DOKTORI (PhD) ÉRTEKEZÉS

Rácz-Szabó András

Pannon Egyetem
2024

Ipar 4.0 támogató gépi tanulási, folyamatmodellezési és optimalizációs algoritmusok fejlesztése

Az értekezés doktori (PhD) fokozat elnyerése érdekében készült a Pannon Egyetem Vegyészmérnöki- és Anyagtudományok Doktori Iskolája keretében

Bio-, környezet- és vegyészmérnöki tudományok tudományágban

Írta: Rácz-Szabó András

Témavezetői: Dr. Abonyi János, Dr. Ruppert Tamás

Elfogadásra javaslom (igen / nem)

……………………….
(témavezető)

Elfogadásra javaslom (igen / nem)

……………………….
(témavezető)

Az értekezést bírálóként elfogadásra javaslom:

Bíráló neve: Dr. Földes Dávid igen /nem

……………………….
(bíráló)

Bíráló neve: Dr. Mezőfi Dávid Csaba igen /nem

……………………….
(bíráló)

A jelölt az értekezés nyilvános vitáján ....%-ot ért el.

Veszprém,

……………………….
(a Bíráló Bizottság elnöke)

A doktori (PhD) oklevél minősítése…................................

Veszprém,

……………………….
(az EDHT elnöke)

Development of machine learning, process modeling and optimization algorithms for supporting Industry 4.0

Thesis for obtaining a PhD degree in the Doctoral School of Chemical Engineering and Material Sciences of the University of Pannonia

in the branch of Biochemical, Environmental and Chemical Engineering

Written by András Rácz-Szabó

Supervisors: Dr. János Abonyi, Dr. Tamás Ruppert

propose acceptance (yes / no) ……………………….
(supervisor)

propose acceptance (yes / no) ……………………….
(supervisor)

As reviewer, I propose acceptance of the thesis:

Name of Reviewer: Dr. Dávid Csaba Mezőfi yes / no
……………………….
(reviewer)

Name of Reviewer: Dr. Dávid Földes yes / no
……………………….
(reviewer)

The PhD-candidate has achieved …..........% at the public discussion.

Veszprém, ……………………….
(Chairman of the Committee)

The grade of the PhD Diploma …..................................... (….. %)

Veszprém,

……………………….
(Chairman of UDHC)

PANNON EGYETEM

Doktori (PhD) Értekezés

Ipar 4.0 támogató gépi tanulási, folyamatmodellezési és optimalizációs algoritmusok fejlesztése

DOI:10.18136/PE.2025.916

Szerző:
Rácz-Szabó András

Konzulensek:
Prof. Dr. habil. Abonyi János
Dr. Ruppert Tamás

Értekezés doktori (PhD) fokozat elnyerése érdekében

a ***Pannon Egyetem***

Vegyészmérnöki- és Anyagtudományok
Doktori Iskolájához tartozóan

Folyamatmérnöki Intézeti Tanszék

Pannon Egyetem

2024

UNIVERSITY OF PANNONIA

Doctoral (PhD) Thesis

Development of machine learning, process modeling and optimization algorithms for supporting Industry 4.0

Author:
András Rácz-Szabó

Supervisors:
Prof. Dr. habil. János Abonyi
Dr. Tamás Ruppert

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Doctoral School in Chemical Engineering and Material Sciences
of ***University of Pannonia***

Department of Process Engineering

University of Pannonia

2024

Kivonat

Ipar 4.0 támogató gépi tanulási, folyamatmodellezési és optimalizációs algoritmusok fejlesztése

Rácz-Szabó András

Kivonat

Ez a disszertáció a pozícióadat-alapú optimalizálás alkalmazását vizsgálja, kiemelve azokat az előnyöket, amelyeket a vállalatok nyerhetnek, valamint azokat az értékes betekintéseket, amelyeket adatelemzési módszerekkel lehet kinyerni. A valós idejű helymeghatározó rendszerek (RTLS) adatai lehetővé teszik a gyártócégek számára, hogy jobban megértsék az erőforrások áramlását, az üzemeltetési hiányosságokat és az optimalizálási lehetőségeket, ezáltal hozzájárulva az adatalapú döntéshozatalhoz és a termelékenység növeléséhez.

Az első témakör az RTLS helymeghatározási adatok gyakorlati hasznosságára összpontosít a gyártócégek számára. A szerző kialakított egy strukturális modellt az RTLS technológiák alkalmazásához, amely az integráció lépéseit is tartalmazza a gyártási rendszerekben. Az adatelemzési technikák, mint például az adatbányászat és a statisztikai elemzés alkalmazásával az RTLS rendszerekből származó információk segítenek a belső logisztikai folyamatokban előforduló minták és hatékonysághiányok feltárásában. Ez gyakorlati betekintéseket tesz lehetővé, amelyek javítják az erőforrások elosztását, a munkafolyamatok hatékonyságát és az általános működési teljesítményt.

A második témakör az RTLS adatok pontosságával foglalkozik, bemutatva egy adategyeztetési módszert, amely növeli a helymeghatározási adatok megbízhatóságát. Az adatok pontosságának javítása biztosítja, hogy a folyamatfigyelési és döntéshozatali rendszerek pontosabb információkat kapjanak, ezáltal jobb működési betekintést és hatékonyabb folyamatoptimalizálást tesznek lehetővé.

A harmadik témakör olyan algoritmus fejlesztésére összpontosít, amely a pozícióadatok segítségével azonosítja az értékteremtő és nem értékteremtő tevékenységeket egy többrétegű hálózati megközelítésen keresztül. Az alkalmazás egy taxiútvonal-optimalizálási példán keresztül kerül bemutatásra, miközben az erőforrások áramlása és az optimalizálási lehetőségek többrétegű időbeli hálózati modell segítségével kerülnek elemzésre. A módszer esettanulmányként New York-i taxiadatokon került alkalmazásra, amely során megállapításra került, hogy a módszer használatával

jelentős várakozási időcsökkenés érhető el.

A negyedik témakör olyan módszerek kidolgozására összpontosít, amelyek a logisztikai környezetekben végzett erőforrás-tevékenységeket osztályozzák és optimalizálják beltéri helymeghatározási adatok és fejlett klaszterezési algoritmusok segítségével, ezzel növelve a folyamatok hatékonyságát és támogatva az ellátási lánc elemzését. Az alkalmazás belső logisztikai folyamatok optimalizálására összpontosít egy UWB-alapú helymeghatározó rendszer használatával. Itt egy módosított DB-SCAN algoritmust és egy többrétegű hálózati modellt dolgoztott ki a szerző, amely csökkenti a holtidőket és javítja az erőforrások kihasználását egy raktári környezetben.

Az ötödik témakör az RTLS helymeghatározási adatok felhasználásával végzett algoritmusok fejlesztésére és keretrendszerének kialakítására összpontosít, különösen a mikrologisztika és az anyagkezelési folyamatok területén. Egy olyan keretrendszert kerül kidolgozásra, amely támogatja az automatizált vezetett járművek (AGV-k) ütemezését gyártási környezetekben megerősítéses tanulás és Markov-döntési folyamat (MDP) módszerek alkalmazásával. A módszertant egy valós ipari esettanulmány mutatja be, amely szemlélteti az RTLS helymeghatározási adatokban rejlő potenciált az AGV-k ütemezésének javításában, az erőforrás-kihasználás és a feladatütemezés optimalizálásában, miközben csökkenti a holtidőket és növeli a logisztikai folyamatok hatékonyságát.

Ezeket a megközelítéseket valós gyártási esettanulmányokon validáltam, amelyek bizonyítják azok alkalmazkodóképességét és hatékonyságát az erőforrás-gazdálkodás, a működési hatékonyság és a döntéshozatali folyamatok javításában.

Abstract

Development of machine learning, process modeling and optimization algorithms for supporting Industry 4.0

Andras Racz-Szabo

Abstract

This dissertation explores the use of positioning data to optimize manufacturing processes, focusing on the integration of Real-Time Location System (RTLS) data with advanced data science methodologies. Frameworks and algorithms are presented to demonstrate how positioning data can reveal resource flows, operational inefficiencies, and opportunities for process optimization, enabling data-driven decision making and improving productivity.

The research emphasizes the practical application of RTLS data in manufacturing environments. Methods for improving the accuracy and reliability of positioning data through data reconciliation techniques are explored. Algorithms for classifying value-added and non-value-added activities are highlighted, using innovative multi-layer network approaches and clustering techniques, such as a modified DBSCAN algorithm. These methods are applied to real-world logistics scenarios, including warehouse optimization and taxi route analysis, demonstrating their adaptability and effectiveness in reducing idle time, increasing resource utilization, and improving workflow efficiency.

In addition, the dissertation presents a framework for using RTLS data in micro-logistics and material handling. Reinforcement learning and Markov decision processes are incorporated to optimize the scheduling of Automated Guided Vehicles (AGVs). Industrial case studies validate these methods and demonstrate their effectiveness in overcoming logistical challenges, improving resource management, and increasing operational efficiency in manufacturing systems.

Auszug

Entwicklung von Algorithmen für maschinelles Lernen, Prozessmodellierung und -optimierung zur Unterstützung von Industrie 4.0

Andras Racz-Szabo

Abstract

Diese Dissertation untersucht die Nutzung von Positionsdaten zur Optimierung von Fertigungsprozessen, wobei der Schwerpunkt auf der Integration von Echtzeit-Positionierungssystemen (RTLS) und fortgeschrittenen Methoden der Datenwissenschaft liegt. Es werden Frameworks und Algorithmen vorgestellt, die zeigen, wie Positionsdaten genutzt werden können, um Ressourcennutzung, betriebliche Ineffizienzen und Optimierungsmöglichkeiten aufzudecken und so datenbasierte Entscheidungsprozesse zu unterstützen und die Produktivität zu steigern.

Besonderes Augenmerk wird auf die praktische Anwendung von RTLS-Daten in Produktionsumgebungen gelegt. Methoden zur Verbesserung der Genauigkeit und Zuverlässigkeit von Positionsdaten durch Datenabgleichsverfahren werden untersucht. Algorithmen zur Klassifizierung von wertschöpfenden und nicht wertschöpfenden Aktivitäten werden unter Verwendung von innovativen mehrschichtigen Netzwerkansätzen und Clustering-Techniken, wie z.B. einem modifizierten DBSCAN-Algorithmus, untersucht. Diese Methoden werden auf reale Logistikszenarien angewendet, einschließlich der Optimierung von Lagerprozessen und der Analyse von Taxirouten, und zeigen ihre Anpassungsfähigkeit und Effizienz bei der Reduzierung von Leerlaufzeiten, der Steigerung der Ressourcennutzung und der Verbesserung von Arbeitsabläufen.

Darüber hinaus wird ein Rahmenwerk für die Nutzung von RTLS-Daten in der Mikrologistik und Materialflusssteuerung vorgestellt. Verstärktes Lernen und Markov-Entscheidungsverfahren werden integriert, um die Einsatzplanung von automatisierten Transportfahrzeugen (AGVs) zu optimieren. Industrielle Fallstudien validieren diese Methoden und zeigen ihre Wirksamkeit bei der Bewältigung logistischer Herausforderungen, der Verbesserung des Ressourcenmanagements und der Steigerung der Betriebseffizienz in Produktionssystemen.

Dedicated to my Family and Friends.

Acknowledgements

I am deeply grateful to my supervisors, Prof. Dr. habil. János Abonyi and Dr. Tamás Ruppert, for their support and guidance throughout this journey. Their mentorship has provided me with valuable knowledge and perspectives that will undoubtedly assist me in tackling future challenges.
I would also like to express my heartfelt thanks to my family and friends for their unwavering support and encouragement. Without their constant reassurance and belief in me, this achievement would not have been possible.

Contents

Chapter 1

Introduction

The concept of Industry 4.0 focuses on the digitalisation, automation, and optimisation of production and logistics processes. Industry 4.0 encompasses various technological and digitalisation elements, including digital traceability, which is a fundamental element for the virtual model or representation of our real manufacturing processes. Using virtual model, companies can gain a deeper understanding of their processes, identify bottlenecks, improve productivity and efficiency, and respond more quickly to changes. In many cases, the operation of the system often relies on an indoor positioning system (IPS) [[1], 2, 3], which enables spatial localisation and positioning within manufacturing environments or warehouses.

Several studies address the role of Industry 4.0 technologies, such as indoor positioning systems , in improving manufacturing and logistics processes. These articles often explore specific applications, such as tracking, improving efficiency, or identifying bottlenecks. However, there is a noticeable gap in providing a comprehensive framework that integrates IPS data into the development of algorithms for process optimization and decision support. Furthermore, the literature lacks a detailed examination of how IPS-based algorithms can systematically analyze positioning data to identify inefficiencies, improve resource utilization, and enhance overall operational performance. The first thesis addresses this gap by presenting a methodology that not only improves positioning accuracy, but also emphasizes the development of IPS-based algorithms tailored for process optimization and actionable insights in manufacturing and logistics environments.

However, data provided by IPS frequently contains measurement errors. These errors can be divided into four main groups as the measurement environment, like the indoor environments often contain obstructive elements, such as walls or racks, which can introduce interference [4, 5]. The second one is the computing algorithms, as software algorithms can contribute to inaccuracies, including estimation errors and unfiltered data [6]. At times, IPSs have limited computational resources, impacting processing time and accuracy [7]. The monitoring machines (third) and the human or user interface [8] (fourth) errors are not relevant in our case. Depending on the associated costs, these errors can be mitigated to some extent. However, there comes a point where improving the accuracy of the system exceeds the return on investment. Therefore, alternative solutions should be explored, including external data processing and filtering of the data provided by the system, as well as improving accuracy using various mathematical techniques. Different methods can be used for subsequent algorithm-based refinement of indoor position data. One such method is the Kalman filter, which leverages data from diverse sensors such as accelerometers and gyroscopes to continuously estimate and correct positions over time [9, 10, 11]. Another effective technique is the use of particle filters, which employ probabilistic estimation to handle non-linear and intricate systems, making them particularly suitable for indoor environments [12, 13]. Machine learning algorithms, including neural networks, offer another avenue for enhancing indoor positioning accuracy. These models can be trained on sensor data to discern complex patterns and relationships, thereby refining position estimates [14, 15, 16]. Fingerprinting strategies involve creating databases of signal fingerprints from Bluetooth Low Energy (BLE) beacons or Wi-Fi access points within indoor spaces. Real-time signal strength comparisons against these fingerprints contribute to improved positioning accuracy [17, 18, 19]. Dead reckoning relies on initial known positions and integrates data from sensors like accelerometers and gyroscopes to track movement, making it suitable for short-term indoor positioning needs [20, 21, 22]. Crowdsourced data collection from multiple users' devices can also refine positioning information by amalgamating their measurements. Map matching techniques use pre-existing indoor maps or floorplans to refine indoor positioning accuracy by aligning sensor data with known map features [23]. Visual SLAM methodologies adapt Simultaneous Localisation and Mapping (SLAM) techniques using cameras or depth sensors for indoor positioning [24]. Radio signal propagation models, on the other hand, predict positions based on wireless signal strength and attenuation patterns within buildings. These models are often employed alongside Wi-Fi or Bluetooth signals [25]. Based on the literature review outlined in the previous paragraph, it can be concluded that a simple and efficient approach is needed to achieve increased indoor positioning accuracy when using multiple sensor datasets for object tracking. Ensuring adaptability to variations within the indoor environment is paramount to such a solution. Therefore data reconciliation techniques are also usable for enhancing indoor positioning system accuracy. By delineating physical constraints and measurement errors, data reconciliation ensures the precision of measured position data.

Existing research to improve indoor positioning accuracy has largely focused on isolated techniques or sensor-specific solutions that often lack adaptability to varying indoor environments. While these methods show potential, they often do not address the systematic reconciliation of measurement errors and constraints. The second thesis fills this gap by introducing a data reconciliation approach that systematically improves IPS accuracy by identifying and correcting measurement inconsistencies. By ensuring accurate position data, the methodology supports more robust and reliable applications in dynamic indoor environments.

