaining the same quantity of non-dominated solutions.

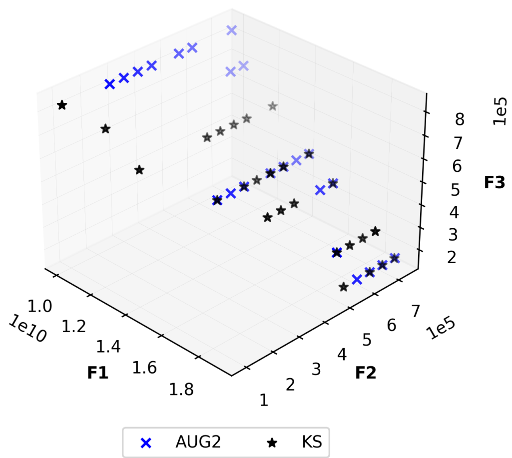
Table 4.9: NPS and CPU time of AUG2 and KS

Inst	NPS		CPU_time	
	AUG2	KS	AUG2	KS
INS_1	30.00	33.00	455	280
INS_2	22.00	22.00	7259	162
INS_3	24.00	25.00	7250	297
INS_4	30.00	24.00	7248	295
INS_5	36.00	39.00	7281	891
INS_6	36.00	39.00	7427	956
INS_7	30.00	33.00	7639	1182
INS_8	45.00	47.00	7286	850
INS_9	36.00	41.00	7442	2001
INS_10	31.00	38.00	7422	1895
INS_11	50.00	51.00	9604	5339
INS_12	50.00	50.00	7633	2821
INS_13	31.00	24.00	13154	6250
INS_14	52.00	33.00	17166	8689
Avg	**35.93**	**35.64**	**8161.75**	**2279.14**

Table 4.10 and 4.11 display the results of performance metrics per method (see Figure 4.11). On average, the KS matheuristic improved over AUG2 by 28.18% on QM, 27.70% on RPOS, 1.24% on MID, 0.47% on SM, and 2.15% on HV.



(a) INS_1



(b) INS_14

Figure 4.11: Non-dominated combined Pareto front of AUG2 & KS

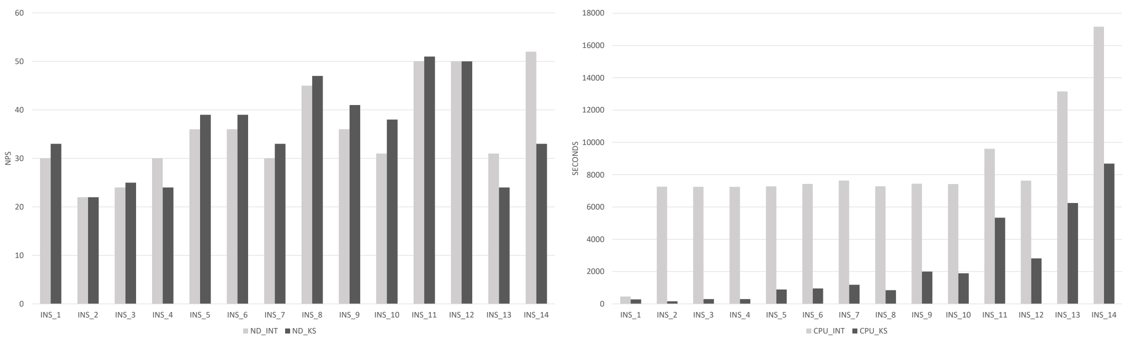
Table 4.10: Cardinality metrics of AUG2 & KS

Inst	NPS AUG2 & KS			QM		RPOS	
	MIX	AUG2	KS	AUG2	KS	AUG2	KS
INS_1	46.00	21.00	28.00	0.4565	0.6087	0.7000	0.8485
INS_2	38.00	18.00	24.00	0.4737	0.6316	0.8182	1.0909
INS_3	37.00	22.00	15.00	0.5946	0.4054	0.9167	0.6000
INS_4	43.00	24.00	23.00	0.5581	0.5349	0.8000	0.9583
INS_5	57.00	20.00	37.00	0.3509	0.6491	0.5556	0.9487
INS_6	64.00	27.00	37.00	0.4219	0.5781	0.7500	0.9487
INS_7	50.00	25.00	27.00	0.5000	0.5400	0.8333	0.8182
INS_8	74.00	31.00	46.00	0.4189	0.6216	0.6889	0.9787
INS_9	57.00	29.00	31.00	0.5088	0.5439	0.8056	0.7561
INS_10	53.00	31.00	25.00	0.5849	0.4717	1.0000	0.6579
INS_11	70.00	24.00	46.00	0.3429	0.6571	0.4800	0.9020
INS_12	75.00	27.00	50.00	0.3600	0.6667	0.5400	1.0000
INS_13	34.00	11.00	23.00	0.3235	0.6765	0.3548	0.9583
INS_14	53.00	27.00	33.00	0.5094	0.6226	0.5192	1.0000
Avg	**53.64**	**24.07**	**31.79**	**0.4574**	**0.5863**	**0.6973**	**0.8905**

To determine a significant difference between AUG2 and KS, the non-parametric Kruskal-Wallis test is applied (see Figure A.3), which uses median values. With a 95% confidence level, there is a significant difference in CPU time, QM, and RPOS; however, there is no significant difference in MID, SM, and HV.

Table 4.11: Diversity and quality metrics of AUG2 & KS

Inst	MID		SM		HV	
	AUG2	KS	AUG2	KS	AUG2	KS
INS_1	1.0092	0.9985	1.8582	1.8696	0.0351	0.0353
INS_2	1.0050	0.9433	1.8088	1.8078	0.0544	0.0545
INS_3	1.0384	0.9896	1.7382	1.7490	0.1478	0.1486
INS_4	0.8890	0.8851	1.8608	1.8323	0.1186	0.1219
INS_5	1.1174	1.0775	1.7704	1.7884	0.0500	0.0501
INS_6	1.0954	1.0565	1.8271	1.8394	0.0370	0.0372
INS_7	0.9993	1.0186	1.8125	1.8252	0.0792	0.0794
INS_8	1.0151	1.0056	1.8156	1.8214	0.0285	0.0286
INS_9	0.9834	0.9941	1.8247	1.8462	0.0423	0.0423
INS_10	1.0870	1.0840	1.7990	1.8363	0.0460	0.0469
INS_11	0.9121	0.9180	1.7937	1.7771	0.0762	0.0772
INS_12	0.9094	0.9112	1.7945	1.7944	0.1449	0.1452
INS_13	1.2059	1.2117	1.7982	1.7360	0.0153	0.0158
INS_14	1.0753	1.0692	1.7624	1.6221	0.0973	0.1104
Avg	**1.0244**	**1.0116**	**1.8046**	**1.7961**	**0.0695**	**0.0710**



(a) NPS

(b) CPU time

Figure 4.12: NPS and CPU time of AUG2 & KS

The NPS results per method (see Figure 4.12(a)) indicate a good performance regarding both algorithms. In particular, it observed that KS accounted for 64.3%

of the instances with more solutions, and AUG2 accounted for 21.4% of the total instances (see Table 4.9). According to the CPU time (see Figure 4.12(b)), the KS matheuristic outperforms the AUG2 method in each instance.

4.4 Hybrid NSGAII application

The implementation of the hybrid NSGAII, including the design of experiments for the parameters and results, is described. For simplicity, the hybrid NSGAII is referred to as HNSGAII. The parameter input needed for the HNSGAII is the population size (*pop_size*), the number of generations (*n_gen*), the time limit (*time_ga*), and the probability for generating a random binary vector (*prob*). To determine the parameters, a fractional factorial design with two levels was used (2^k, $k = 4$) (See Table 4.12). The experimental design was generated using Minitab software; the selected instances are INS_1 and INS_14, and the selected configurations are bold and underlined (See Table 4.13).