To address the challenges of indoor positioning accuracy, the methodologies used in urban mobility studies offer valuable insights. Urban mobility research often deals with large-scale position data to analyze patterns, identify inefficiencies, and optimize flows in complex transportation networks. These studies provide a strong foundation for exploring how similar approaches can be applied in industrial contexts, where position data from Indoor Positioning Systems (IPS) is abundant and increasingly critical. The study of urban networks is particularly relevant because it demonstrates how position data can reveal actionable patterns and support resource optimization. The techniques developed for urban scenarios, such as multilayer networks and clustering algorithms, are adaptable to IPS-based industrial applications. Leveraging these methods allows for the analysis of intralogistics and manufacturing processes, enabling the transition of proven solutions from urban environments to dynamic industrial systems. This interdisciplinary connection emphasizes the potential to extend the scope of position-data-driven optimization beyond its traditional domains.

Studying urban networks is essential for the continuous development of sustainable and resilient cities, e.g. public transportation is widely analyzed to explore its efficiency [26]. The mobility patterns of cities have a crucial role in understanding transportation behavior [27]. Exploring frequent flow patterns can identify bottlenecks [28], high-traffic areas dependent on time [29], and popular places [30] as possible starting points of the route [31].

Optimizing traffic flows requires models that represent the frequent patterns of urban network flows [32]. Transport systems are complex and involve a lot of stochastic processes, making them challenging to model. One sources of information for the discovery of mobility patterns is the trajectories of taxis, although these primarily reflect the travel behavior of a specific subset of users, such as higher-income individuals, nighttime travelers, or those using taxis as feeders for long-distance public transportation [33]. An integrated analysis of mobility patterns between taxi and bus mobility records is already proposed [34]. Therefore, this work also focuses on how taxi-related mobilities can be used to evaluate the efficiency of related transport services.

The essential problem of transport systems is the imbalance of demands and resources. The balance of traffic flows in an urban transportation system can be optimized using an appropriate information system [35]. Shared mobility-on-demand solutions also require balancing-based fleet management [36]. For example, bike-sharing systems should be able to handle time-varying needs through the balanced distribution of assets [37]. Balancing network flows and resources has benefits in decreasing the required fleet capacity [38], which also helps to reduce congestion, emissions [39], and noise pollution [40].

Increasing process and resource efficiency plays an important role in the 21st century in a sustainable economy [41]. Not all activities add value to the product or the service. These activities are considered wasteful because they do not directly provide customer value, even if necessary. The concept of lean philosophy means the usage of a system of techniques and activities for the purpose of eliminating all non-value-added activities (Waste of overproduction, Waste of time on hand, Waste of transportation, Waste of processing, Waste of excess inventory, and Waste of movement) and waste (Waste of making defective products) from business with maximum resource utilization (Waste of underutilized workers) [42]. These seven wastes, or the so-called “Muda”, are determined by lean philosophy [43]. There are many lean tools available to support waste identification [44]. By identifying these wastes, non-value-added activities can be minimized and value-added activities can be optimized. Therefore, it is important to represent them as a separate temporal network. For example, when a taxi driver taxes a passenger, the trip can be considered a value-added activity. In contrast, when a cab is empty after dropping off the passengers and the taxi driver drives to the next place of demand, this state transition can be considered a non-value-added activity. The main idea of the work is that a model is provided to represent these non-value-added activities as different layers in a multilayer network, distinct from the value-added activities.

Model-based problem solving plays a crucial role in logistics and in transportation, employing case-dependent methods and models to optimize decisions and enhance cost-effectiveness. This approach is exemplified in various studies such as optimizing vehicle routes in retail distribution networks using mathematical modeling solved with meta-heuristic algorithms [45], determining optimal prices and inventory strategies through game theory models [46], and defining the optimal lot number and price of perishable products for retailers [47]. Similar to perishable products, optimizing the supply and distribution of blood products is also challenging due to uncertain demand, but the mathematical model with a heuristic solution algorithm selects optimal routes and minimizes costs [48]. Grey optimization can also be used to improve the performance and efficiency of contracts used in the supply chain [49] or blockchain technology for causal associations to determine the purpose of personalized customer service [50].

Discrete event simulations can be highly effective for evaluating the decision to implement lean manufacturing principles and assessing their benefits [51], and they are also used in Value Stream Mapping to identify non-value-added activities, such as in the case of a tire distribution company [52]. Petri nets can also be relevant in manufacturing optimization, such as scheduling production activities in cylinder block remanufacturing, where they serve to search for optimal or near-optimal schedules to improve system performance [53] or for example in disassembly process [54]. Implementing and validating the model-based problem solving solution, such as discrete event simulations requires significant time and resources, making them costly. Any changes to the system during model updates can be complex and labor-intensive. In addition, the high cost of the necessary software and expertise, as well as the significant computing power required for accurate simulations, can be major drawbacks. The drawbacks are similar to those of discrete-event simulation, with the added challenge of validation to ensure that the model accurately reflects real-world processes.

Multilayer networks can process complex data and find connections at different levels, which is why they are increasingly common in traffic-related analyzes.

Recently, multilayer network-based models have been developed to handle different modes of transportation [55, 56, 57]. Multitemporal transport networks have also been shown to be beneficial in traffic modeling by handling different intensities over time [58] or detecting an anomaly in the system [59].

Other example, analyze the Chinese Airline Network from the perspective of multilayer networks [60] or the impact of new air route strategies on the resilience of a multilayer Chinese Airline Network [61]. Multilayer networks can be used for resilience testing in case of random flight errors in the European Air Transport Network [62]. The method is also usable for the detection of urban dynamics as in the case of a railway network [63]. Multilayer network-based models can also be used for efficiency testing in public transportation systems [64].

The state-of-the-art in the field of multilayer network-based evaluation of the efficiency and resilience of network flows involves several key approaches:

- Network flow analysis: Utilizing mathematical models to understand the flow of resources within a network and identify areas for improvement [65].
- Temporal analysis: Examination of network flows over time to identify trends and patterns and assess long-term efficiency and resilience [57].
- Resilience evaluation: Assessment of the network's ability to withstand disruptions and continue to function effectively [66].

- Integration of data sources: The use of a wide range of data sources, such as transportation data, demographic data, and weather data, to inform the evaluation [67].
- Predictive analytics: Use of machine learning algorithms to anticipate future network flows and assess the possible impacts of changes [68].
- Dynamic routing: Real-time adjustment of routes and allocation of resources to improve efficiency and resilience [69].
- Collaborative transportation: Encouragement of collaboration among stakeholders, such as carriers, shippers, and government agencies, to improve efficiency and resilience [70].
- Infrastructure improvements: Investment in physical infrastructure, such as roads, bridges and airports, to improve capacity and reduce congestion [71].
- Environmental sustainability: Consideration of the environmental impact of network flows and efforts to reduce emissions and promote sustainable transportation [72].

The third thesis addresses a gap in the analysis of location data by introducing a framework that distinguishes between value-added and non-value-added activities within complex systems. This research bridges the gap by applying multi-layer network modeling to classify and optimize resource flows, providing a transferable methodology for IPS-based industrial environments. By integrating advanced clustering and network analysis techniques, it improves the identification of inefficiencies, aligning with the goals of process optimization and resource allocation in manufacturing and logistics.

By using multi-layer network modeling to analyze resource flows, the foundation is laid for further refinement through clustering techniques tailored to classify activities as value-added or non-value-added. These methodologies are consistent with lean principles and provide a way to optimize logistics and manufacturing processes by systematically reducing waste and improving resource efficiency.

The Lean philosophy and logistics processes are deeply intertwined, as the application of Lean principles optimizes logistics activities by eliminating waste, maximizing resource efficiency, and streamlining processes in both logistics and manufacturing environments [73, 74]. By focusing on reducing unnecessary activities and enhancing value-added tasks, companies can significantly improve their operational performance.

With the advent of Industry 4.0, digital technologies such as advanced analytics and the Internet of Things (IoT) have become critical in supporting Lean practices in manufacturing and supply chain management. These technologies outline eight waste reduction mechanisms that improve operational efficiency and enable data-driven decision-making, providing a clear framework for selecting the most effective methods to enhance logistics processes [75].

The concept of "Digital Lean" highlights the role of digital tools, such as Information Technologies (ITs) and Operational Technologies (OTs), in enhancing Lean manufacturing. These tools allow for the detection and prevention of physical waste through simulations and real-time monitoring. Digital Lean also addresses digital waste, which arises from the underuse or overuse of smart manufacturing technologies [76].

Building on this, Lean 4.0 incorporates a broader set of Industry 4.0 technologies like artificial intelligence, robotics, and cloud computing to further optimize manufacturing processes. This integration enables more intelligent automation and deeper alignment of Lean principles within modern manufacturing environments [77].

As previously mentioned, the concept of muda in industrial processes focuses on eliminating waste, with idle activities in transportation, such as transport waste, being classified as non-value-added, as they do not contribute to value creation from a logistics perspective. Minimizing idle activities is crucial for effective resource utilization and improving process efficiency [78].

One of the key challenges is identifying these non-value-added activities. Several methods can be employed to address this, such as the Spaghetti Diagram [79, 80, 81], which is a visualization tool that helps depict movement and interactions during processes, allowing them to be classified and analyzed. Another approach is Value Stream Mapping (VSM) [82, 83, 84], which reviews intralogistics processes to identify idle times and eliminate unnecessary steps, thus improving material flow and reducing downtime.

Another solution involves using Manufacturing Execution Systems (MES) and Indoor Positioning Systems data, combined with data analytics and data science techniques, to identify non-value-added activities. MES is a software system that monitors and controls manufacturing operations on the shop floor, providing detailed, real-time production data, while IPS provides precise location information for tools, materials, and personnel. By integrating these data sources, advanced analytics can be used to identify inefficiencies such as idle time, unnecessary movements, or bottlenecks, enabling a more detailed understanding and optimization of intralogistics processes.

These IPS systems address critical logistics challenges, such as improving the accuracy of warehouse positioning and ensuring reliable monitoring of goods in transit within large facilities [85]. In addition, IPS facilitates real-time value stream mapping, helping to identify bottlenecks and optimize processes through accurate state tracking [[86]].

Clustering techniques, such as DBSCAN, can be applied to position data provided by Indoor Positioning Systems to identify patterns in resource movements, helping to distinguish between value-added and non-value-added activities, thus optimizing logistics processes and improving overall efficiency.

Clustering [87, 88] is a method that can assist in identifying areas or periods where the movement or activity of resources significantly deviates from the norm, or in identifying potential areas for optimization. Time-series classification (TSC) is widely utilized in manufacturing systems, including applications such as supply chain optimization. The importance of TSC has grown significantly in smart manufacturing systems, driven by the integration of Machine Learning (ML) and Deep Learning (DL) algorithms, which are essential for processing the vast amounts of time-series data generated by these systems [89].

Additionally, Machine Learning-based methods [90] facilitate the creation and training of models to recognize different states, analyzing data to detect anomalies, such as periods of idleness, through various algorithms. Movement Trajectory Analysis [91] examines resource movement patterns to identify areas or routes where resources are less active or exhibit significantly different trajectories.

A multilayer network framework enables the analysis of complex systems by capturing multiple types of interactions between entities across different layers, providing a comprehensive understanding of interconnected subsystems [[92]]. These networks are also utilized for inventory optimization in e-commerce platforms, integrating material flow, inventory management, and pricing strategies to improve supply chain efficiency [93]. Furthermore, a multilayered temporal network-based model distinguishes value-added from non-value-added resource flows, enabling a comprehensive view of the flow of resources in the system [94].

The fourth thesis addresses the gap in the use of clustering techniques, specifically DBSCAN, to distinguish value-added from non-value-added activities in industrial settings. By refining DBSCAN to classify activities based on position data, this thesis provides a methodology to systematically identify inefficiencies. It bridges the gap by integrating clustering results into process optimization, thereby improving resource allocation and operational performance in industrial environments.

Analyzing position data and classifying activities are essential for optimizing manufacturing and logistics processes. These insights not only improve the efficiency of existing systems, but also lay the foundation for automated decision-making frameworks, such as optimal task scheduling for AGVs, driven by Industry 4.0 advances.

The principles and technologies of Industry 4.0 are transforming manufacturing paradigms by integrating digital technologies and automation into production processes. Automated production lines, designed according to Industry 4.0 principles, allow machines and devices to collaborate within the Internet of Things (IoT) framework [95], fostering a lean manufacturing system [96] characterized by increased efficiency, reduced possibility of errors, and flexible management of fluctuations in production volumes. The further application of Industry 4.0 creates opportunities for automated material supply, where scheduled automated transport of raw materials, semifinished, and finished products optimizes manufacturing processes [97]. It also enables the implementation of inventory strategies within the lean philosophy that optimize inventory levels as a type of waste [98, 79]. This requires the use of tools such as Automated Guided Vehicles (AGVs) or automated warehouses that are seamlessly integrated into the production system to ensure flexible and intelligent material flow in manufacturing.

To optimize these processes, it is crucial to understand the specific activities performed by each logistics and production element at every stage. Although some of this information can be extracted from the enterprise resource planning system, tracking the tools involved in the supply chain requires the use of Indoor Positioning Systems [[1]]. This technology plays a crucial role in achieving full traceability by enabling the definition of states using precise position information provided by the system, allowing for accurate tracking of the current position and state of AGVs. As a result, the ability to monitor production in real-time is gained, allowing forecasts to be made based on this information and a simulation environment to be created to analyze the impact of changes in the production process, thereby laying the foundations for the creation of a digital twin [99].

However, integrating industrial automation into the IoT framework often presents challenges [100] due to the complexity of production processes, the heterogeneous manufacturing environment, and the intricacies of internal stocking and material supply scheduling caused by unbalanced production processes. The application of machine learning methods [101, 102], such as reinforcement learning algorithms [103, 104], could provide a potential solution. The Markov Decision Process (MDP) model serves as a mathematical tool to achieve optimal task scheduling for AGVs [105, 106, 107, 108]. The MDP model's components include states, actions, state transitions, and rewards, which are crucial for defining the state of the system, state transitions, possible actions, and rewards [109].

To achieve a model-level representation of the real production system, it is essential to analyze our existing production and logistics processes. Following this, it is worthwhile to examine whether process optimization is possible, as these factors impact the development of the MDP model. The design of the model is of paramount importance in the context of the reinforcement learning agent's learning process and the determination of the optimal decision strategy. The agent, which functions as the decision-making entity, seeks to maximize the cumulative rewards obtained, continuously learning from interactions with the environment and feedback received. By collaborating with the MDP model and the agent's learning process, RL algorithms shape the optimal decision strategy based on available information, taking into account the complex interconnections of states, actions, and rewards [110, 111, 112]. The gap addressed by the fifth thesis lies in the limited exploration of how Markov Decision Process (MDP) modeling can be effectively applied to optimize task scheduling and resource allocation in industrial environments.

Chapter 2

Utilizing RTLS positioning data in manufacturing: insights into process efficiency and operational applications

2.1 Introduction to data-driven process optimization with RTLS

Getting accurate and actual information of a process status is very important in the management and development of production systems. Information is often position located; this way, it defines the actual position of a workpiece or resource in the production area. This location based information may be suitable to connect information of resources and activities/workpieces. The purpose of this thesis is to introduce the potential in tools developed for indoor positioning, as well as the available technologies and the possible use of data hidden in information.

According to the ISO/IEC 24730-1:2014 standard, the real-time locating system (RTLS) is a wireless system used to locate the position of an item anywhere in a defined space at a point in time that is or is close to real-time. Indoors positioning systems (IPS) [113] locate objects in closed structures, such as office buildings, hospitals, stores, factories, and warehouses, where the GPS proves to be inaccurate [114]. In this section, the focus is placed on how indoor positioning can be utilized in manufacturing, and for simplicity, these indoor positioning systems are referred to as RTLS.