Table 4.12: Factors levels for HNSGAII

Factors	Low (-)	High (+)
Time limit (*time_ga*)	25	50
Population size (*pop_size*)	25	50
Number of generations (*n_gen*)	10	20
Probability (*prob*)	25%	50%

Table 4.13: HV for the fractional factorial design for HNSGAII

time_ga	*pop_size*	*n_gen*	*prob*	INS_1	INS_14
+	+	-	-	0.0636	0.0805
-	-	+	+	0.0673	**<u>0.1194</u>**
-	+	+	-	0.0662	0.1163
-	-	-	-	0.0604	0.1143
+	-	+	-	0.0670	0.0772
+	-	-	+	0.0662	0.1047
+	+	+	+	**<u>0.0691</u>**	0.1004
-	+	-	+	0.0673	0.1086

After the preliminary experiment running, it was concluded that the configuration of $time\_ga = 50$, $pop\_size = 50$, $n\_gen = 20$, and $prob = 50\%$ is selected for instances INS_1 to INS_8, and the configuration of $time\_ga = 25$, $pop\_size = 25$, $n\_gen = 20$, and $prob = 50\%$ for INS_9 to INS_14.

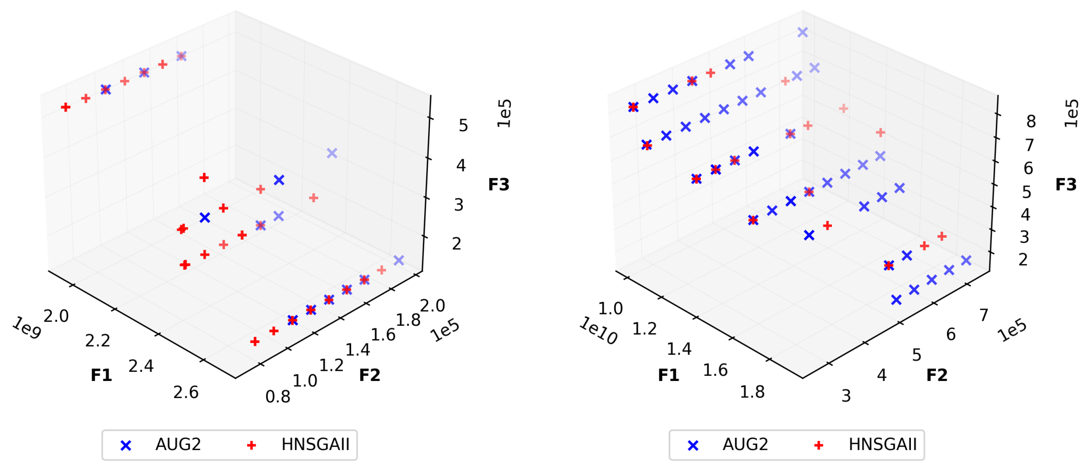
The performance of HNSGAII was compared to the results of the AUG2 method. Performance metrics compared the Pareto fronts combined in QM, RPOS, and HV. The MID and SM metrics were evaluated independently. Table 4.14 displays the combined non-dominated solutions per method and the CPU time per instance. On average, the HNSGAII improved the AUG2 method by 56.90% in CPU time while obtaining a similar quantity of non-dominated solutions.

Table 4.14: NPS and CPU time of AUG2 and HNSGAII

Inst	NPS		CPU time	
	AUG2	HNSGAII	AUG2	HNSGAII
INS_1	30.00	50.00	455.26	2790.92
INS_2	22.00	48.00	7258.60	1184.17
INS_3	24.00	50.00	7249.86	1526.05
INS_4	30.00	50.00	7247.70	1340.54
INS_5	36.00	32.00	7280.93	2042.03
INS_6	36.00	50.00	7426.73	4440.03
INS_7	30.00	34.00	7639.08	3566.52
INS_8	45.00	50.00	7285.80	2838.39
INS_9	36.00	25.00	7442.04	2428.37
INS_10	31.00	25.00	7421.70	2825.83
INS_11	50.00	25.00	9604.20	4476.50
INS_12	50.00	25.00	7632.73	3932.43
INS_13	31.00	25.00	13153.91	7241.79
INS_14	52.00	25.00	17165.95	8613.32
Avg	**35.93**	**36.71**	**8161.75**	**3517.64**

Tables 4.15 and 4.16 display the performance metrics for each method. On average, the HNSGAII performed worse than AUG2 by 15.54% for QM, 15.04% for RPOS, and 5.40% for HV, and better than AUG2 by 3.11% for MID and 8.37% for SM. As illustrated in Figure 4.13, the AUG2 Pareto front is the most densely populated, a finding that is directly corroborated by the cardinality metrics

To determine a significant difference between AUG2 and HNSGAII, a non-parametric test, Kruskal-Wallis, is applied (see Figure A.4). According to the results, with a 95% confidence level, there is a significant difference in CPU time.



(a) INS_1

(b) INS_14

Figure 4.13: Non-dominated combined Pareto front of AUG2 & HNSGAII

Table 4.15: Cardinality metrics of AUG2 & HNSGAII

Inst	NPS AUG2 & HNSGAII			QM		RPOS	
	MIX	AUG2	HNSGAII	AUG2	HNSGAII	AUG2	HNSGAII
INS_1	60.00	15.00	45.00	0.2500	0.7500	0.5000	0.9000
INS_2	59.00	12.00	47.00	0.2034	0.7966	0.5455	0.9792
INS_3	68.00	20.00	48.00	0.2941	0.7059	0.8333	0.9600
INS_4	64.00	25.00	40.00	0.3906	0.6250	0.8333	0.8000
INS_5	49.00	34.00	17.00	0.6939	0.3469	0.9444	0.5312
INS_6	77.00	35.00	42.00	0.4545	0.5455	0.9722	0.8400
INS_7	50.00	28.00	22.00	0.5600	0.4400	0.9333	0.6471
INS_8	57.00	30.00	28.00	0.5263	0.4912	0.6667	0.5600
INS_9	48.00	33.00	15.00	0.6875	0.3125	0.9167	0.6000
INS_10	41.00	30.00	11.00	0.7317	0.2683	0.9677	0.4400
INS_11	63.00	42.00	21.00	0.6667	0.3333	0.8400	0.8400
INS_12	54.00	46.00	8.00	0.8519	0.1481	0.9200	0.3200
INS_13	44.00	28.00	16.00	0.6364	0.3636	0.9032	0.6400
INS_14	63.00	43.00	20.00	0.6825	0.3175	0.8269	0.8000
Avg	**56.93**	**30.07**	**27.14**	**0.5450**	**0.4603**	**0.8288**	**0.7041**

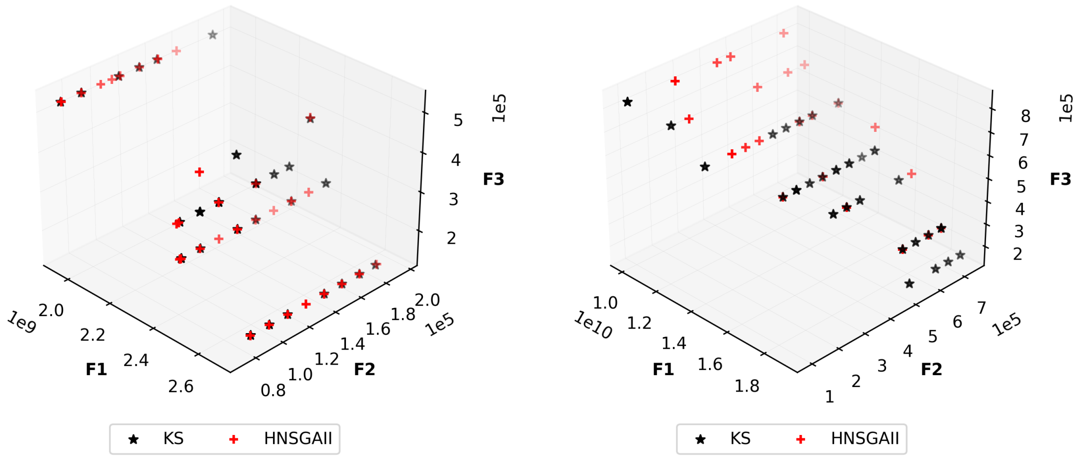
Table 4.16: Diversity and quality metrics of AUG2 & HNSGAII