Several surveys and comparative analyses can be found on indoor tracking technology based on localization techniques [115, 116, 117]. One article has attempted to classify techniques and systems by presenting a comprehensive performance comparison of the accuracy, precision, complexity, scalability, robustness and cost [118]. Similarly, studies comparing RTLS technologies can also be found in the literature [119, 120, 121, 122, 123, 124, 125]. Furthermore, a meta-review provides a comprehensive compilation of 62 survey papers on the topic of RTLS [126], and the classification of current typical RTLS is introduced with a layered conceptual framework [127]. Many technologies are available such as infrared light, ultrasound, laser and their combinations. The reviews of these technologies focus on the technical elements and standalone applications and show that only a few specific industrial applications are available [128].

This section aims to provide an overview of the applicability of RTLS in manufacturing to support the practical applications and provide a guideline or reference for implementation, research and development of indoor positioning and RTLS.

In order to explore the potential applications a systematic examination of literature was performed in Scopus, following the PRISMA-P protocol. The used keyword set ("real-time positioning systems" OR "indoor positioning systems") AND ("manufacturing" OR "industry") resulted near to 300 articles from which the thematic groups of the related research were identified.

Positioning data in the production system is the key information for traceability [129], and digitalization [130]. The potential technologies and possible traceability levels are overviewed in Section 2.2. The levels represent the identification unit from the transportation unit (highest level— trucks, ships) to item unit (lowest level–raw material). Determination of the traceability level depends on more factors such as the complexity of the production process, the number of raw material types and the conditions of the information system and infrastructure.

The potential manufacturing applications are discussed according to the tasks depicted in Figure 2.1. The figure describes how positioning-based information allows for continuous improvement to other parts of the manufacturing environment, such as production control, logistics, applications in quality management, safety and RTLS-based efficiency monitoring. These applications and the required data analysis tasks are discussed in Section 2.3.

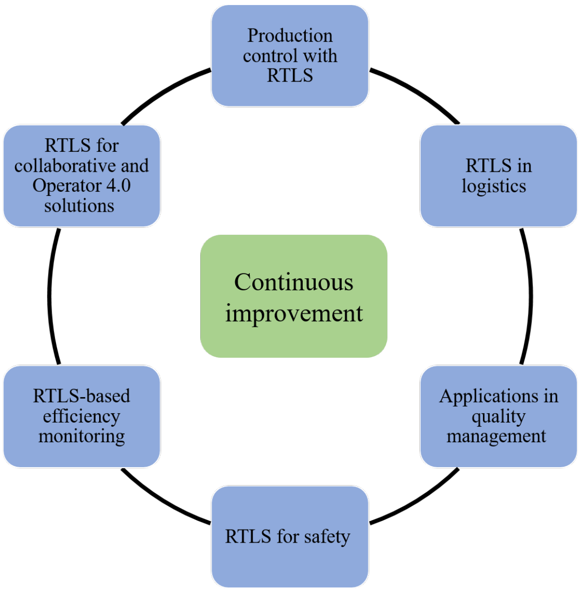


Figure 2.1: Use of real-time locating system (RTLS)-based positioning information by the different parts of the manufacturing environment. Continuous improvement is a central element of an RTLS project.

Section 2.4 describes a workflow to implement an RTLS-based digitalization project such as installation with the necessary hardware elements and data processing to allow the data integration. Finally, the applicability of the RTLS in manufacturing is illustrated by a case study presented in Section 2.5.

2.2 Levels of location information in manufacturing industries

To determine the appropriate tracking technology, the identification levels along with the associated technologies need to be known. In terms of integration into our system, it is important to see the relevant characteristics for the selection process of the particular RTLS technology. In this section, the criteria for choosing a method for a tracking solution arising in a production system are presented. This section also includes recommended indoor positioning-based tracking technologies as part of its content. Please note that a detailed description of the technologies is not the purpose of this research.

Identification levels and technology solutions

Different applications require different types of tracking systems. Figure 2.2 shows the identification layers based on the possible available levels.

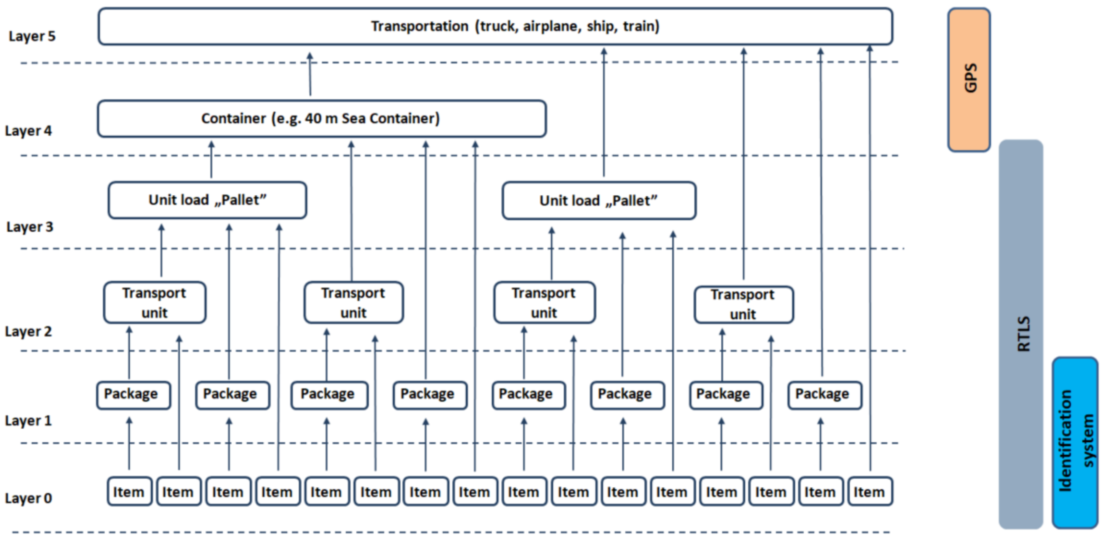


Figure 2.2: Identification levels in a production system. Layers define the logistic units from raw material (items) to trucks (transportation).

GPS is used for tracking containers and transportation equipment. At the lower level, where intralogistics are considered, GPS is not accurate, or in many cases, it is deemed unusable and unsuitable for general asset tracking due to energy consumption. RTLS can handle indoor container identification as well as the unit load, transportation unit and package, that is, the third, second and first layers. RFID and barcode technologies are possible solutions for item identification, but it is essential to consider whether using a particular technology is appropriate or if it is worthwhile to combine the available technologies, such as UWB technology (which is not recommended for raw material tracking). This approach is called hybrid traceability technology in the literature [131]. However, with RFID tags, unit identification can be achieved with a lower cost [119].

Table 2.1 shows solutions for these different identification levels (see in Figure 2.2) with the advantages and disadvantages. Four different traceability solutions are discussed to support the chosen technology. A decision-making model for selection is proposed in Reference [128], where UWB, RFID, Wifi, Zigbee and BLE (Bluetooth Low Energy) are compared with several aspects based on the developed methodology. The steps of this methodology are RTLS definition, market analyses, weights of criteria, ranking. They made a comparison between the technologies with the many parameters (coverage area, accuracy, room level usefulness, RF interference potential, bit rate, complexity, initial cost, security and privacy, health concern). Application-oriented parameters are focused on in Table 2.2.

Structure of Indoor Positioning Systems and Potential Traceability Technologies

Table 2.1 helps us to choose the right technology in the case of layer 0 and 1. The applicability of the RTLS is now the focus. A multilevel selection criteria [132] was found, where the three levels are Economic, Technical, and Implementation. An overview of the most relevant RTLS technologies (excluding the no-radio based technologies) is provided in Table 2.2. It summarizes these technologies with critical performance criteria, including accuracy, power consumption and costs.

Figure 2.3 gives us a classification of RTLS. The left side of the figure was considered because there are very few examples of no radio-based technologies in actual industrial applications. In the production environment, ultrasound signal transmission is also accompanied by a radio frequency (RF) pulse to combine the high accuracy of ultrasound with the high communications capacity of RF, which enables tracking of hundreds of simultaneously moving tags [133]. The technology is not used independently in the manufacturing environment because of the communications capacity, and environmental noise can degrade the localization accuracy [122]. One standalone application of RTLS is tracking the locations of construction resources such as labor, materials, machinery, and vehicles [134]. This application uses WiFi-based RTLS because GPS is limited in indoor environments, such as tunnels and buildings under construction. Another example is the development of a self-governing mobile robot navigation system for indoor construction applications [135]. Several navigation strategies with a mobile robot were tested with various combinations of localization sensors, including wheel encoders, sonar/infrared/thermal proximity sensors, motion sensors, a digital compass, and ultra-wideband (UWB) technology. The findings can be adapted to several potential construction or manufacturing applications such as robotic material delivery, inspection, and onsite security.

Two RTLS applications of UWB and ultrasound technology have been tested in the SmartFactory KL [136].

Table 2.1: The most widely applied solutions for traceability. The advantages and disadvantages help us to choose the right solution for the right layer (see in Figure 2.2).

Solution		Advantages	Disadvantages
Identification system	Barcode	Cost effective, spread technology	The potential for human error is high
			Additional installation cost per every layout changing
		The cost per item is low	The cost of printed labels can be relevant in the case of enormous stock
	RFiD	Cost effective, spread technology	Every new station needs new hardware and installation
		The cost per item is low	Additional installation cost per every layout changing
		The human error can be minimized	
GPS		Spread technology	The accuracy is not enough in the case of indoor positioning
		Many devices are already compatible	The reliability is low in the case of indoor positioning
		It is highly scalable	
RTLS		The error of human is excluded	The cost depends on the number of tracked items
		The traceability is available at the covered area	
		The system is fully flexible, any layout changing can be handle in the software application	Any new item can be added at any time to the system (highly scalable)

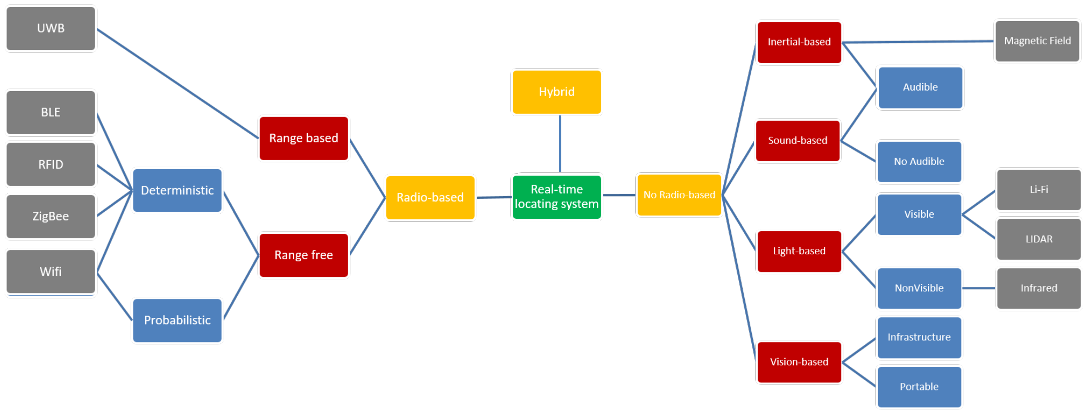


Figure 2.3: Classification of RTLS [137].

Compared to other technologies, Zigbee has not spread substantially in industry [138]. However, industrial applications in the literature, such as the Zigbee positioning system for coal miners [139], have also been studied [140]. Laser-based systems are also used for navigation [141] and production tracking [142]. RFID is used in the production independently as an identification system [143] because only the presence of tags, such as barcodes, can be accounted for at the RFID reader. Other technologies must be used for real-time location [144]. An RFID-based RTLS solution exists, but this solution is less widely used because it is more expensive and inaccurate than UWB [145]. RTLS must be able to locate, track and identify objects in an indoor environment; therefore, RFID technology is not appropriate for RTLS. Given the controversial literature in this area, various types of technology, such as Bluetooth, WiFi, Zigbee, and UWB, are considered to have the ability to support RTLS. Apple proposed the iBeacon protocol [146] in 2013 as another tool [147]. The new iPhone 11 from Apple already includes UWB beacons, which may be suitable for indoor positioning [148]. Moreover, it is essential to mention 5G technology [149], which could be crucial for future smart manufacturing, including highly accurate indoor localization. Because of the large-signal bandwidth and beamforming capabilities, localization and tracking could be more robust and efficient [150]. The technique is only now beginning to spread in industry; currently, it can only be applied in test and development environments [151]. The different indoor positioning-based traceability technologies can be combined in RTLS (already mentioned hybrid technology) to take advantage of different solutions in one system; for example, the ZigBee and UWB technologies or RSS measurements and a fingerprinting location algorithm usage for better position estimation [152]. To use hybrid technology, a platform is also needed. In general, roughly five layers are worth defining. A hardware layer, where position data are generated; a processing layer, where the position is calculated and filtered; a data layer, where location data is stored; a service layer, where the system can be optimized; and a visualization layer, where real-time location data can be analyzed and monitored.

Table 2.2: Review of indoor positioning-based traceability technologies

Techn.	**Tag Cost [154, 155, 156]**	**Module Cost [153]**	**Accuracy**	**Space dim.**	**Power cons. [157, 153]**
Scale	**L:<3$ M:<10$ H:>20$**	**L:<10$ M:<40$ H:>70$**	**L:>1 m M:10 cm H:<10 cm**	**2D/3D**	**L:<100 mA H:>200 mA**
Zigbee [158]	M	M	M	2D	L
RFID [159]	L	H	L/M	2D/3D	L
BLE [160]	L	L	L/M	2D	L
Wifi [161]	H	H	L/M	2D	H
UWB [162]	H	H	H	2D/3D	H

In the next Section 2.3, potential industrial applications are described, while in Section 2.4 a workflow for setting up an RTLS-based manufacturing support system is proposed. Finally, a use case is described to illustrate the applicability of RTLS.

2.3 Industrial applications of RTLS

An approach to RTLS selection is reviewed [128] and the RTLS based articles and the main advantages are summarized within Table 2.3 with the related fields of application in production and logistics categorized according to areas of use such as Quality Management, Safety and Efficiency Monitoring. Within the topics, reference is made to existing solutions, but possible directions for development are also presented in the following subsections.

Different application possibilities exist in manufacturing departments with RTLS. Table 2.4 presents various types of applications, where the RTLS-provided information is defined for every industrial application. The possible benefits define how the efficiency of the RTLS project can be validated. The applied positioning system provides real-time information about where equipment, semi-finished or finished products and specified logistic vehicles or workers are located in the manufacturing area [163].

Production control with RTLS

Cycle time optimization is a critical task, especially in the case of modular or just-in-time (JIT) production [164]. A positive correlation exists between the potential of RTLS and JIT manufacturing. Several objects are defined as the main focus for tracking in the case of industry applications (mobile assets, workers, materials, key components, forklifts, pallets) [132].

The cycle time optimization is also possibly based on the position of products. For that, it is necessary to know which products are on the assembly line at every moment in time [164]. Pairing a semi-finished product with a tag makes inter-manufacturing tracking possible and makes the following information available:

- Time spent on the workstation for a given product;
- The production sequence;
- Which products are/have been on rework;

- Which products are/were in quality assurance;
- Average lead time for a particular product type (tact time);
- The goods in production are available with a continuous, real-time production status that supports production programming and shift design.

A position data-based decision-making approach is presented that relies on advanced data analytics for asset location systems to help production [165]. A potential use case in construction is discussed in Reference [166] and another application of material tracking in a pipe spool fabrication shop in [167].

Table 2.3: Industrial applications of RTLS technologies.

Application type	**Application**	**Technology**
Production	Cycle time optimization	UWB [164]
Production	Position data-based decision making	UWB [165]; RFID [168, 169]
Production	Activity-Time monitoring in production line	UWB [170]
Production	Digital Facility Layout Planning	Independent [171]
Logistics	Logistics management	RFID [172]; Hybrid [133]; Independent [132]
Logistics	Warehouse management	RFID [173]; WiFi [174]
Logistics	Pallet management	RFID [143]
Logistics	Material/component and production tracking	WiFi [167]; UWB [175, 143]; RFID [176, 163, 177, 178, 143]; Hybrid [179]; Laser [142]; Barcode [180]
Logistics	Assets tracking	Bluetooth [166]; RFID [181, 182, 183, 184]; Hybrid [185]; UWB [186, 187]; Laser [141]; Barcode [180]; ZigBee [158]
Quality	Weak spot analyzis in production	UWB [188]
Safety	Safety management	RFID [145]
Safety	Collision avoidance	UWB [189]
Safety	Personal protective equipment monitoring	Hybrid [190]
Safety	Person tracking	ZigBee [139]; RFID [191]; UWB [192]
Safety	Contact tracking	Independent [193]
Efficiency monitoring	Performance of manufacturing process	RFID [194]
Efficiency monitoring	Lean manufacturing	UWB [195]; BLE [160]
Efficiency monitoring	Human resource monitoring	RFID [196]

Table 2.4: Application of RTLS in manufacturing based on Section 2.3, the useful information it provides and possible benefits.

Application Name	Information Provided by RTLS	Possible Benefits
Production control with RTLS	Footprint of semi-finished products and cycle time control	More efficient production planning
RTLS in logistics	Tracking of logistical assets in the production system	More cost-effective logistics process planning
Applications in quality management	Root cause analysis depends on position data	Help quality management department comply with standards and regulations
RTLS for safety	Human and material handling equipment tracking can help in collision detection	Reduction in occupational accidents
RTLS-based efficiency monitoring	Efficiency indicators provide a realistic picture of real-time production	Real-time efficiency monitoring assigned to machines or tools can support making better decisions
RTLS for collaborative and Operator 4.0 solutions	Precise real-time position of operators to predict the possible collaboration situations	More efficient decision making for the smart operator and collaborative system

Similar RTLS-based position data are already available in the literature, where a real-time connection between operator performance and varying product complexity was designed [170]. Another paper proposes an RTLS-based solution for a logistics problem with hybrid traceability technology (WiFi with RFID) to realize materials tracking, which can automate considerable amounts of warehouse work, such as stock-taking and storage positioning and checking [163]. Related research proposes an RFID-based intelligent decision support system architecture to handle production monitoring and scheduling in a distributed manufacturing environment [168]. Furthermore, RTLS technology can even be an element of reconfigurable facility layout planning. With its help, the processing steps of the activity and their relationships can be easily mapped and recorded in a database. More complex material flow can be provided (with information of the real flow between machines or congestion phenomena), which is not possible with a simple flowchart [171].

With the proliferation of Industry 4.0, it can be seen that there is also a strong emphasis on production monitoring. This is evidenced by the numerous references presented in the topic of production management.

RTLS in logistics

Logistics efficiency is largely dependent on the movement of forklifts, pallet trucks and stacker trucks. To improve logistics efficiency, current processes should be understood to identify points where lead time reductions can be achieved in the supply chain. A properly selected RTLS technology can be a tool for exploring logistics processes [175]. The following information is made available with logistical vehicles tracking:

- Routes and time spent in specific areas;
- Speed of forklift;
- Data for predictive maintenance;
- Forklift overall equipment effectiveness (OEE).

The objective of [173] is to propose an IoT and advanced data analytics-based warehouse management system (WMS) to enable smart logistics for Industry 4.0. The proposed IoT-based WMS can improve warehouse productivity, picking accuracy and efficiency, and it is robust to order variability. In [180], the authors present a sophisticated algorithm for tracking production and determining the traceability of a product. Reference [176] gives an example of how to apply RTLS across the supply chain and manage various assets within shop floors. A forklift based use-case is described in [179] where movement inside a warehouse determined by the RTLS is associated with assets that it picks up (attach through UHF-RFID reading) or puts down (detach through the loss of RFID signal).

IR-UWB-based RTLS has been deployed in an in-operation warehouse to track forklifts [187], and an RFID-enabled positioning system in AGV for a smart factory has also been presented [181]. Observations and lessons from simulation and testbed studies could be used to guide automated logistics within a smart manufacturing shop floor. The framework of an R-AGV-based material distribution scheme is proposed [185] based on an RTLS platform and electronic workshop map. The analysis and experimental results indicate that the R-AGV-based material distribution system provides new levels of process visibility and efficiency compared to traditional AGV-based distribution systems.

Tracking of transportation device is mandatory to obtain an accurate picture of intralogistics processes. A related paper describes an industrial forklift tracking problem that requires precise internal positioning [186]. It presents a study on the feasibility of meeting this challenge using UWB technology. Placing two tags on the forklift enables even more robust localization, as the measurements from the two tags are combined.

Based on the studied articles, it can be seen that much RTLS-related research is being done in logistics; there are still, of course, unexplored application possibilities.

Applications in quality management

It is possible to see where losses are generated with a real-time tracking system. The monitoring of the material flow with RTLS and the average duration of the processes provide information about problematic weak spots in the production process. Based on this information, possible reasons for the delay in production is discussed in Reference [188]. Root cause analysis is an essential component of quality assurance for the customers of the manufacturing company. After exploring the root cause, different action plans can be implemented, such as a review of the workflow, redesign of the workspace, education of workers and modification of work instructions.

Another possible advantage is RTLS-based dynamic work instruction. A crucial point in non-automated and human resource-required production processes is to ensure the well-supported work of the operators. One possible solution is showing just the information required to process the actual workpiece and no more. Based on full traceability, the actual work instruction at every workstation can be shown using the product information from RTLS. There are relatively few practical applications on the topic, but potential development opportunities can be clearly identified such as dynamic work instruction.

RTLS for safety

Collision avoidance is one possible improvement in the EHS (Environment, Health and Safety). Real-time alerts and notifications can be developed to prevent accidents based on the movement of vehicles and workers. A significant portion of the workplace accidents can be traced back to failure to use the required protective equipment. The monitoring of the personal protective equipment (PPE) usage is also an available function using RTLS [190]. RTLS technology can be used to control access to restricted areas for employees by sending automatic alerts whenever someone enters an unauthorized area [191]. Moreover, in the event of an emergency or natural disaster, such technology can be used to determine if everyone has already left the area or whether every worker used the designated route to leave the building [192].

The main purpose of contact-tracking solutions in industry is to help identify the contact matrix when the infection is recognized. RTLS is one possible solution to support to explore the potentially infected people [193]. Therefore, it can be used for the protection against the COVID-19 pandemic [197]. RTLS can be used in production systems also to monitor adherence to distance requirements between the operators. This is a possible useful function for a COVID-19-like epidemic situation. Due to the pandemic, the field of research for RTLS is topical, but there are also several solutions in the literature in the field of EHS (Environment, Health and Safety).

RTLS-based efficiency monitoring

A related research article presents an RFID-based RTLS solution for performance metrics through RTLS data analysis to evaluate workflow performance and to obtain a lean process [194]. Spaghetti diagrams (visual representation using a continuous flow line to trace the path of an item or activity throughout a process) are time-consuming and static and, therefore, do not reflect the dynamics of logistics systems. RTLS was proposed to overcome this drawback [195].

The efficiency of a human resources personalized measurement is challenging. To achieve set goals, every organization must devise adequate, effective and efficient means of managing its HR. Related research reported on the development of an RFID and RTLS-based real-life personnel monitoring system to accurately and reliably estimate distance and coordinate the location of personnel at any instant [196]. This method can be used to measure—based on RTLS positioning data—how much time each product has spent at a particular station. This measurement can be further developed, and performance indicators can be obtained for workstations where operators work. By breaking down the overall process into sub-processes, the zones that allow for personalized performance tracking can be refined. The proposed RTLS can provide a solution to compute availability, a key parameter of OEE (Overall Equipment Efficiency) based on position data. Position data could improve the accuracy of the measurement of human resource efficiency (HRE) [198] and integrate other sensor measures for real-time activity monitoring [170]. Real-time sensor data assigned to the location of tools or machines on the shop floor allows online efficiency monitoring and supports the development and maintenance of digital twins [198] or intelligent decision-making systems.

In terms of human resources, there is still quite a bit of related research. More potential RTLS applications—like the mentioned personalized performance tracking—can help to achieve more efficient operations.

RTLS for collaborative and operator 4.0 solutions

The future of manufacturing will be the personalization, and Industry 5.0 defines by the co-operation between man and machine [199]. During the current fourth industrial revolution, companies have realized they need to put humans back into industrial production with collaborative robots [200]. The workers need to be upskilled to provide value-added tasks in production to handle the mass customization and personalization for customers. This philosophy overlaps with the Operator 4.0 concept [201].

Shop floor trackers are one of the required technologies of Industry 5.0 [200]. RTLS could be a suitable solution for the full traceability on the shop floor. The smarter operator is an element of Operator 4.0 methodology [202], which is used to be the intelligent personal assistant-based solution. Real-time position data providing precise location information helps the system to make better decisions for operators and make possible the trajectory prediction of operators [203].

Analysis of position data and building data-driven solutions

Raw data provided by the RTLS cannot directly be utilized to support the manufacturing. The purpose of this section is to introduce data based solutions and the related data analysis techniques needed for data pre-processing and building data-driven solutions. Again, a systematic examination of literature in Scopus was done, using the keyword set ("indoor positioning" AND ("machine learning" OR "data science" OR "data mining")). The network of the mentioned keywords can be seen in Figure 2.4. Based on this network, the key thematic groups of machine learning techniques and the related application areas, which will also be discussed in detail in this section, can be defined. Data mining techniques are reviewed in Reference [204] to solve indoor navigation systems problems. The performance of the RTLS is shown via the integration of different features and classification algorithms, including decision tree, multi-layer perceptrons, and Bayesian networks [205]. In another article, naive Bayes theorem-based classification techniques and other classification techniques to enhance the classification accuracy are compared to identify the best location estimation algorithm [206]. K-nearest neighbor [207], support vector machine [208], decision tree [209], naıve Bayes [210] and Bayesian network methods [211] are compared and combined with ensemble learning algorithms to improve the performance, i.e., accuracy, f-score and computation time [212]. Decision tree-based classification is applied to estimate the position to improve the accuracy [209]. The clustering machine learning (ML) technique is usually used to improve RTLS accuracy, like K-means clustering backpropagation NN [213], Spatial Division Clustering (SDC) method [214], affinity propagation clustering [215]. The feeding behavior of cows is measured with RTLS in [216]. The presence at the feeder (feeding probability) of the cows was calculated using the logistic regression model. Support Vector Regression (SVR) is used to calculate the efficient RTLS [217]. Based on the aforementioned literature study, the common combinations of ML techniques are presented in Table 2.5 and RTLS technologies can be seen in Figure 2.5. Based on the qualitative analysis of the literature it can be highlighted that many researchers have successfully applied NNs to the indoor positioning problem via convolutional neural networks (CNNs) [218]. A ZigBee [219] indoor positioning research scheme based on the location fingerprinting approach uses an NN locating model. This model, with the signal-index-pair data pre-processing method, is used to increase positioning precision [220]. Related research uses a particle swarm optimization-based backpropagation (PSO-BP) NN to determine the relationship between RFID signals and the position of a tag for an RFID-based positioning system [221]. Furthermore, to improve the quality of training samples, the experimental data are pre-processed via Gauss filtering. The following section is devoted to show how RTLS systems and the presented models can be integrated into one system.

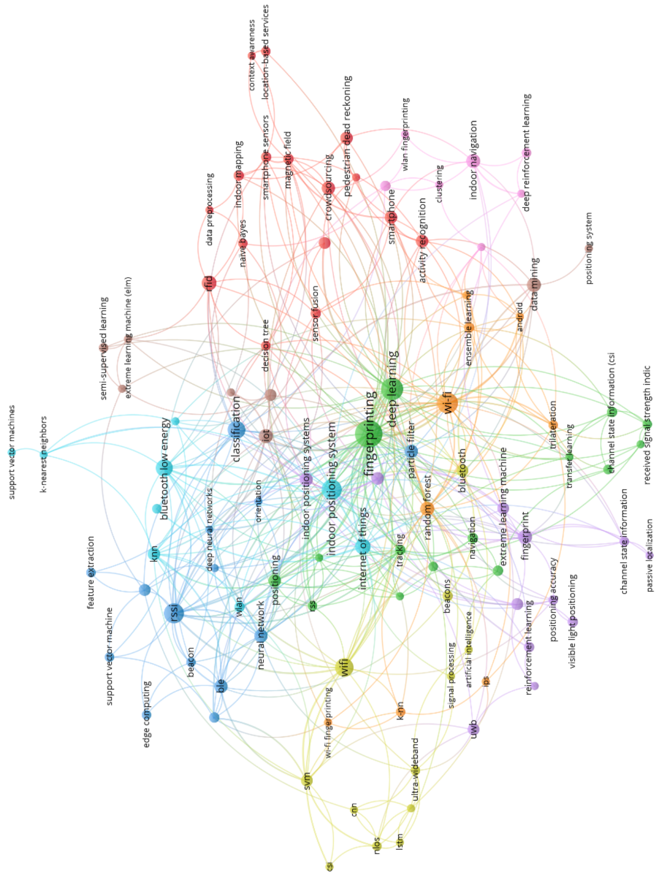


Figure 2.4: Network of keywords based on Scopus database.

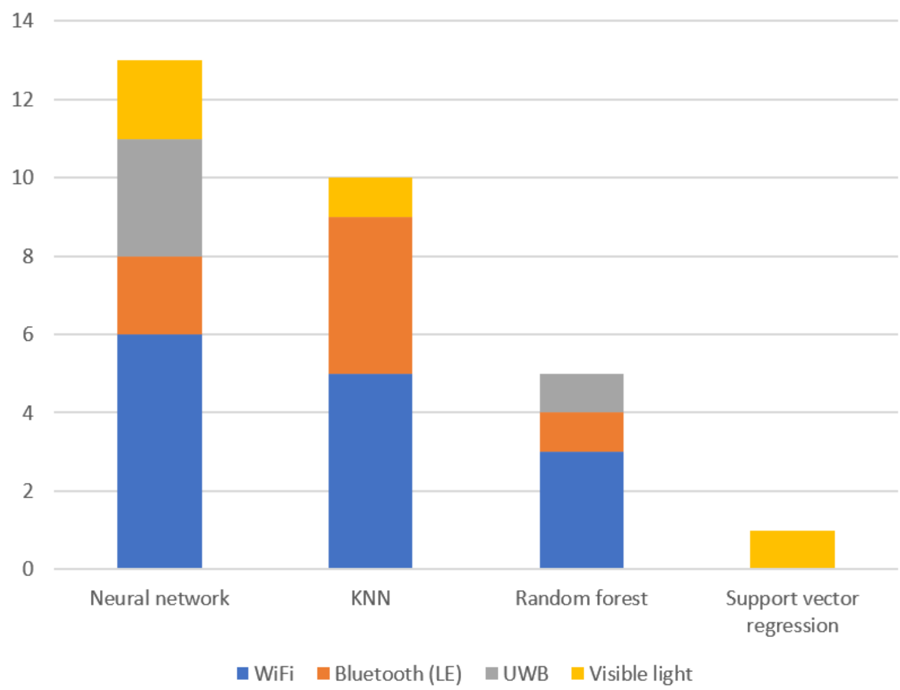


Figure 2.5: Concurrence of machine learning (ML) techniques and RTLS technologies in articles.