Inst	MID		SM		HV	
	AUG2	HNSGAII	AUG2	HNSGAII	AUG2	HNSGAII
INS_1	1.0087	0.9858	1.8582	1.9139	0.0351	0.0359
INS_2	1.0056	0.9595	1.8088	1.8706	0.0544	0.0546
INS_3	1.0396	0.8428	1.7382	1.8735	0.1476	0.1482
INS_4	0.8917	0.9624	1.8608	1.9155	0.1187	0.1237
INS_5	1.1179	1.0396	1.7704	1.3301	0.0500	0.0428
INS_6	1.0957	1.0702	1.8271	1.8596	0.0370	0.0375
INS_7	0.9989	0.9450	1.8125	1.7034	0.0794	0.0784
INS_8	1.0100	0.9626	1.8156	1.7383	0.0291	0.0280
INS_9	0.9666	0.9657	1.8247	1.5773	0.0530	0.0493
INS_10	1.0536	1.0554	1.7990	1.3927	0.0601	0.0525
INS_11	0.9108	0.9097	1.7937	1.4250	0.0770	0.0668
INS_12	0.9044	0.8873	1.7945	1.3890	0.1501	0.1293
INS_13	1.1776	1.1626	1.7982	1.5955	0.0222	0.0120
INS_14	1.0194	1.0087	1.7624	1.5630	0.0973	0.0969
Avg	**1.0143**	**0.9827**	**1.8046**	**1.6534**	**0.0722**	**0.0683**

4.5 Comparison of the exact method and the heuristic algorithms

The performance of the KS was compared with that of the HNSGAII. Performance metrics for the combined Pareto fronts from QM, RPOS, and HV were compared across algorithms. MID and SM metrics were assessed independently. Tables 4.17 and 4.18 display the results of performance metrics per method. On average, KS outperformed HNSGAII by 89.71% in CPU time, 30.19% in QM, 33.35% in RPOS, and 7.45% in HV, while performing 1.87% worse in MID and 7.94% worse in SM. Something interesting in terms of solutions obtained HNSGAII found several solutions which KS did not find them (see Figure 4.14). To determine a significant difference between KS and HNSGAII, a non-parametric test, Kruskal-Wallis, is applied (see Figure A.5). With a 95% confidence level, there is a significant difference in CPU time, QM, and RPOS.

The performance of the AUG2, KS, and HNSGAII was compared on the Pareto fronts combined in QM, RPOS, and HV. The MID and SM metrics were evaluated independently. Tables 4.20 and 4.21 display the results of performance metrics per method (see Figure 4.15). On average, in QM, RPOS, and HV terms, KS outper-



(a) INS_1 (b) INS_14

Figure 4.14: Non-dominated combined Pareto front of KS & HNSGAII

Table 4.17: Cardinality metrics of KS & HNSGAII

Inst	NPS KS & HNSGAII			QM		RPOS	
	MIX	KS	HNSGAII	KS	HNSGAII	KS	HNSGAII
INS_1	60.00	19.00	41.00	0.3167	0.6833	0.5758	0.8200
INS_2	56.00	14.00	42.00	0.2500	0.7500	0.6364	0.8750
INS_3	51.00	17.00	34.00	0.3333	0.6667	0.6800	0.6800
INS_4	47.00	22.00	25.00	0.4681	0.5319	0.9167	0.5000
INS_5	48.00	35.00	13.00	0.7292	0.2708	0.8974	0.4062
INS_6	52.00	27.00	25.00	0.5385	0.5769	0.7179	0.6000
INS_7	45.00	29.00	16.00	0.6444	0.3778	0.8788	0.5000
INS_8	69.00	43.00	26.00	0.6232	0.3768	0.9149	0.5200
INS_9	53.00	39.00	14.00	0.7358	0.2642	0.9512	0.5600
INS_10	45.00	36.00	9.00	0.8000	0.2000	0.9474	0.3600
INS_11	61.00	50.00	11.00	0.8197	0.1967	0.9804	0.4800
INS_12	58.00	49.00	9.00	0.8448	0.1552	0.9800	0.3600
INS_13	34.00	18.00	16.00	0.5294	0.4706	0.7500	0.6400
INS_14	37.00	26.00	11.00	0.7027	0.2973	0.7879	0.4400
Avg	**51.14**	**30.29**	**20.86**	**0.5954**	**0.4156**	**0.8296**	**0.5529**

Table 4.18: Diversity and quality metrics of KS & HNSGAII

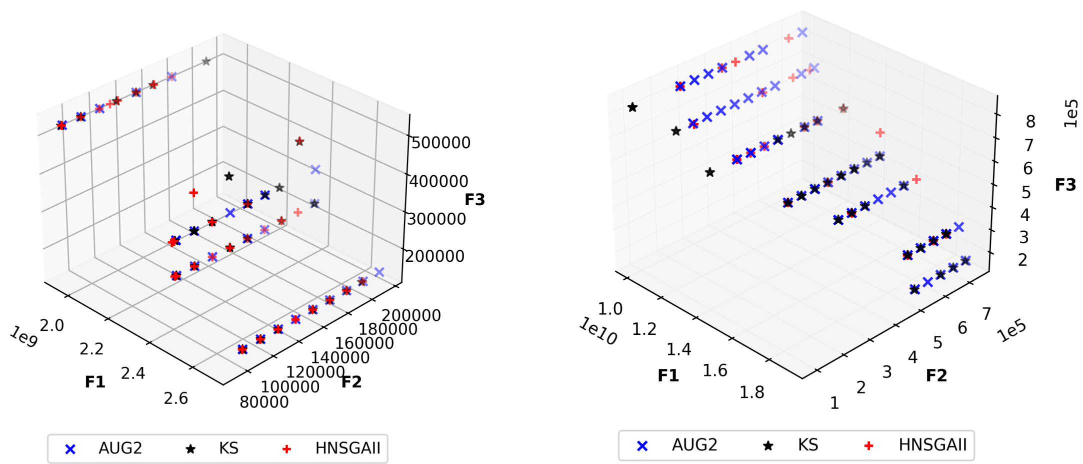
Inst	MID		SM		HV	
	KS	HNSGAII	KS	HNSGAII	KS	HNSGAII
INS_1	0.9987	0.9865	1.8696	1.9139	0.0352	0.0359
INS_2	0.9440	0.9595	1.8078	1.8706	0.0544	0.0546
INS_3	0.9901	0.8422	1.7490	1.8735	0.1486	0.1484
INS_4	0.8878	0.9624	1.8323	1.9155	0.1220	0.1237
INS_5	1.0780	1.0397	1.7884	1.3301	0.0501	0.0428
INS_6	1.0571	1.0704	1.8394	1.8596	0.0372	0.0375
INS_7	1.0182	0.9450	1.8252	1.7034	0.0795	0.0784
INS_8	1.0009	0.9630	1.8214	1.7383	0.0292	0.0280
INS_9	0.9787	0.9649	1.8462	1.5773	0.0533	0.0493
INS_10	1.0452	1.0551	1.8363	1.3927	0.0616	0.0525
INS_11	0.9172	0.9100	1.7771	1.4250	0.0780	0.0667
INS_12	0.9066	0.8876	1.7944	1.3890	0.1504	0.1292
INS_13	1.1818	1.1633	1.7360	1.5955	0.0228	0.0119
INS_14	1.0692	1.0603	1.6221	1.5630	0.1104	0.0969
Avg	**1.0052**	**0.9864**	**1.7961**	**1.6534**	**0.0738**	**0.0683**

forms; however, in MID and SM, HNSGAII achieves better results. To determine a significant difference between AUG2, KS, and HNSGAII, a non-parametric test, Mann-Whitney, is applied (see Figure A.6). At the 95% confidence level, there is a significant difference in CPU time and RPOS; however, there is no significant difference in QM, MID, SM, or HV.

Through various comparisons of the solution approaches, on average, KS outperforms HNSGAII in terms of speed and cardinality metrics, as confirmed by statistical tests. KS demonstrates a balance between exploration and exploitation, as well as computational efficiency due to the small size of its subproblems. In contrast, HNSGAII achieved better diversity and quality on average across the tested instances, though the difference is not statistically significant and therefore cannot be generalized.

Table 4.19: NPS and CPU time of AUG2 & KS & HNSGAII

Inst	NPS			CPU time		
	AUG2	KS	HNSGAII	AUG2	KS	HNSGAII
INS_1	30.00	33.00	50.00	455.26	41.27	2790.92
INS_2	22.00	22.00	48.00	7258.60	22.78	1184.17
INS_3	24.00	25.00	50.00	7249.86	50.93	1526.05
INS_4	30.00	24.00	50.00	7247.70	25.50	1340.54
INS_5	36.00	39.00	32.00	7280.93	191.81	2042.03
INS_6	36.00	39.00	50.00	7426.73	114.55	4440.03
INS_7	30.00	33.00	34.00	7639.08	181.92	3566.52
INS_8	45.00	47.00	50.00	7285.80	78.26	2838.39
INS_9	36.00	41.00	25.00	7442.04	238.15	2428.37
INS_10	31.00	38.00	25.00	7421.70	299.58	2825.83
INS_11	50.00	51.00	25.00	9604.20	1074.46	4476.50
INS_12	50.00	50.00	25.00	7632.73	364.35	3932.43
INS_13	31.00	24.00	25.00	13153.91	898.85	7241.79
INS_14	52.00	33.00	25.00	17165.95	1480.64	8613.32
Avg	35.93	35.64	36.71	8161.75	361.65	3517.64



(a) INS_1

(b) INS_14

Figure 4.15: Non-dominated combined Pareto front of AUG2 & KS & HNSGAII

Table 4.20: Cardinality metrics of AUG2 & KS & HNSGAII

Inst	NPS AUG2 & KS & HNSGAII				QM			RPOS		
	MIX	AUG2	KS	HNSGAII	AUG2	KS	HNSGAII	AUG2	KS	HNSGAII
INS_1	67	12	17	38	0.194	0.269	0.567	0.433	0.546	0.760
INS_2	64	8	14	42	0.141	0.234	0.656	0.409	0.682	0.875
INS_3	67	20	14	33	0.299	0.209	0.493	0.833	0.560	0.660
INS_4	62	17	20	25	0.307	0.339	0.419	0.633	0.875	0.520
INS_5	65	20	34	11	0.308	0.523	0.169	0.556	0.872	0.344
INS_6	75	27	25	23	0.360	0.347	0.373	0.750	0.667	0.560
INS_7	60	22	22	16	0.367	0.367	0.283	0.733	0.667	0.500
INS_8	85	20	42	23	0.259	0.506	0.271	0.489	0.915	0.460
INS_9	69	28	28	13	0.420	0.435	0.188	0.806	0.732	0.520
INS_10	60	28	23	9	0.500	0.400	0.150	0.968	0.632	0.360
INS_11	79	23	45	11	0.291	0.570	0.139	0.460	0.882	0.440
INS_12	80	23	49	8	0.300	0.625	0.100	0.480	1.000	0.320
INS_13	43	10	17	16	0.233	0.395	0.372	0.323	0.708	0.640
INS_14	57	23	25	9	0.474	0.491	0.158	0.519	0.849	0.360
Avg	**66.64**	**20.07**	**26.79**	**19.79**	**0.318**	**0.408**	**0.310**	**0.599**	**0.756**	**0.523**

Table 4.21: Diversity and quality metrics of AUG2 & KS & HNSGAII

Inst	MID			SM			HV		
	AUG2	KS	HNSGAII	AUG2	KS	HNSGAII	AUG2	KS	HNSGAII
INS_1	1.0092	0.9985	0.9863	1.8582	1.8696	1.9139	0.0351	0.0353	0.0359
INS_2	1.0055	0.9439	0.9594	1.8088	1.8078	1.8706	0.0544	0.0545	0.0546
INS_3	1.0389	0.9901	0.8422	1.7382	1.7490	1.8735	0.1478	0.1486	0.1484
INS_4	0.8917	0.8878	0.9624	1.8608	1.8323	1.9155	0.1187	0.1220	0.1237
INS_5	1.1179	1.0780	1.0396	1.7704	1.7884	1.3301	0.0500	0.0501	0.0428
INS_6	1.0957	1.0569	1.0702	1.8271	1.8394	1.8596	0.0370	0.0372	0.0375
INS_7	0.9989	1.0182	0.9450	1.8125	1.8252	1.7034	0.0794	0.0795	0.0784
INS_8	1.0104	1.0009	0.9630	1.8156	1.8214	1.7383	0.0291	0.0292	0.0280
INS_9	0.9657	0.9787	0.9649	1.8247	1.8462	1.5773	0.0531	0.0533	0.0493
INS_10	1.0532	1.0452	1.0551	1.7990	1.8363	1.3927	0.0601	0.0616	0.0525
INS_11	0.9108	0.9166	0.9097	1.7937	1.7771	1.4250	0.0770	0.0781	0.0668
INS_12	0.9045	0.9062	0.8873	1.7945	1.7944	1.3890	0.1501	0.1505	0.1293
INS_13	1.1776	1.1809	1.1626	1.7982	1.7360	1.5955	0.0222	0.0229	0.0120
INS_14	1.0753	1.0692	1.0603	1.7624	1.6221	1.5630	0.0973	0.1104	0.0969
Avg	**1.0182**	**1.0051**	**0.9863**	**1.8046**	**1.7961**	**1.6534**	**0.0722**	**0.0738**	**0.0683**

Bibliography

Archetti, C., Guastaroba, G., Huerta-Muñoz, D. L., and Speranza, M. G. (2021). A kernel search heuristic for the multivehicle inventory routing problem. *International Transactions in Operational Research*, 28:2984–3013.

Biscani, F. and Izzo, D. (2020). A parallel global multiobjective framework for optimization: pagmo. *Journal of Open Source Software*, 5:2338.

Borčinová, Z. (2022). Kernel search for the capacitated vehicle routing problem. *Applied Sciences 2022, Vol. 12, Page 11421*, 12:11421.

da Silva, M. A., de Paula Garcia, R., and Carlo, J. C. (2023). Performance assessment of rbfmopt, nsga2, and mhaco on the thermal and energy optimization of an office building. *Journal of Building Performance Simulation*, 16:366–380.

Mar-Ortiz, J., Torres, A. J. R., and Adenso-Díaz, B. (2022). Scheduling in parallel machines with two objectives: analysis of factors that influence the pareto frontier. *Operational Research*, 22:4585–4605.

Mohammadgholibeyki, N., Nazari, F., Venkatraj, V., Koliou, M., Yan, W., Dixit, M., and Sideris, P. (2023). A decision-making framework for life-cycle energy and seismic loss assessment of buildings. *Structure and Infrastructure Engineering*, 19:875–889.

Sbayti, M., Bahloul, R., BelHadjSalah, H., and Zemzemi, F. (2018). Optimization techniques applied to single point incremental forming process for biomedical application. *International Journal of Advanced Manufacturing Technology*, 95:1789–1804.

Zhang, Y., Chu, F., Che, A., and Li, Y. (2024). Closed-loop inventory routing problem for perishable food with returnable transport items selection. *International Journal of Production Research*, 62:501–521.