Table 2.5: Data mining techniques and areas of RTLS-based application

Method	Definition	Data Analytic Techniques	Application Areas	RTLS-based Applications
Classification	Discriminating data into different labeled subsets pursuant to class attribute. Retrieving important and relevant information about data and metadata.	Neural network Support vector machine (SVM) Decision tree k-Nearest neighbor Bayesian network Genetic algorithm	Pre-defined distribution (e.g., identification of differences) Fault detection Anomaly detection problems	For intralogistics navigation problems [204], shows the performance of RTLS [205], find the best location estimation algorithm [206]
Clustering	Grouping the database according to their similarities. Discovering similarities and dissimilarities between the data.	Partition based algorithms (e.g., K-means, fuzzy c-means) Hierarchical clustering (e.g., dendrogram) Density-based method Grid-based methods Model-based methods	Data segmentation (division into homogeneous sets) Identification of typical prototypes (e.g., simultaneous identification of time-homogeneous periods and their averages/trends)	Improve RTLS accuracy [213, 214, 215] Pedestrian motion learning [222]
Regression analysis	Identifying and analyzing the relationship between variables. Predicting and forecasting the process or dependent variables.	Multivariate linear regression Neural network Regression tree	Creating a model that predicts time (e.g., creating a model for predicting temperature)	Used to calculate the efficient RTLS [217] feeding behavior of cows [216]

2.4 Steps of setting up an RTLS for manufacturing support

An installation of RTLS is described in this section with a proposed workflow to illustrate the difficulties of RTLS projects. Figure 2.6 shows the necessary steps for RTLS-based process analysis. In general, the first step of an RTLS-based digitization project is the identification of requirements, where the physical area on the shop floor and the possible applications are defined. The next step is the installation of the sensor network. After the system is running and the position information is being successfully gathered, the accuracy of the system should be validated. The multi-tag concept substantially improves the object detection probability and makes the system more robust [223, 224]. Generally, the position engines of RTLS apply filtering methods to pre-process the position data (e.g., a Kalman-filter in the case of GPS) [225]. Several accuracy improvement solutions are available based on RTLS, including regression [226] and k-nearest neighbor classification [207].

With the spread of RTLS, position data pre-processing and cleaning methods have become an important research topic, based mainly on pedestrian dead reckoning [227] and wireless signal positioning methods [10]. The cleared and filtered position data provides more accurate information to the production system.

The integration of position data into the Manufacturing Execution System (MES) is a crucial element of the implementation.

To obtain usable data from the position information, zones of the manufacturing process should be defined. A zone (Station ID) represents a workstation or storage space, and the RTLS can obtain zone information from every tag in real-time. Figure 2.7 shows the connections among the RTLS, MES and production. At the beginning of production, the operator pairs a product with an RTLS tag (with a barcode scanner or manually at a PC) to identify the actual product ID in the system. The RTLS provides the zone information (based on the position data of tags and zones definition) with a timestamp to the MES with an application programming interface (API). The MES changes the status of the actual product ID based on the information from RTLS (e.g., the product is tested at a testing station). If a digital interface is provided at the workstation, then the MES can show the work instruction for the actual product or can set the optimal cycle time based on the product content [164].

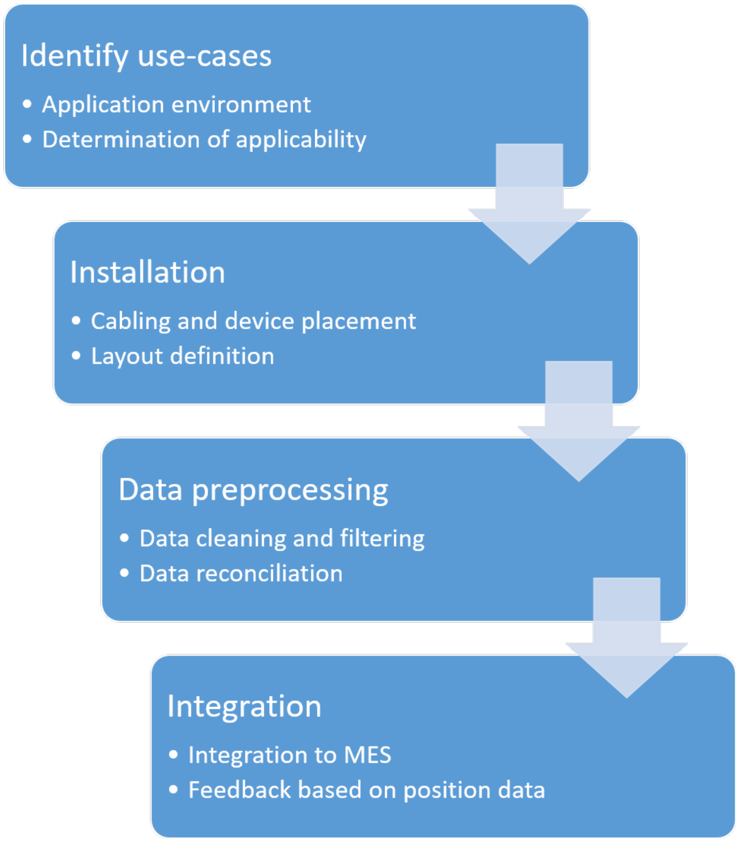


Figure 2.6: The full setup of RTLS in manufacturing. After the physical system installation, the layout and zone definition is necessary for system integration into the Manufacturing Execution System.

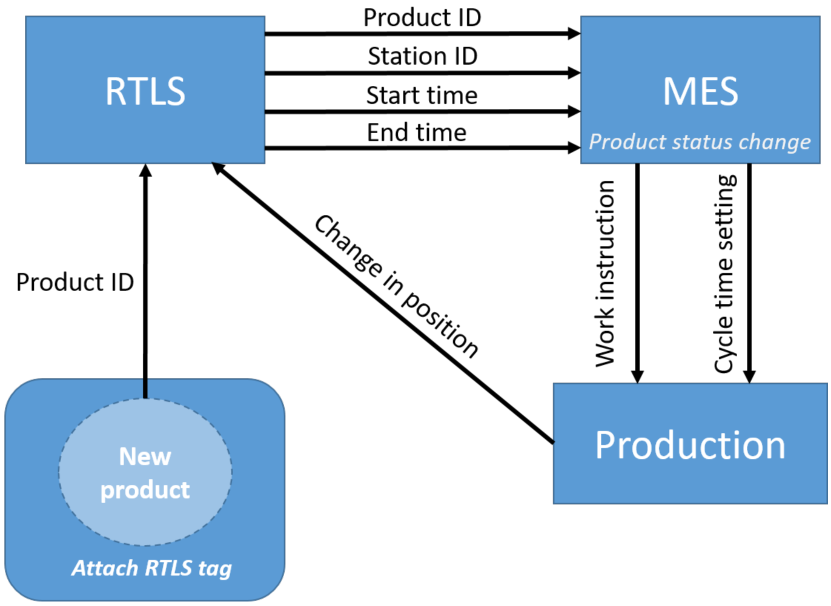


Figure 2.7: The real-time connection between the Manufacturing Execution System (MES) and production is available based on the RTLS.

2.5 Analysis based on position data - a case study

Position data-based production tracking has considerable potential to optimize production processes. In this section, a case study based on an implemented RTLS in a manufacturing environment is presented. The purpose of the position data in production is to transform to relevant information, in the interest of comparing the defined production zones with the position based clusters. Hidden information can be extracted from the position data for production management.

The use case presented is an anonymized example from a Tier 1 supplier company in the automotive sector. The production company used the Sunstone-RTLS Ltd. system, which is accurate to 50 cm with eight anchors per every 2000–3000 m^2. The system architecture is shown in Figure 2.8. There are seven workstations that are used to produce a small wire harness. The zones define the workstations, storage units and routes. The workstations are Tubing station I., Tubing station II., Channeling station, Test station, Screwdriver station, Packaging station and Quality check.

Thanks to the position data, the full traceability of products is available. The cycle time of workstations can be measured based on the classified zone data. The operator scans the product identification label at the first workstation, where the system paired it to the current tag ID (ID is also scanned at the station). The spent time of the actual p product (T_p^z) in the actual zone (z) is the difference between the last timestamp ($T_p^z(l)$ of the actual product (p) position data in the actual zone and the first one ($T_p^z(f)$).

The operator attaches the RTLS tag to the product at the first station (Tubing station I/II), and the final station is the Packaging.

The goal of the position-based zone identification is to determine the temporary storage at the temporary station area: a K-means algorithm is applied for position data classification. Figure 2.9 shows the classified position data where the algorithm

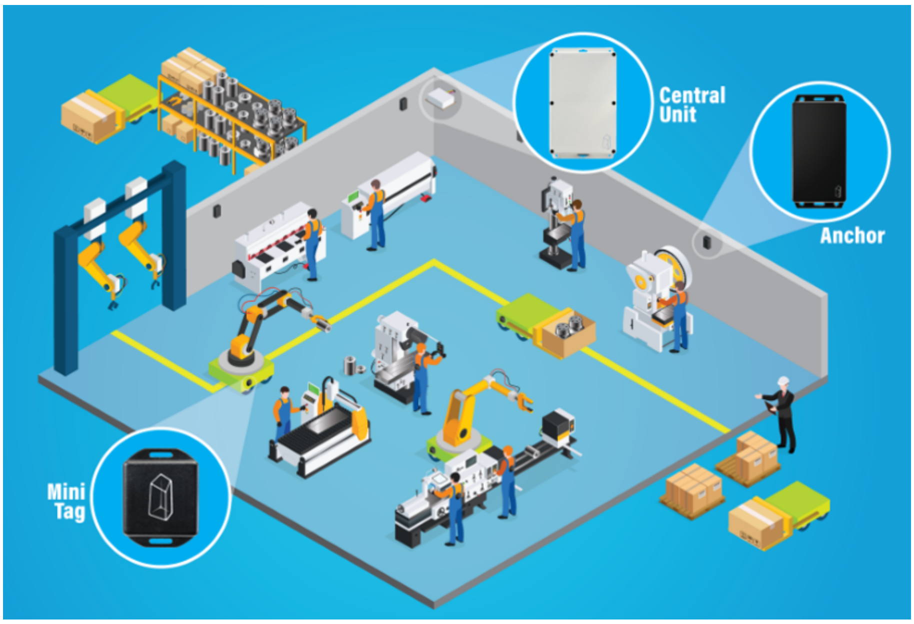


Figure 2.8: The infrastructure of the Sunstone-RTLS. Every central unit (CU) has eight anchors (which collect data from tags), and the CUs can be connected to create a cascade installation.

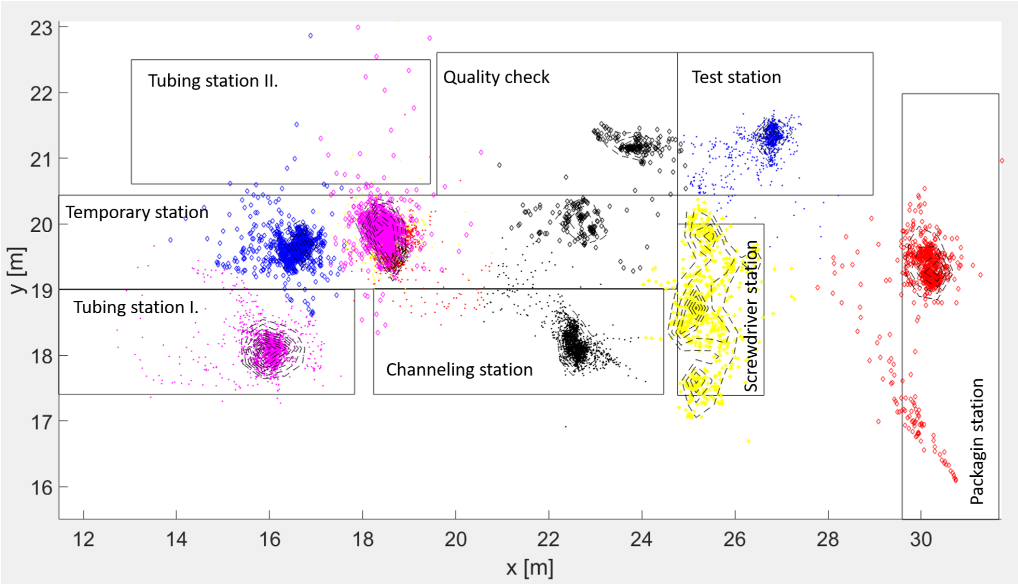


Figure 2.9: The production layout with seven pre-defined workstations. The classified and pre-defined (rectangles) zones are shown. The algorithm detects three small areas behind the Temporary station and indicates that Tubing station II was not used in this period.

detects the three small storage areas over the pre-defined zones. These three undefined or unplanned stations could be the cause of several losses.

Figure 2.10 shows the cycle time deviation of every workstation. The boxplot shows the distribution of the times related to the production of more than 150 products (in one shift). It can be observed that the packaging station has a shorter cycle time and the smallest deviation. This is because the real finish time at this station cannot be identified, as the tags traced by the RTLS are collected after production, and instead, the test label is scanned by the operator during packaging.

The figure shows the Tubing station I. is the bottleneck, but it can be seen in Figure 2.9, there is a second station (Tubing station II.), which is a spare workstation. In the current situation, there are not enough resources to operate both workstations, but it could be the solution to improve the process.

As this example highlighted, the RTLS can provide accurate and real-time information about the current status of the production process that could be utilised in the development of production processes.

2.6 Summary of the chapter

The purpose of this thesis was to provide a comprehensive overview of the application and development possibilities of RTLS in the manufacturing field. The overview of the solutions determined the value of the positioning data and specified which traceability technologies are suitable for real-time locating in different situations to ensure traceability. The research explored the possible applications in the production and logistics process. Finally, the implementation of RTLS and a data cleaning method are represented. The end of the Section 2 presented a case study, in which it was demonstrated what kind of information an RTLS system can provide.

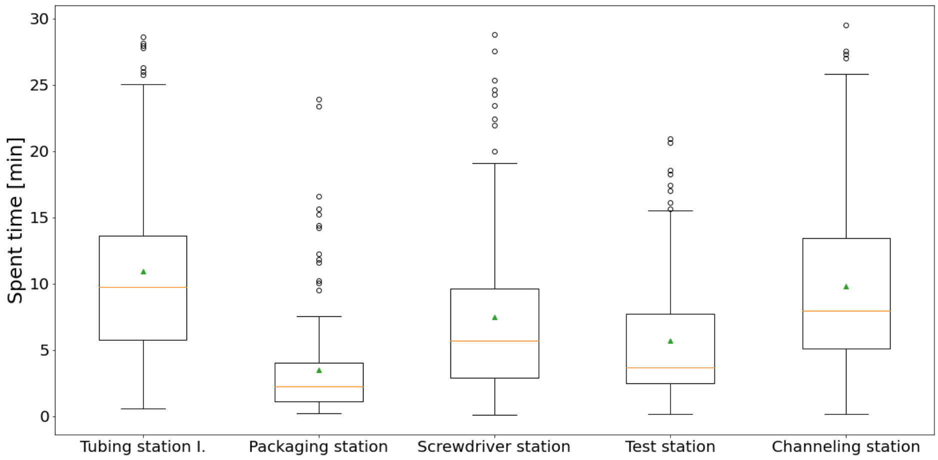


Figure 2.10: Boxplots of the cycle times measured at different zones of the production process. (Green triangles represent the averages, while red lines the medians).

The research pointed out that information extracted from RTLS is highly applicable for performance monitoring. Based on this fact, RTLS supported LEAN projects are very important research topics of the future. This study also introduced that machine learning and state estimation techniques are getting used more and more widely in the development of position data based models. Another conclusion of the case study is that typical states of the production process can be easily determined based on clustering algorithms. Analyzing the sequence of these means a significant increase in the understanding of the processes and in support of process models. It can be suggested that process mining will be the most relevant research topic in the future.

It is also important to highlight that an installed RTLS makes the integration of more sensor data available, and in this way the quick implementation of IoT solutions. This opportunity is seen as particularly beneficial for the development of existing processes, such as the introduction of brownfield Industry 4.0 solutions.

The advantage of these solutions can be exploited well if the production system is supported by a Manufacturing Execution System (MES), in which a system ensures that information derived from position data can be used in production process optimizations. In line with this, an RTLS project should be connected with an MES development. This MES development process can be supported by semantic models, which are helping to structure sensor and production data. The application of these models is also a research and development topic for the future.