Chapter 5

Conclusions

The general objective of this thesis, as stated in chapter 1, was to develop and implement an exact method, a metaheuristic, and a matheuristic for designing a multi-objective sustainable closed-loop supply chain network (MOSCLSCN) with hybrid facilities. The following specific objectives were achieved:

- To review relevant literature on solution methods to solve a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- To develop a mathematical model for the design of a multi-objective sustainable closed-loop supply chain network with hybrid facilities.
- To develop and implement an exact method for optimizing a multi-objective, sustainable closed-loop supply chain network design with hybrid facilities.
- To develop and implement a matheuristic for optimizing a multi-objective sustainable closed-loop supply chain network design with hybrid facilities.
- To develop and implementa metaheuristic for optimizing a multi-objective sustainable closed-loop supply chain network design with hybrid facilities.
- To perform a comparative quality analysis of the solution methods in solving multi-objective sustainable closed-loop supply chain problems.

The answers to the questions that this research project answered are as follows:

- Can an exact method obtain optimal solutions for a multi-objective sustainable closed-loop supply chain network with hybrid facilities in a reasonable time? Yes, in small instances.

- Can a matheuristic obtain high-quality solutions for a multi-objective sustainable closed-loop supply chain network with hybrid facilities in a reasonable time? Yes.
- Can a metaheuristic obtain high-quality solutions for a multi-objective sustainable closed-loop supply chain network with hybrid facilities in a reasonable time? Yes.
- Can one of the proposed approaches consistently perform the others differently in terms of quality of the solutions and computational time? Yes, the kernel search matheuristic.

According to the results of the computational experience, the conclusions are stated as follows:

The proposed exact method delivers feasible and high-quality solutions; however, it does not achieve competitive computational times. The alternative hypothesis H1 was accepted, while the null hypothesis H2 was not rejected:

- Alternative H1: The proposed exact method can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Null H2: The proposed exact method cannot find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.

The proposed matheuristic delivers high-quality, feasible solutions within competitive computational times. Accordingly, the alternative hypotheses H3 and H4 were accepted:

- Alternative H3: The proposed matheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Alternative H4: The proposed matheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.

The proposed metaheuristic produces high-quality, feasible solutions in competitive computational time. Consequently, the alternative hypotheses H5 and H6 were accepted:

- Alternative H5: The proposed metaheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.
- Alternative H6: The proposed metaheuristic can find feasible, high-quality solutions to the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities in a reasonable computational time.

In terms of performance method, there is a significant difference in solution quality and computational efficiency among the exact method, matheuristic, and metaheuristic for the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities. Consequently, the alternative hypotheses H7 was accepted:

- Alternative H7: There is a significant difference in solution quality and computational efficiency among the exact method, matheuristic, and metaheuristic for the multi-objective sustainable closed-loop supply chain network problem with hybrid facilities.

A review of the literature (Chapter 2) reveals that 55% of solution methods for closed-loop supply chain problems are metaheuristics, 28% are exact methods, and 17% are matheuristics. Nevertheless, to the best of our knowledge, no study has yet developed a multi-objective sustainable closed-loop supply chain network using an exact method, the kernel search (KS) matheuristic, or a hybrid non-dominated sorting genetic algorithm II (HNSGAII). The present study addresses this research gap by employing these three approaches.

This work proposed a novel MOSCLSCN based on the revised network design model with hybrid recovery centers. The MOSCLSCN model minimizes the total economic cost F_1, the CO_2 emissions of the vehicles used F_2, and the total population within the negative-impact zone F_3, a novel social factor that accounts for the obnoxiousness of facilities. The closed-loop model also considers potential circularity applications, such as reuse, remanufacturing, refurbishment, and recycling.

This proposal advances United Nations Sustainable Development Goals 12 and 13 by promoting circular-economy practices and reducing emissions. The results show the model's behavior when one objective is optimized, with the others left unrestricted during the lexicographic step. When economic costs F_1 are minimized, the solution tends to use internal combustion vehicles and discard facilities; solutions minimizing total CO_2 emission F_2 tend to use electric vehicles, and solutions minimizing obnoxious impact F_3 tend to reduce the hybrid facility location.

The dataset was generated from the supply chain parameters found in the literature. It was established as the set of instances used to test the model and methods.

The AUG2-EXTENDED approaches were implemented to obtain Pareto fronts. The results show that an AUG2-EXTENDED (PRX version) outperforms the other versions. On average, it achieves at least 90% of the RPOS ratio, at least 60% of the QM performance, and improves HV by 5.7%. Additionally, it achieves, on average, 25% more solutions than the AUG2 method (original application) and requires only 56% of the CPU time.

A sensitivity analysis was developed using three factors to evaluate the effects in a multi-objective model. The hypervolume indicator responses reflect that the factors of retailer demand (DEM), vehicle capacity (CAP), and interaction (DEM * CAP) are significant.

The KS matheuristic has previously been applied only as a binary approach; in this work, it is used with integer variables, demonstrating its effectiveness for both binary and integer variables. Also, to our knowledge, no MOSCLSCN problem has been solved with the KS matheuristic. The results demonstrate that the KS matheuristic outperforms the exact method, achieving average computational time reductions of 72% and 35% relative to HNSGAII, and improvements in cardinality metrics in both cases.

To our knowledge, no MOSCLSCN problem has been solved using HNSGAII, which enables exploration of the metaheuristic's diversity and convergence characteristics and, on average, yields good performance results in MID and SM metrics. In the hierarchy of the solution methods, KS ranks first, HNSGAII ranks second, and AUG2 ranks third. Since the proposed approaches demonstrated their capability to address the problem, this study can serve as a foundation for future research to expand or improve them.

Research on MOSCLSN demonstrates their growing relevance across diverse industries, including tires, electronics, pharmaceuticals, textiles, beverages, wood-plastic composites, and automotive sectors. These industries are increasingly adopting circular practices to comply with environmental regulations, address resource scarcity, meet consumer expectations, and strengthen competitiveness. By simultaneously optimizing the three pillars of sustainability, economic (cost minimization), environmental (emission and energy reduction, resource recovery), and social (minimizing community exposure to undesirable facilities), MOSCLSCN models enable companies to build more balanced, resilient, and circular supply networks that deliver long-term profitability alongside meaningful environmental and social benefits.

In practice, decision-making often requires rapid evaluation of multiple scenarios, sensitivity analyses, and frequent re-optimization under changing conditions, such as demand fluctuations, cost variations, or input uncertainty. While exact methods ensure optimality, their high computational cost frequently renders them unsuitable for time-sensitive or iterative applications. This limitation underscores the value of matheuristics, which effectively integrate mathematical programming and heuristic techniques to deliver high-quality, near-optimal solutions within practical time limits. Such approaches support fast what-if analyses, interactive decision-making, and real-time implementation in operational environments.

For future work, some interesting opportunities include case studies on waste management, material recycling, reusing appliance components, and battery disposal. Other potential areas include exploring alternative objective functions in the model, implementing the AUG2-EXTENDED method in various models, and utilizing matheuristic algorithms. Due to the improved performance, future work will include comparing the KS matheuristic to other mathematical frameworks, such as the weighted sum, the T-Chebyshev, and the Goal programming methods. Additionally, the application supports different configurations for HNSGAII, including various operators and parameters. The problem can also be modified to include additional decisions or configurations, thereby increasing complexity and approximating the reality of the situation. Exploring other objective functions, the stochastic version of the problem, and matheuristic applications.

Finally, portions of the research from this thesis have been previously published through a collaborative effort with Dr. Elías Olivares Benítez and Dr. Samuel Moisés Nucamendi Guillén from Universidad Panamericana, and Dr. Julie

Drzymalski from the College of Engineering, Temple University, in the papers (Rodríguez-Escoto et al., 2025) and (Rodríguez-Escoto et al., 2026) (see Fig. B.1 and B.2). Parts of this thesis were presented at Congreso Sociedad Mexicana de Investigación de Operaciones (CSMIO XI) in 2023 (see Fig. B.3), at Congreso Latino Iberoamericano de investigación Operativa (CLAIO XXII) — Congreso Sociedad Mexicana de Investigación de Operaciones (CSMIO XII) in 2024 (see Fig. B.4), and at Congreso Sociedad Mexicana de Investigación de Operaciones (CSMIO XIII) in 2025 (see Fig. B.5). Also, the doctoral stay completion letter at Temple University (see Fig. B.6) is attached in the appendix chapter B.

Bibliography

Rodríguez-Escoto, J.-N., Nucamendi-Guillén, S., Olivares-Benitez, E., and Drzymalski, J. (2026). Circular economy modeling: A multiobjective closed-loop sustainable supply chain problem solved by kernel search. *Mathematics*, 14(5).

Rodríguez-Escoto, J.-N., Olivares-Benitez, E., Nucamendi-Guillén, S., and Drzymalski, J. (2025). A multi-objective sustainable closed-loop supply chain network problem with hybrid facilities. *International Transactions in Operational Research*, 32(6):3497–3527.

Appendix A

Extended results

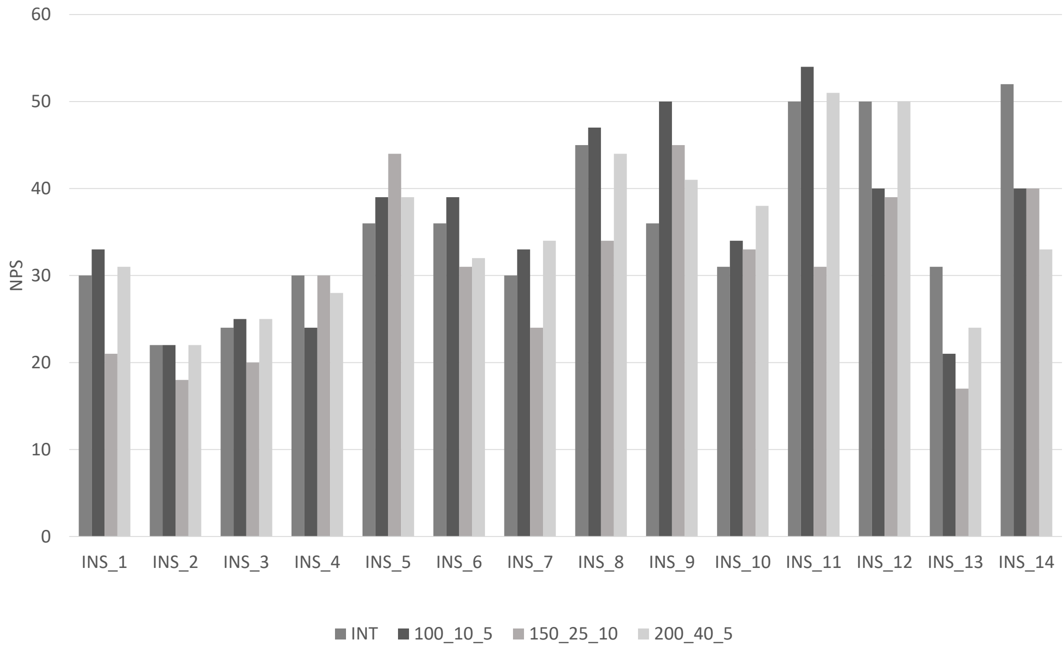


Figure A.1: NPS of kernel search configurations

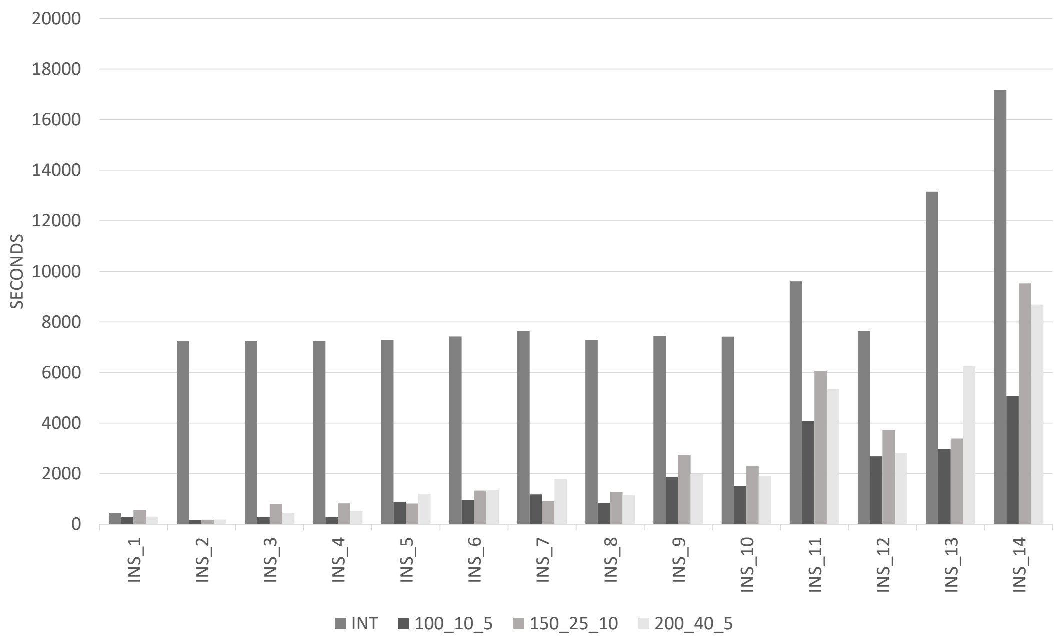


Figure A.2: CPU time of kernel search configurations

Method

η_1: median of CPU_AUG2
η_2: median of CPU_KS
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
CPU_AUG2	14	7424.22
CPU_KS	14	1069.00

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
6404.23	(4605.73, 7126.70)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
281.00	0.00036952

Method

η_1: median of QM_AUG2
η_2: median of QM_KS
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
QM_AUG2	14	0.465103
QM_KS	14	0.615159

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
-0.123874	(-0.201003, -0.0625483)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
133.00	0.001

Method

η_1: median of RPOS_AUG2
η_2: median of RPOS_KS
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
RPOS_AUG2	14	0.725000
RPOS_KS	14	0.948718

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
-0.180271	(-0.311111, -0.0672087)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

Method	W-Value	P-Value
Not adjusted for ties	142.50	0.006
Adjusted for ties	142.50	0.006

Method

η_1: median of MID_AUG2
η_2: median of MID_KS
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
MID_AUG2	14	1.01214
MID_KS	14	1.00207

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.0107931	(-0.0599700, 0.0884805)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
215.00	0.597

Method

η_1: median of SM_AUG2
η_2: median of SM_KS
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
SM_AUG2	14	1.80390
SM_KS	14	1.81459

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
-0.0055073	(-0.0320202, 0.0329887)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
200.00	0.909

Method

η_1: median of HV_AUG2
η_2: median of HV_KS
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
HV_AUG2	14	0.0522205
HV_KS	14	0.0522604

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
-0.0002664	(-0.0315817, 0.0292639)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
196.00	0.765

Figure A.3: Kruskal-Wallis tests of AUG2 & KS

Method

η_1: median of CPU_AUG2
η_2: median of CPU_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
CPU_AUG2	14	7424.22
CPU_HNSGAII	14	2832.11

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
4623.49	(3192.70, 5918.06)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
277.00	0.001

Method

η_1: median of QM_AUG2
η_2: median of QM_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
QM_AUG2	14	0.598182
QM_HNSGAII	14	0.401818

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.0963805	(-0.0695187, 0.291707)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
224.00	0.346