Chapter 3

Enhancing position data accuracy in RTLS: A data reconciliation approach for improved process insights

3.1 Introduction to position data accuracy enhancement through data reconciliation

Data reconciliation based on indoor positioning data using map-based error refinement is a novel approach characterised by its ability to reconcile disparate data sources and improve accuracy. The equations of equilibrium can be further extended depending on the specific solution. While the map-based error evolves dynamically, the current method only calculates an average error for the designated area, so it may be beneficial to consider a weighted average for the error map to allow for adaptability. Such adaptation involves the prioritisation new data, allowing a faster response to changes in the production or logistics environment. Integration into production environments requires consideration of factors such as industrial integrability, scalability, data security, reliability and resilience of the algorithm, and user acceptance. The advantage of the method is that it can be easily applied to the data refinement provided by indoor positioning systems. The data in the error map is constantly updated using weighted error averaging, making the method flexible to changes in the production or logistics environment.

The contributions of this work are the following:

- Definition of tags associated with resources and bring the data to a common time base by interpolation.
- Using physical and logical relationships between sensor tags to determine the measurement error from historical IPS data.
- Classification of the error average of the position data based on the map.
- Application of the data reconciliation method to refine the real-time position data based on the error map.

According to these contributions, the Section 3 is structured as follows: The next section presents the preparation of position data provided by the IPS and the estimation of map-based measurement errors. After that, the position data refined based on the data reconciliation method is shown. Finally, the applicability of the methodology is illustrated through a case study of forklift position data in a warehouse environment.

3.2 Methodology - enhancing position accuracy in RTLS: a data reconciliation approach

A detailed methodological overview is provided in this section based on the steps illustrated in Figure 3.1. The first step is to create the relationship matrix between sensors and resources and to determine the physical distances between sensors within the same resource. Due to the lack of synchronisation in the data recording frequency between individual tags, it is necessary to interpolate the coordinates over time. Therefore, the unique recording times for the position data of the sensors associated with each resource are identified for interpolation. The second step in dealing with missing positional data is to apply interpolation techniques to fill the gaps for all unique recording times across all resource-related sensors, resulting in a comprehensive positional data matrix. With this matrix in hand, the third step is to calculate the distances between all sensors within the same resource group based on their positional data. Discrepancies between calculated distances and actual physical distances are quantified by generating an error matrix through subtractive analysis, which is the fourth step. As described in the fifth step, these errors are then spatially distributed on a map by averaging the discrepancies to effectively distribute them based on measured coordinates. Finally, in the sixth step, a data reconciliation algorithm is implemented to refine the measured position data, effectively correcting inaccuracies due to the layout-based errors identified in the previous steps. This iterative process ensures the accuracy and reliability of the final position data obtained from the sensor network.

In the context of resource localisation, multiple tags are used for each $O = \{o_1, o_2, \ldots, o_{N_o}\}$ set of objects, where N_o represents the number of objects, and

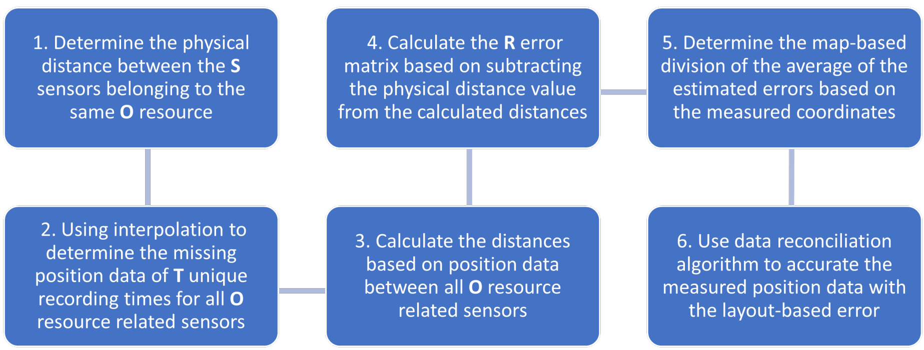


Figure 3.1: Process steps of the applied methodology.

$S = \{s_1, s_2, \ldots, s_{N_s}\}$ denotes the set of sensor tags, where N_s indicates the number of tags. Leveraging the unique identification numbers associated with the tags, the relationship between the tags and their respective objects can be established: $o_j \rightarrow \{s_1, s_2, \ldots, s_{N_l}\} \subseteq S$, where $N_l^{(o_j)}$ is the count of tags assigned to o_j object. The $\mathbf{A}$ matrix contains all the relationships between the S sensors and the O objects, where 1 means the S sensor belongs to the O object and 0 means the S sensor does not belong to the O object:

$$\mathbf{A} = \left[\begin{array}{c|cccccc} & o_1 & o_2 & \cdots & o_j & \cdots & o_{N_o} \\ \hline s_1 & 0 & 0 & \cdots & 1 & \cdots & 0 \\ s_2 & 0 & 0 & \cdots & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ s_{N_s} & 0 & 0 & \cdots & 1 & \cdots & 0 \end{array}\right] \tag{3.1}$$

If it is known which sensor tags belong to an object, the relationship between these sensors within the object can be defined, for example the physical distance between them. The $\mathbf{D}$ matrix contains this information:

$$\mathbf{D} = \left[\begin{array}{c|cccc} & s_1 & s_2 & \cdots & s_{N_s} \\ \hline s_1 & \text{N/A} & d_{1,2} & \cdots & d_{1,N_s} \\ s_2 & d_{2,1} & \text{N/A} & \cdots & d_{2,N_s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{N_s} & d_{N_s,1} & d_{N_s,2} & \cdots & \text{N/A} \end{array}\right] \tag{3.2}$$

$$D_{S_{N_s-1},S_{N_s}} = \begin{cases} d_{N_s-1,N_s} & \text{is the distance between } s_{N_s-1} \text{ and } s_{N_s} \\ & \text{if the sensors belong to the same object,} \\ \text{N/A} & \text{if the sensors do not belong to the same object.} \end{cases} \tag{3.3}$$

The sensors are situated at a consistent separation from each other. Periodically, these tags transmit positional data, representing the 3D spatial coordinates for indoor positioning. Typically, this raw data is stored in a large $\mathbf{P^l} \subseteq \mathbf{T^l} \times \mathbf{X^l}$ database, where $\mathbf{T^l} = \{t_1^l, t_2^l, \ldots, t_{N_d}^l\}$ are the timestamps and $\mathbf{X^l} = \{\mathbf{x_1^l}, \mathbf{x_2^l}, \ldots, \mathbf{x_k^l}, \ldots, \mathbf{x_{N_d}^l}\}$ where $\mathbf{x_k^l} = [x_{k,1}^l, x_{k,2}^l, x_{k,3}^l] \in \mathbb{R}^3$ are the coordinate-based position data at the k-th time of the l-th sensor. N_d^l represents the number of the recorded position data of the l-th sensor. Inaccurate data may occur, such as discrepancies relative to coordinates aligned with the actual production environment, or simply missing data. It is crucial to identify and filter out these occurrences at the outset of our process. The data combined per resource may differ over time; therefore, an initial crucial aspect in the methodology description is the synchronization of timing data, achieved by linearly interpolating the position data. This interpolation yields pairs of position data points, allowing for the computation of the distance between the tags, accounting for errors. The goal is to determine the set of unique times and identify the missing time points in both databases. This process entails linearly connecting temporal data points during interpolation, thereby estimating them at the desired time points. Therefore, at all the $\mathbf{T} = [t_1, t_2, \ldots, t_g, \ldots, t_N]$ time points, the $\mathbf{X^l}$ coordinates are available in the $\mathbf{X}^{(l_{\text{interp}})}$ matrix, where $\mathbf{x}_g^l \in \mathbf{X}^{(l_{\text{interp}})}$ is the interpolated position data vector of the l-th sensor at the t_g time point, $\mathbf{x}_{(g-1)}^l$ and $\mathbf{x}_{(g+1)}^l$ are the nearest available measured data points of the l-th sensor before and after interpolation at the t_g time point.

$$\mathbf{X}^{(l_{\text{interp}})} = \begin{bmatrix} \mathbf{x}_1^l \\ \mathbf{x}_2^l \\ \vdots \\ \mathbf{x}_g^l \\ \vdots \\ \mathbf{x}_N^l \end{bmatrix} = \begin{bmatrix} x_{1,1}^l & x_{1,2}^l & x_{1,3}^l \\ x_{2,1}^l & x_{2,2}^l & x_{2,3}^l \\ \vdots & \vdots & \vdots \\ x_{g,1}^l & x_{g,2}^l & x_{g,3}^l \\ \vdots & \vdots & \vdots \\ x_{N,1}^l & x_{N,2}^l & x_{N,3}^l \end{bmatrix} \tag{3.4}$$

where, l_{interp} means the l-th sensor with the interpolated data and N is the number of all unique timestamps with available position data. It is important to note that for starting and ending data, where there are no intermediate values for interpolation, the missing values can only be determined by extrapolation. Since a consistent solution regarding the current position data may not be achieved, these values are ignored during further data processing. The interpolation for a position is the following:

$$\mathbf{x}_g^l = \mathbf{x}_{(g-1)}^l + \left(t_g - t_{(g-1)}\right) \cdot \frac{\left(\mathbf{x}_{(g+1)}^l - \mathbf{x}_{(g-1)}^l\right)}{\left(t_{(g+1)} - t_{(g-1)}\right)} \tag{3.5}$$

The $\mathbf{X}_{\text{all}}$ matrix containing the position data of all sensors can be written as follows:

$$\mathbf{X}_{\text{all}} = \begin{bmatrix} \mathbf{x}_1^1 & \mathbf{x}_1^2 & \cdots & \mathbf{x}_1^{N_s} \\ \mathbf{x}_2^1 & \mathbf{x}_2^2 & \cdots & \mathbf{x}_2^{N_s} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_N^1 & \mathbf{x}_N^2 & \cdots & \mathbf{x}_N^{N_s} \end{bmatrix} \tag{3.6}$$

Based on the $\mathbf{A}$ matrix, the associated sensor position data can be determined per object in the $\mathbf{X}_{\text{all}}^{(\mathbf{o_j})}$ tuple:

$$\mathbf{X}_{\text{all}}^{(\mathbf{o}_j)} = \mathbf{X}_{\text{all}} \times \mathbf{A}(:, o_j) = \begin{bmatrix} \mathbf{x}_1^1 & \mathbf{x}_1^2 & \cdots & \mathbf{x}_1^{\mathbf{N}_s} \\ \mathbf{x}_2^1 & \mathbf{x}_2^2 & \cdots & \mathbf{x}_2^{\mathbf{N}_s} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_\mathbf{N}^1 & \mathbf{x}_\mathbf{N}^2 & \cdots & \mathbf{x}_\mathbf{N}^{\mathbf{N}_s} \end{bmatrix} \times \mathbf{A}(:, o_j) = \begin{bmatrix} \mathbf{x}_1^{(\mathbf{o}_j)} \\ \mathbf{x}_2^{(\mathbf{o}_j)} \\ \vdots \\ \mathbf{x}_\mathbf{N}^{(\mathbf{o}_j)} \end{bmatrix} \tag{3.7}$$

where $\mathbf{x}_\mathbf{n}^{(\mathbf{o_j})} \subseteq A(:, o_j)$ is an $N_s \times 3$ matrix containing the position data of sensors belonging to the o_j object:

$$\mathbf{x}_n^{(\mathbf{o}_j)} = \begin{bmatrix} \mathbf{x}_{n_1}^{(\mathbf{o}_j)} \\ \mathbf{x}_{n_2}^{(\mathbf{o}_j)} \\ \vdots \\ \mathbf{x}_{n_{N_s}}^{(\mathbf{o}_j)} \end{bmatrix} = \begin{bmatrix} x_{n_{1,1}}^{(o_j)} & x_{n_{1,2}}^{(o_j)} & x_{n_{1,3}}^{(o_j)} \\ x_{n_{2,1}}^{(o_j)} & x_{n_{2,2}}^{(o_j)} & x_{n_{2,3}}^{(o_j)} \\ \vdots & \vdots & \vdots \\ x_{n_{N_s,1}}^{(o_j)} & x_{n_{N_s,2}}^{(o_j)} & x_{n_{N_s,3}}^{(o_j)} \end{bmatrix} \tag{3.8}$$

The physical distances between the related sensors are defined in the $\mathbf{D}$ matrix, and the $\mathbf{D}_\mathbf{n}^{(\mathbf{o_j})}$ distance matrix can also be calculated between sensors from the measured data. Ideally, the two distances should be the same, but due to measurement inaccuracy, an r error can be calculated as follows:

$$\mathbf{r}_n^{(\mathbf{o}_j)} = \sum_{g=1}^{N_s} \sum_{f=1}^{N_s} \left| \mathbf{D}_{f,g}^{(\mathbf{o}_j)} - \mathbf{D}_{f,g} \right| \tag{3.9}$$

$$\mathbf{r}_n^{(\mathbf{o}_j)} = \sum_{g=1}^{N_s} \sum_{f=1}^{N_s} \left| \left(\left| \mathbf{x}_{n_g}^{(\mathbf{o}_j)} - \mathbf{x}_{n_f}^{(\mathbf{o}_j)} \right| - \mathbf{D}_{f,g} \right) \right| \tag{3.10}$$

$$\mathbf{r}_n^{(\mathbf{o}_j)} = \left| \begin{bmatrix} d_{n_1,1}^{(o_j)} & d_{n_1,2}^{(o_j)} & \cdots & d_{n_1,g}^{(o_j)} \\ d_{n_2,1}^{(o_j)} & d_{n_2,2}^{(o_j)} & \cdots & d_{n_2,g}^{(o_j)} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n_f,1}^{(o_j)} & d_{n_f,2}^{(o_j)} & \cdots & d_{n_f,g}^{(o_j)} \end{bmatrix} - \begin{bmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,g} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,g} \\ \vdots & \vdots & \ddots & \vdots \\ d_{f,1} & d_{f,2} & \cdots & d_{f,g} \end{bmatrix} \right| \tag{3.11}$$

If all the errors are calculated for all O objects based on $\mathbf{X}_{\text{all}}$, a matrix $\mathbf{R}_{\text{all}}$ is obtained that contains all the errors:

$$\mathbf{R}_{\text{all}} = \begin{bmatrix} \mathbf{r}_1^1 & \mathbf{r}_1^2 & \cdots & \mathbf{r}_1^{\mathbf{N}_o} \\ \mathbf{r}_2^1 & \mathbf{r}_2^2 & \cdots & \mathbf{r}_2^{\mathbf{N}_o} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{r}_{\mathbf{N}}^1 & \mathbf{r}_{\mathbf{N}}^2 & \cdots & \mathbf{r}_{\mathbf{N}}^{\mathbf{N}_o} \end{bmatrix} \tag{3.12}$$

The measured position data from $\mathbf{X}_{\text{all}}$ with the calculated errors from $\mathbf{R}_{\text{all}}$ are available based on the objects and through the accumulation of a substantial volume of measurement data, the prospect of constructing $\mathbf{R}_{\mathbf{x}}$ error cartography matrix is the following:

$$\mathbf{R}_x = \frac{1}{N_{(p,q)}} \sum_{j=1}^{N_o} \sum_{i=1}^{N} \mathbf{r}_i^j \,|\, \mathbf{x}_i^j = \begin{bmatrix} r_{(x_{1,1})} & r_{(x_{1,2})} & \cdots & r_{(x_{1,q})} \\ r_{(x_{2,1})} & r_{(x_{2,2})} & \cdots & r_{(x_{2,q})} \\ \vdots & \vdots & \ddots & \vdots \\ r_{(x_{p,1})} & r_{(x_{p,2})} & \cdots & r_{(x_{p,q})} \end{bmatrix} \tag{3.13}$$

where p is the x-axis, q is the y-axis layout-based boundary, and $N_{(p,q)}$ represents the numbers of the calculated error for every $\mathbf{x}_{(\mathbf{p},\mathbf{q})}$ position. In this case, the position data are known and the related error, so one possible method to improve the accuracy of the position data is data reconciliation.

$$\begin{aligned} \text{minimizing } & \mathbf{J}(\hat{\mathbf{x}}_k^{(\mathbf{o}_j)}) = (\mathbf{x}_k^{(\mathbf{o}_j)} - \hat{\mathbf{x}}_k^{(\mathbf{o}_j)})^T \mathbf{V}_{\mathbf{x}_k}^{-1} (\mathbf{x}_k^{(\mathbf{o}_j)} - \hat{\mathbf{x}}_k^{(\mathbf{o}_j)}) \\ \text{subject to } & f(\mathbf{x}_k^{(\mathbf{o}_j)}) = 0 \end{aligned} \tag{3.14}$$

where,

- $\mathbf{x}_\mathbf{k}^{(\mathbf{o_j})}$ is the $M \times 1$ vector with the measured position data pairs: our case, $\mathbf{x}_\mathbf{k}^{(\mathbf{o_j})} \in \mathbb{R}^3$, therefore $M = 3$;
- $\hat{\mathbf{x}}_k^{(\mathbf{o}_j)}$ is the $M \times 1$ vector with the reconciled position data pairs: in our case, $\hat{\mathbf{x}}_k^{(\mathbf{o}_j)} \in \mathbb{R}^3$;
- $\mathbf{V}$ is the $M \times M$ covariance matrix of the measurement, which is calculated based on the $r_{(x_{(p,q)})} \in \mathbf{R}_{\mathbf{x}}$ error cartography;

- f is a $C \times 1$ vector, which describes the functional form of the model's equality constraints, for example, the physical distance constraint equation.