Method

η_1: median of RPOS_AUG2
η_2: median of RPOS_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
RPOS_AUG2	14	0.871613
RPOS_HNSGAII	14	0.723529

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.111667	(-0.0114247, 0.293333)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

Method	W-Value	P-Value
Not adjusted for ties	240.00	0.094
Adjusted for ties	240.00	0.093

Method

η_1: median of MID_AUG2
η_2: median of MID_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
MID_AUG2	14	1.00935
MID_HNSGAII	14	0.96414

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.0400767	(-0.0407476, 0.0946909)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
230.00	0.223

Method

η_1: median of SM_AUG2
η_2: median of SM_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
SM_AUG2	14	1.80390
SM_HNSGAII	14	1.64944

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.156132	(-0.0531173, 0.337363)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
231.00	0.206

Method

η_1: median of HV_AUG2
η_2: median of HV_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
HV_AUG2	14	0.0572514
HV_HNSGAII	14	0.0535412

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.0015443	(-0.0255351, 0.0320783)	95.44%

Test

Null hypothesis	H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis	H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
209.00	0.800

Figure A.4: Kruskal-Wallis tests of AUG2 & HNSGAII

Method
η_1: median of CPU_KS
η_2: median of CPU_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
CPU_KS	14	1069.00
CPU_HNSGAII	14	2832.11

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
-1593.64	(-2663.83, -147.031)	95.44%

Test
Null hypothesis H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
155.00	0.029

Method
η_1: median of QM_KS
η_2: median of QM_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
QM_KS	14	0.633816
QM_HNSGAII	14	0.377295

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.172322	(0.0360360, 0.359037)	95.44%

Test
Null hypothesis H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
250.00	0.033

Method
η_1: median of RPOS_KS
η_2: median of RPOS_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
RPOS_KS	14	0.888112
RPOS_HNSGAII	14	0.510000

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.298718	(0.156364, 0.417436)	95.44%

Test
Null hypothesis H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis H_1: $\eta_1 - \eta_2 \neq 0$

Method	W-Value	P-Value
Not adjusted for ties	283.50	0.00023709
Adjusted for ties	283.50	0.00023569

Method
η_1: median of MID_KS
η_2: median of MID_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
MID_KS	14	0.999806
MID_HNSGAII	14	0.963969

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.0187523	(-0.0522662, 0.0908432)	95.44%

Test
Null hypothesis H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
219.00	0.476

Method
η_1: median of SM_KS
η_2: median of SM_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
SM_KS	14	1.81459
SM_HNSGAII	14	1.64944

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.134505	(-0.0443311, 0.291984)	95.44%

Test
Null hypothesis H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
229.00	0.241

Method
η_1: median of HV_KS
η_2: median of HV_HNSGAII
Difference: $\eta_1 - \eta_2$

Descriptive Statistics

Sample	N	Median
HV_KS	14	0.0580382
HV_HNSGAII	14	0.0535743

Estimation for Difference

Difference	CI for Difference	Achieved Confidence
0.0020029	(-0.0239860, 0.0352841)	95.44%

Test
Null hypothesis H_0: $\eta_1 - \eta_2 = 0$
Alternative hypothesis H_1: $\eta_1 - \eta_2 \neq 0$

W-Value	P-Value
210.00	0.765

Figure A.5: Kruskal-Wallis tests of KS & HNSGAII

CPU time

Descriptive Statistics

Method	N	Median	Mean Rank	Z-Value
AUG2	14	7424.22	32.4	4.06
KS	14	1069.00	12.6	-3.31
HNSGAII	14	2832.11	19.5	-0.75
Overall	42		21.5	

Test
Null hypothesis H_0: All medians are equal
Alternative hypothesis H_1: At least one median is different

Method	DF	H-Value	P-Value
Not adjusted for ties	2	18.64	0.00008984
Adjusted for ties	2	18.64	0.00008977

QM

Descriptive Statistics

Method	N	Median	Mean Rank	Z-Value
AUG2	14	0.303226	19.3	-0.81
KS	14	0.397674	27.1	2.09
HNSGAII	14	0.276961	18.1	-1.28
Overall	42		21.5	

Test
Null hypothesis H_0: All medians are equal
Alternative hypothesis H_1: At least one median is different

Method	DF	H-Value	P-Value
Not adjusted for ties	2	4.46	0.108
Adjusted for ties	2	4.46	0.108

RPOS

Descriptive Statistics

Method	N	Median	Mean Rank	Z-Value
AUG2	14	0.537393	19.3	-0.81
KS	14	0.720020	30.1	3.23
HNSGAII	14	0.510000	15.0	-2.41
Overall	42		21.5	

Test
Null hypothesis H_0: All medians are equal
Alternative hypothesis H_1: At least one median is different

Method	DF	H-Value	P-Value
Not adjusted for ties	2	11.28	0.004
Adjusted for ties	2	11.28	0.004

MID

Descriptive Statistics

Method	N	Median	Mean Rank	Z-Value
AUG2	14	1.00978	24.1	0.96
KS	14	0.99969	21.9	0.13
HNSGAII	14	0.96397	18.6	-1.09
Overall	42		21.5	

Test
Null hypothesis H_0: All medians are equal
Alternative hypothesis H_1: At least one median is different

DF	H-Value	P-Value
2	1.42	0.490

SM

Descriptive Statistics

Method	N	Median	Mean Rank	Z-Value
AUG2	14	1.80390	23.3	0.67
KS	14	1.81459	23.6	0.77
HNSGAII	14	1.64944	17.6	-1.44
Overall	42		21.5	

Test
Null hypothesis H_0: All medians are equal
Alternative hypothesis H_1: At least one median is different

DF	H-Value	P-Value
2	2.08	0.354

HV

Descriptive Statistics

Method	N	Median	Mean Rank	Z-Value
AUG2	14	0.0572866	21.4	-0.03
KS	14	0.0580419	22.5	0.37
HNSGAII	14	0.0535780	20.6	-0.35
Overall	42		21.5	

Test
Null hypothesis H_0: All medians are equal
Alternative hypothesis H_1: At least one median is different

DF	H-Value	P-Value
2	0.17	0.917

Figure A.6: Mann-Whitney tests of AUG2 & KS & HNSGAII

Appendix B

Divulgation outcomes

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 0 (2024) 1–31
DOI: 10.1111/itor.13523

A multi-objective sustainable closed-loop supply chain network problem with hybrid facilities

Joel-Novi Rodríguez-Escoto[a], Elias Olivares-Benitez[a,*], Samuel Nucamendi-Guillén[a] and Julie Drzymalski[b]

[a]*Facultad de Ingeniería, Universidad Panamericana, Álvaro del Portillo 49, Zapopan, Jalisco 45010, Mexico*
[b]*College of Engineering, Temple University, Philadelphia, PA 19122, USA*
E-mail: jorodrigu@up.edu.mx [Rodríguez-Escoto]; eolivaresb@up.edu.mx [Olivares-Benitez]; snucamendi@up.edu.mx [Nucamendi-Guillén]; julie.drzymalski@temple.edu [Drzymalski]

Received 20 November 2023; received in revised form 20 May 2024; accepted 13 July 2024

Abstract

A sustainable closed-loop supply chain network requires conjunctive implementation of reverse logistics in the supply chain, with decisions that consider economic, environmental, and social factors. In real life, the problem needs to be addressed by prioritizing targets or interacting between them to give a range of solutions to the decision maker. In this context, this work proposes a novel multi-objective sustainable closed-loop supply chain network problem based on the revised network design model with hybrid recovery centers minimizing (1) the total economic cost, (2) the CO_2 emission of vehicles used, and (3) the total obnoxious distance. The latter objective is a novel implementation of the social dimension of a sustainable model. A sensitivity analysis of the multi-objective model is developed through ANOVA. A dataset of instances was generated to test the model and the solution methods, which are configured with AUGMECON2, a linear programming relaxation implemented to improve the CPU time, and AUGMECON2-EXTENDED to obtain more solutions to avoid exploring all space of the solution. The results show that an AUGMECON2-EXTENDED implementation outperforms all the selected performance metrics. These performance metrics include NPS, CPU time, RPOS, QM, and HV. The results show an improvement on average of at least 50%, 60%, 90%, 60%, and 5.7%, respectively, in those metrics, in comparison to other implementations.