In our case, the $\mathbf{J}(\hat{\mathbf{x}})$ objective function is the following:

$$\mathbf{J}(\hat{\mathbf{x}}) = \left(\begin{bmatrix} x^{(o_j)}_{(k,1)} \\ x^{(o_j)}_{(k,2)} \\ x^{(o_j)}_{(k,3)} \end{bmatrix} - \begin{bmatrix} \hat{x}^{(o_j)}_{(k,1)} \\ \hat{x}^{(o_j)}_{(k,2)} \\ \hat{x}^{(o_j)}_{(k,3)} \end{bmatrix} \right)^T \mathbf{V}^{-1}_{(\mathbf{x}_k)} \left(\begin{bmatrix} x^{(o_j)}_{(k,1)} \\ x^{(o_j)}_{(k,2)} \\ x^{(o_j)}_{(k,3)} \end{bmatrix} - \begin{bmatrix} \hat{x}^{(o_j)}_{(k,1)} \\ \hat{x}^{(o_j)}_{(k,2)} \\ \hat{x}^{(o_j)}_{(k,3)} \end{bmatrix} \right) \tag{3.15}$$

To determine the $\mathbf{V}^{-1}_{(\mathbf{x_k})}$ covariance matrix, the error values are needed from the $\mathbf{R_x}$ error cartography matrix based on the two $\mathbf{x}^{(o_j)}_{(k_1)}$ and $\mathbf{x}^{(o_j)}_{(k_2)}$ position data of the o_j object's relevant sensors:

$$\mathbf{V}^{-1}_{(\mathbf{x}_k)} = \begin{bmatrix} \mathbf{r}^{(k_1)}_{(\mathbf{x}_{(p,q)})} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{r}^{(k_1)}_{(\mathbf{x}_{(p,q)})} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{r}^{(k_1)}_{(\mathbf{x}_{(p,q)})} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{r}^{(k_2)}_{(\mathbf{x}_{(p,q)})} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{r}^{(k_2)}_{(\mathbf{x}_{(p,q)})} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{r}^{(k_2)}_{(\mathbf{x}_{(p,q)})} \end{bmatrix} \tag{3.16}$$

The equality of the f cost function can be the mentioned distance-based constraint as the minimizing basis of the $\mathbf{J}(\hat{\mathbf{x}}^{(\mathbf{o_j})}_{\mathbf{k}})$ objective function:

$$f_{\min} \left\{ \|\mathbf{x}^{(o_j)}_{(k_1)} - \mathbf{x}^{(o_j)}_{(k_2)}\| - \mathbf{D}(o_j) \right\} = f_{\min} \left\{ d^{(o_j)}_{(x_k)} - d_{(o_j)} \right\} = 0 \tag{3.17}$$

By using the reconciled position data pairs obtained in this manner, more precise real-time indoor positioning can be achieved. In the following section, the applicability of the developed method is demonstrated through a warehouse case study.

3.3 Improving real-time forklift tracking with data reconciliation - a case study

The use case originates from a real logistics environment, where forklifts are tracked by a UWB-based IPS during the storage process. The forklifts are equipped with two tags (as seen in Figure 3.2). These tags continuously transmit real-time position data every three seconds during movement.

Five forklifts are being monitored, and each forklift is equipped with two sensors. The IPS sensor tag I is fixed in position on the forklift cabin, while the IPS sensor tag II moves with the fork in the Z-direction. As a result, only the X-Y coordinates were used to calculate the distance, and the relative distance between the two sensors was fixed in these two dimensions at 1.7 meters in our case. The $\mathbf{A}$ matrix defines the tags paired with the forklifts.

$$\mathbf{A} = \begin{array}{c} \\ s_{4968} \\ s_{4967} \\ s_{4966} \\ s_{4965} \\ s_{4957} \\ s_{5294} \\ s_{5378} \\ s_{5302} \\ s_{5298} \\ s_{5301} \end{array} \begin{array}{c} \begin{array}{ccccc} o_1 & o_2 & o_3 & o_4 & o_5 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \tag{3.18}$$

A total of ten sensor data points are summarised in Table 3.1. The number of position data available for each sensor before and after filtering (data is filtered out when the forklift is not active), and after interpolation. The data recording interval of the sensors becomes less frequent, at 5 seconds in our case, if no movement is detected, and these downtimes are filtered out from the position data. The interpolation is performed in the same manner as described in Equations 3.4 and 3.5. The recording time data of the sensor tag pairs is aggregated, unique values are determined, and interpolation is then carried out based on the position data from the time points before and after the sensor for which data at the given time is missing. In the methodological description, it was also highlighted that if extrapolation is required at the beginning or end of the measurement data, those data are not taken into account.

The project aims to identify the areas within the warehouse where forklifts are operating. The movement of the tag placed on the fork in the Z-direction indicates that an order has been fulfilled. The position data is then used to determine the location in the warehouse where the work has been completed. This approach allows us to assess the efficiency of the logistics service and identify potential opportunities for improvement. Figure 3.3 identifies the movement of O_4 forklift based on the S_{5298}

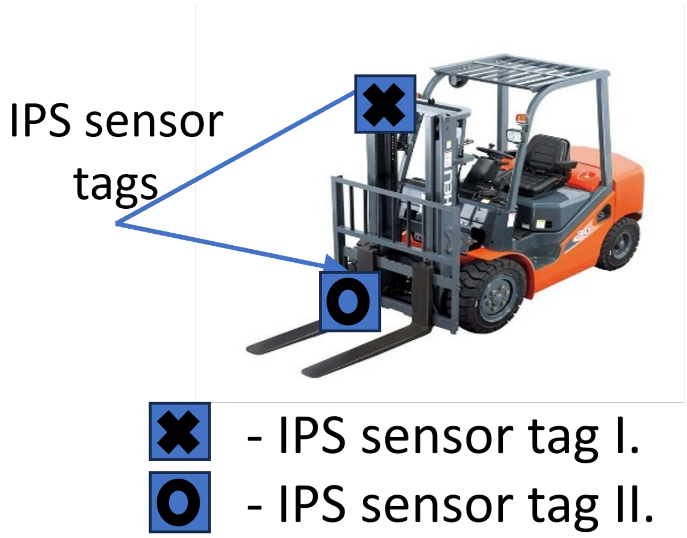


Figure 3.2: IPS sensors placed on the forklift.

Sensor	Number of measured data	Number of measured data after filtering	Number of data after interpolation
s_{4968} (fix)	63773	62417	119501
s_{5294} (fork)	58351	57086	119501
s_{4967} (fix)	66226	64581	133402
s_{5378} (fork)	70628	68823	133402
s_{4966} (fix)	54175	52749	117421
s_{5302} (fork)	66034	64673	117421
s_{4965} (fix)	45535	43969	100496
s_{5298} (fork)	57794	56528	100496
s_{4957} (fix)	27451	26977	53313
s_{5301} (fork)	26854	26366	53313

Table 3.1: Number of the available position data for every sensor before and after filtering, and after interpolation.

sensor data as blue markings and the layout represents a warehouse environment where material flow is managed using the aforementioned forklifts. This includes storage operations from the delivery zone to the high warehouse, as well as the reverse process as picking out.

However, accurate position data is essential, so it is important to enhance the accuracy of the IPS-based position data. To achieve this, the developed data reconciliation-based algorithm is being utilized. The algorithm requires an estimated error, which is determined by comparing the physical positions of the tags placed on the forklifts with the position data-based positions on a layout basis.

The method is demonstrated on randomly selected O_4 forklift with S_{4965} and S_{5298} pair of sensors. As discussed earlier, the physical distance between the related tags is consistently 1.7 meters in each case. By employing time-based synchronization and linear interpolation with the two tags, and then subtracting the physical distance from the theoretical distance calculated from the coordinate data pairs obtained in this manner, the estimated error can be derived. Subtracting the physical distance value from the calculated distances gives the error distribution, as shown in Figure 3.4.

The Figure 3.5 heatmap is based on the measured absolute average distances. Based on the methodology presented in the previous section, it is worth examining the layout-based distribution of the calculated error. After collecting a sufficient number of measurements, in our case minimum fifty pieces of data from each divided

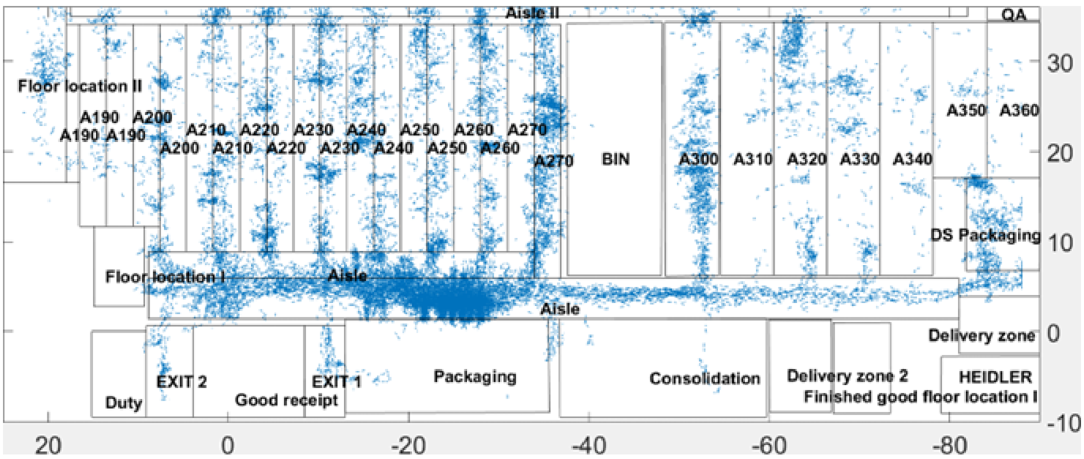


Figure 3.3: Indoor positioning data-based movement of forklift in a warehouse environment. Sufficiently accurate location data is important for identifying the logistics area.

area part (multiple position data from multiple forklifts), the average errors can be calculated as absolute values of the difference between the actual physical distance and the distance calculated from the position data. If there is not enough data for the given area, the average error calculated for the warehouse environment is used. These average errors can then be displayed on the layout as a heatmap, based on the position data, using the R_x error cartography matrix.

The diagram shows the division of the logistics area according to x and y coordinates, and the colour scale on the right indicates the degree of error in metres (the darker blue indicates insufficient data, while lighter shades represent errors between 1 and 2 meters, and yellow indicates errors exceeding 7 meters in the given area). Figure 3.5 can be reconciled with the layout shown in Figure 3.3, revealing identical coordinate distributions along the x and y axes. There is also a notable escalation of error in the high warehouse, a phenomenon that is understandable given the inherent challenges that such environments pose for Indoor Positioning Systems (IPS).

The map-based heatmap characterizes the storage environment by measuring errors based on the defined resolution of the layout. This noise is typically caused by environmental factors such as a metal shelving system or walls. Based on this, the average error of the current position data within the specified section of the storage environment can be determined. Using the layout-based error matrix obtained in this manner, the estimated error considered in the data reconciliation calculations can be enhanced. The error map is used to determine the estimated error associated with the position data that needs to be refined. These data will constitute the elements of the V matrix. Using the previously mentioned physical distance, the data is optimized, and the reconstructed (refined) position data is obtained.

A sample set of this data is presented in Table 3.2. The first column presents measurement data, while the second column indicates the corresponding layout-based error, which is crucial for the data matching algorithm. The third and fourth columns show the distances calculated from the coordinates and the physical distances

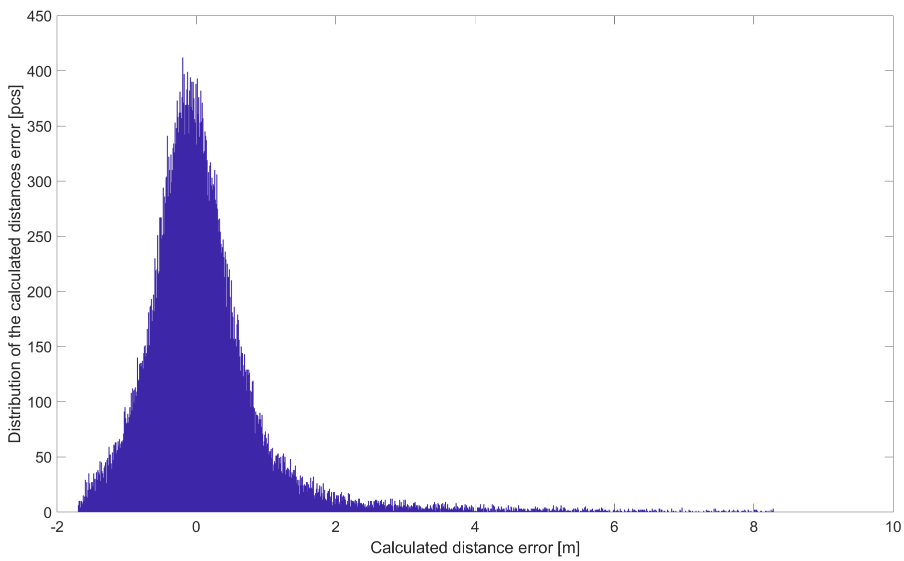


Figure 3.4: Histogram of the distances error belonging to the O_4 forklift.

between the tags respectively. The fifth column shows the distance derived from the difference between these distances. It's worth noting that certain pairs of data have errors that exceed the average. The error determined from new data points is treated in the same way in the map-based error calculation, forming a constantly evolving database. Consequently, a larger volume of measurement data allows for more accurate estimates. The penultimate column shows the coordinates after data matching, alongside the recalculated distances, which match the actual physical distances.

The organized nature of the positions is evident in the fact that the positions of the two components remain stable even during turns, contributing to improved trackability. The results help to obtain a more accurate representation of the paths taken by the forklifts. This is shown in Figure 3.6, with a forklift path segment based on the measured (left) and refined (right) position data. The red connection shows the temporally related data and therefore the distance based on the measurement data, and the green connection shows the related reconciliation data, showing that the algorithm has refined the position data to the actual physical distance. The more accurate position data can also be observed visually, as the O_4 forklift's rotation is clearly visible. Furthermore, this approach has the potential to enhance the precision of indoor positioning data, making it applicable to solving the clustering issue mentioned earlier (Figure 3.3). By improving the accuracy of spatial information, our approach sets the foundation for more effective monitoring and operational optimization in warehouse environments.