Keywords: sustainable closed-loop supply chain network; multi-objective optimization; linear programming relaxation; augmented epsilon constraint 2; multi-objective ANOVA; obnoxious facility

Figure B.1: ITOR published article

mathematics

MDPI

Article

Circular Economy Modeling: A Multiobjective Closed-Loop Sustainable Supply Chain Problem Solved by Kernel Search

Joel-Novi Rodríguez-Escoto [1], Samuel Nucamendi-Guillén [1], Elias Olivares-Benitez [1,*] and Julie Drzymalski [2]

[1] Facultad de Ingeniería, Universidad Panamericana, Álvaro del Portillo 49, Zapopan 45010, Jalisco, Mexico; jorodrigu@up.edu.mx (J.-N.R.-E.); snucamendi@up.edu.mx (S.N.-G.)
[2] College of Engineering, Temple University, Philadelphia, PA 19122, USA; julie.drzymalski@temple.edu
* Correspondence: eolivaresb@up.edu.mx

Abstract

The multi-objective sustainable closed-loop supply chain network studied involves characteristics that produce high complexity due to the interaction of downstream and upstream strategic, tactical, and operational decisions, as well as sustainability elements. For this reason, a matheuristic algorithm, the Kernel search, is presented to solve large instances of the problem. After the algorithm parameter tuning, several instances are solved. A comparison with an augmented epsilon-constraint method is conducted in terms of speed and quality. The results show that the Kernel search matheuristic outperforms in the selected metrics, achieving an average improvement of 72% in computational time and from 0.47% to 28.18% in quality metrics. The solutions obtained deliver Pareto fronts in terms of economic, environmental, and social objectives.

Keywords: sustainable closed-loop supply chain network; multi-objective optimization; matheuristic algorithms; kernel search matheuristic

MSC: 90C11; 90C29; 90C59

Figure B.2: Mathematics published article


UDLAP
XI CSMIO
SMIO
SOCIEDAD MEXICANA DE INVESTIGACIÓN DE OPERACIONES

La Sociedad Mexicana de Investigación de Operaciones y
la Universidad de las Américas Puebla
otorga el presente

RECONOCIMIENTO

A:Joel-Novi Rodriguez-Escoto,
Elias Olivares-Benitez, Samuel Nucamendi-Guillén

Por su ponencia en el XI Congreso de la Sociedad Mexicana de Investigación de Operaciones titulada: “Un enfoque multiobjetivo para el diseño de una cadena de suministro sustentable con instalaciones híbridas y consideraciones económicas, ambientales y sociales”
San Andrés Cholula, Puebla, México del 18 al 20 de octubre del 2023

Dra. Marta Cabo Nodar
Presidenta
SMIO

Dr. Omar Ibarra Rojas
Presidente
Comité Científico

Dra. Paulina Avila Torres
Presidenta
Comité Local

Figure B.3: CSMIO XI certificate


CLAIO XXII
CONGRESO LATINO IBEROAMERICANO DE INVESTIGACIÓN OPERATIVA

CSMIO XII
CONGRESO DE LA SOCIEDAD MEXICANA DE INVESTIGACIÓN DE OPERACIONES

THE MEXICAN SOCIETY OF OPERATIONS RESEARCH (SMIO)
PRESENTS THIS

CERTIFICATE OF RECOGNITION
TO:

Joel-Novi Rodríguez-Escoto (Universidad Panamericana); Elias Olivares-Benitez (Universidad Panamericana); Samuel M Nucamendi-Guillén (Universidad Panamericana); Julie Drzymalski (Temple University)*

IN APPRECIATION FOR YOUR VALUABLE CONTRIBUTION TO THE
JOINT ALIO/SMIO CONFERENCE ON OPERATIONS RESEARCH, FOR YOUR TALK TITLED:
ABSTRACT ID: 211, "Optimización multiobjetivo para el diseño de una cadena de suministro circuito cerrado sustentable"

GUADALAJARA, MÉXICO
OCTOBER 28TH – NOVEMBER 1ST, 2024

Dra. M. Angélica Salazar Aguilar
President of the Scientific Committee

Dr. Elias Olivares Benítez
Conference Chair

Dr. Abraham Mendoza Andrade
Conference Chair

Figure B.4: CLAIO XXII & CSMIO XII certificate


CSMIO 2025
XIII Congreso de la Sociedad Mexicana de Investigación de Operaciones
LEÓN

La **Sociedad Mexicana de Investigación de Operaciones** otorga la presente

CONSTANCIA DE PARTICIPACIÓN a:

JOEL-NOVI RODRÍGUEZ-ESCOTO, ELIAS OLIVARES-BENITEZ, SAMUEL NUCAMEDI-GUILLÉN Y JULIE DRZYMALSKI

Por su valiosa contribución al programa del
XIII Congreso de la Sociedad Mexicana de Investigación de Operaciones
con la ponencia titulada

ID 42: UN PROBLEMA MULTIOBJETIVO PARA EL DISEÑO DE UNA CADENA DE SUMINISTRO CIRCUITO CERRADO SOSTENIBLE RESUELTO POR EL ALGORITMO MATHEURÍSTICO DE BÚSQUEDA KERNEL

Dr. Jorge Raúl Pérez Gallardo	Dr. Adrián Ramírez Nafarrate	Dr. Rodolfo Mendoza Gómez
Presidente	Presidente	Presidente
SMIO	Comité Científico	Comité Organizador

SMIO SOCIEDAD MEXICANA DE INVESTIGACIÓN DE OPERACIONES
Tecnológico de Monterrey

León, Guanajuato, a 10 de octubre de 2025.

Figure B.5: CSMIO XIII certificate

College of Engineering
Industrial & Systems Engineering

Philadelphia, USA, July 21, 2024

Dr. Mario Acevedo Alvarado
Research Secretary, Faculty of Engineering
Universidad Panamericana, Campus Guadalajara

This letter certifies that Joel Novi Rodriguez Escoto, student from the PhD in Engineering Program at Universidad Panamericana, did a Research stay at the College of Engineering of Temple University from July 7 to July 20, 2024. During this period, Joel worked under my supervision at the facilities at Temple University for 60 hours in person and 100 hours independently.

The objectives for this stay were:

- discuss the advances on the Ph.D. thesis project of Mr. Rodriguez.
- prepare a response to the review of the paper "A multi-objective sustainable closed-loop supply chain network problem with hybrid facilities" submitted for publication to International Transactions in Operational Research.
- Mr. Rodriguez will develop computational experience on the application of the kernel search matheuristic to the problem we are studying, and
- integrate the results into a draft manuscript for a second publication about his Ph.D. topic.

The results of the stay were:

- Review of preliminary results of the application of the Kernel Search algorithm in the Multi-Objective Closed-Loop Supply Chain Problem.
- A first design of the experiments to adjust the parameters of the algorithm. Review of the ANOVA results of the design of experiments for three instances.
- Design of the experiments for the rest of the instances according to the ANOVA results.
- Development of a draft of the Literature review section for this part of the project with the potential interest in producing a second paper from the research.
- Aacceptance notification of the paper "A multi-objective sustainable closed-loop supply chain network problem with hybrid facilities" submitted for publication to International Transactions in Operational Research.

Sincerely,

Julie Drzymalski, Ph.D.
Professor of Instruction and Director, Industrial & Systems Engineering
Department of Engineering, Technology, and Management
Temple University
Julie.drzymalski@temple.edu

Figure B.6: Doctoral stay letter completion