The data reconciliation method provides realistic positions in terms of distance, resulting in a simplification of the error histogram shown in Figure 3.4 to zero error. Based on this, a map-based index was developed to indicate the improvement in data usability. To achieve this, a velocity-based metric was created based on the displacement of two sensors over time. From this, two velocity values are derived that

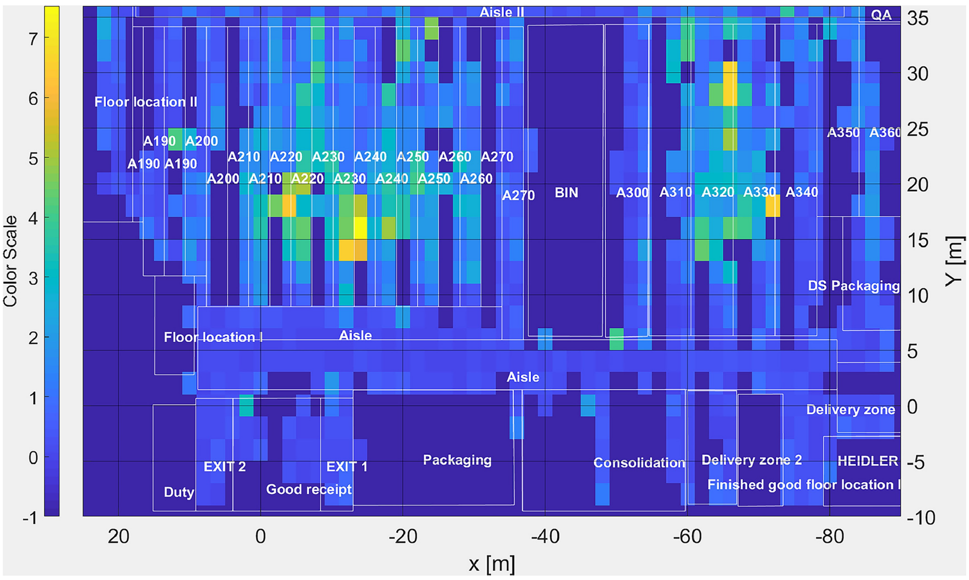


Figure 3.5: The warehouse error heatmap of the measured error based on the average distance between tag pairs [m]. If the number of measurements per given resolution does not reach a limit (in this case 50) in the given area, then -1 value is used on the layout and the average error for data reconciliation.

$x_k^{(o_4)}$ measured position data pairs [m]	$r_{(x_k^{(o_4)})}$ layout-based error [m]	$d_{(x_k)}^{(o_4)}$ Calcu-lated dis tance [m]	$d_{(o_4)}$ Physical Dis tance [m]	dis tance devi ation [m]	$\hat{x}_k^{(o_4)}$ reconciled position data pairs [m]	Calcu-lated distance after recon-cili-ation [m]
7.57; -1.53 7.49; -2.42	0.3935; 0.1000	0.885	1.7	—0.815—	7.63; -0.89 7.48; -2.58	1.7
7.84; -1.72 7.26; -2.40	0.3935; 0.1000	0.896	1.7	—0.804—	8.25; -1.22 7.16; -2.53	1.7
8.06; -1.80 7.25; -2.41	0.3935; 0.1000	0.972	1.7	—0.728—	8.55; -1.55 7.13; -2.49	1.7
8.46; -2.02 6.62; -2.66	0.1000; 0.1000	1.948	1.7	—0.248—	8.27; -2.08 6.67; -2.65	1.7
8.42; -2.57 6.60; -2.67	0.1000; 0.1000	1.822	1.7	—0.122—	8.36; -2.57 6.66; -2.67	1.7
8.32; -2.73 6.58; -2.68	0.1000; 0.1000	1.747	1.7	—0.047—	8.30; -2.73 6.60; -2.68	1.7

Table 3.2: Summary table of data reconciliation results with some example data of O_4 forklift with S_{4965} and S_{5298} (sensor position data, layout-based error, calculated distance, physical distance, distance deviations, reconciled position data and calculated distance after reconciliation).

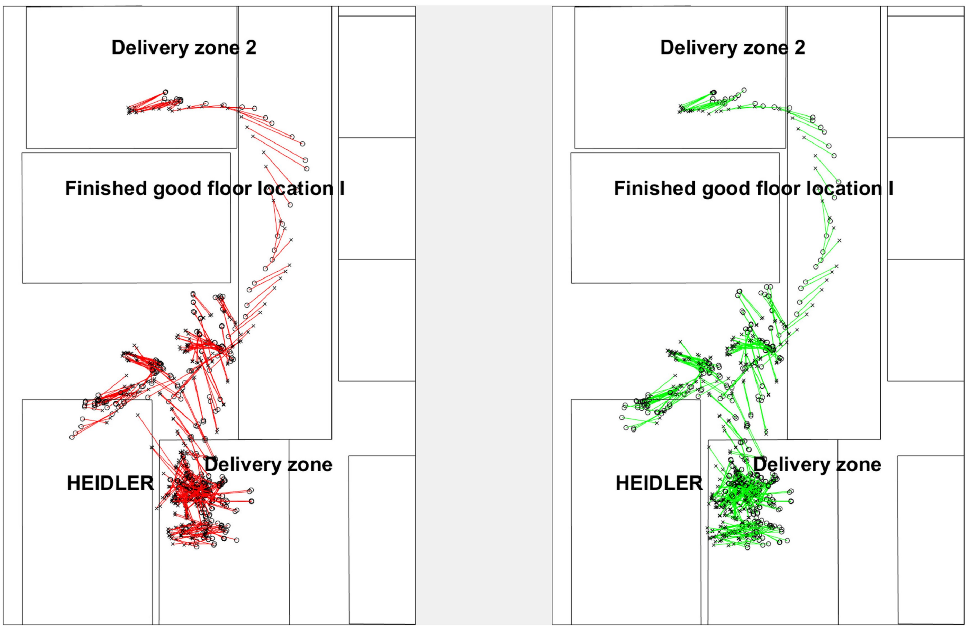


Figure 3.6: Forklift route segment based on measured and reconciliated position data of O_4 forklift with S_4965 and S_5298 sensor.

must not exceed the physical constraints of the forklift, namely below 0.4 and above 5 km/h. These unacceptable velocity values were identified, and their proportions were plotted on the map shown in Figure 3.7 and Figure 3.8. In terms of applying and validating the method, it was recognized that the system does not reliably correct unrealistically erroneous recorded position data with the currently implemented mass balance equation. It should be noted that the application of the method in the case study demonstrates its applicability, but in solving real-world problems, efforts should be made to solve as many mass balance equations as possible. In this case, this was not feasible, so a simple outlier detection method was applied based on these erroneous values. For the original tag distances (Figure 3.4), position data pairs where the calculated distance falls outside the range of 0.5 to 3 m were filtered out.

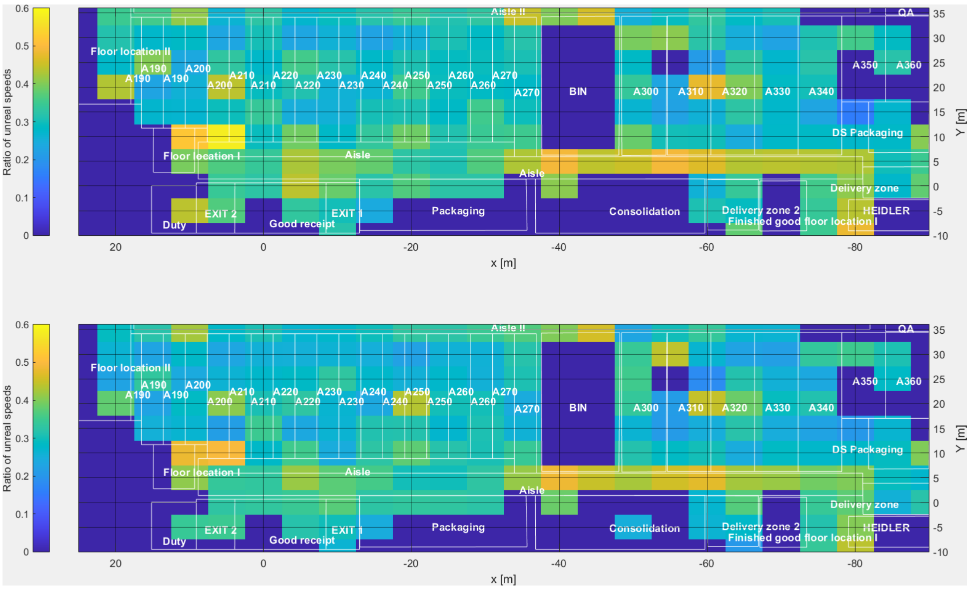


Figure 3.7: Map based ratio of unreal speeds of S_4965 tag. The top graph shows the distribution of velocity data obtained from the original position data, while the bottom graph shows the distribution of velocity data obtained after data matching. If the number of calcilated speed data per given resolution does not reach a limit - which is 50 in this case - in the given area, a value of 0 is used on the layout.

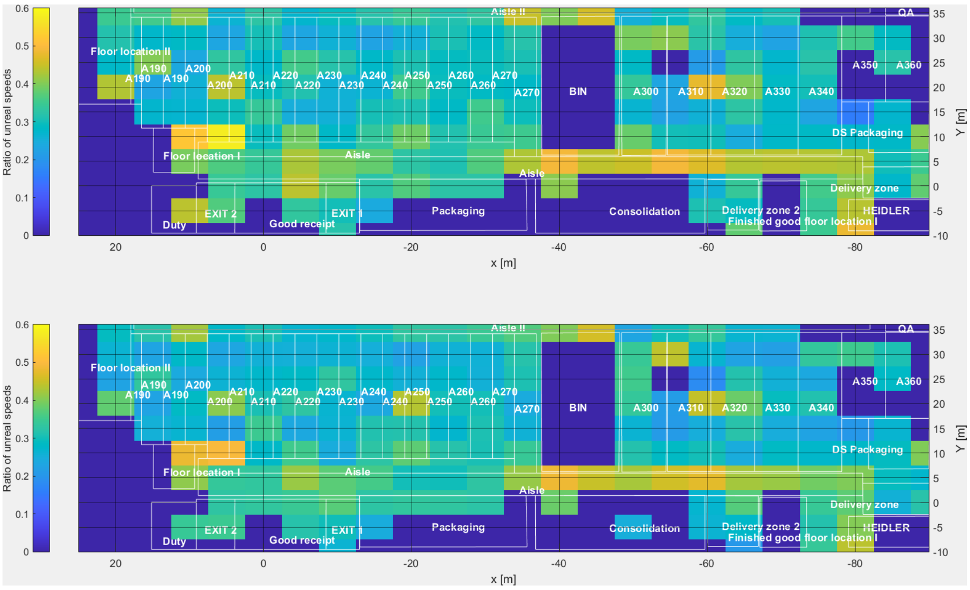


Figure 3.8: Map based ratio of unreal speeds of $S_5$298 tag. The top graph shows the distribution of velocity data obtained from the original position data, while the bottom graph shows the distribution of velocity data obtained after data matching. If the number of calcilated speed data per given resolution does not reach a limit - which is 50 in this case - in the given area, a value of 0 is used on the layout.

3.4 Summary of the chapter

The goal of this thesis is to enhance the accuracy of indoor positioning data by employing the data reconciliation method to map the layout-induced noises in the production and logistics environment. The method is demonstrated through a case study in which position data is obtained by attaching two tags to the resources. By utilising the physical distance between the tags, an estimation of measurement error is derived through distance calculation from position data and comparison with the actual distance on a map. These data represent the values of the covariance matrix for data reconciliation. Based on the results in Table 3.2, it can be seen that the method provides more accurate position data, which is also supported by the turning path of the forklift truck shown in Figure 3.6. The advantage of the method is that it can be easily applied to clarify systems that provide indoor position data, and the continuously updated error map allows flexible handling of changes in the environment under investigation. The algorithm can be extended by introducing additional equalities and inequalities. For example, this could involve considering physical constraints such as walls and racks, or integrating information about direction, velocity, and acceleration to determine the next position. The method allows for the enhancement of real-time indoor positioning data accuracy, which is essential for resource tracking and process optimisation.

Chapter 4

Algorithm development for identifying value-added and non-value-added activities using position data: a multi-layer network approach

4.1 Introduction to multilayer network analysis for optimized resource flow

In general, the multilayer network-based evaluation of network flows involves a combination of mathematical models, data analysis, and collaboration between stakeholders. Based on the production optimization example, the algorithm was first built using the New York Route Database, as the data series had already been added. The model created in the following is used in an AGV-based (Automated Guided Vehicle) raw material service optimization task, where the states of the machines used in production determine the assignment of the AGV task, and multilayer network analysis can also be used to examine value-added (the AGV carries raw materials) and non-value-added (AGV idling) processes.

My suggestion is that advanced solutions should be based on networked models that account for the stochastic nature of the system. A convincing example can be found in an application where taxi driving strategies are interpreted as a Markov decision process model used in reinforcement learning-based optimization [228].

The contributions of this work for multilayer network-based modeling and balancing of complex processes are the following:

- Implementation of a multilayered network to differentiate between value-added and non-value-added resource flows.
- Design of the network's weighted edges to represent time-dependent flow probability distributions.
- Determination of the resource requirements and surpluses for each supply location based on flow analysis and node representation.

- Solving of the optimal transportation problems at different time instances to determine a network layer for optimal resource re-balancing.
- Development of a method to evaluate the efficiency and resilience of the system through the analysis of the overlap of the network layer.
- Validation of the method using an analysis of New York taxi routes.

The multilayer network presented in this study allows the characterization of efficient or resilient flow, as the system is designed to minimize the number of idle runs and optimize resource allocation through the network. The importance of efficient and resilient flow lies in the fact that it leads to optimized resource utilization and improved system performance. A system that is able to minimize idle runs and effectively allocate resources leads to reduced waste, increased productivity, and efficiency.

The transportation problem is a type of linear programming problem that deals with finding the most efficient way to distribute a limited quantity of goods from a set of sources (such as factories) to a set of destinations (such as warehouses or stores) at the lowest cost. The goal is to minimize the total transportation cost while satisfying the demands of the destinations and the supplies of the sources. The problem can be solved using various methods such as the northwest corner rule, the least cost method, and the stepping stone method [229]. The probabilities of the occurrences of the discrete events (such as in our case taxi-cab-related mobilities) are approximated by continuous kernel density functions that significantly improve the transparency and interpretability of the multilayered temporal network-based model.

The objective of this Section 4 is to identify value-added and non-value-added activities in network flows and to represent them as a multilayer network for easier process understanding. It then presents a methodology for minimizing non-value-added flows as waste. The method is based on reallocating the resources involved in the flows to ensure that non-value added activities are minimized. The research questions of this thesis are first, *How can a multilayered network be effectively implemented to differentiate between value-added and non-value-added resource flows, and how does this contribute to overall system efficiency and resilience?* and second, *What methodologies can be employed to design weighted edges that accurately represent time-dependent flow probability distributions within the network, and how do these contribute to optimizing resource re-balancing in transportation systems?*

According to these research questions, the Section 4 is structured as follows: The next section presents the proposed time-dependent multilayer network structure of the value-added and non-value-added network flows. After that, I show the layer that represents how the network can be optimally rebalanced by solving a transportation problem. Finally, the applicability of the methodology is illustrated through a case study of New York taxi routes.

4.2 Methodology - multilayer network-based modeling and balancing of complex processes

Complex processes can be considered as a networked series of activities characterized by starting and ending times and the set of required resources. The proposed method can be used to analyze any flow-oriented process where the activities are logged with the attributes defined by the T_k ordered list as

$$T_k = (v_s, v_e, t_s, t_e, r_l, l_m), \tag{4.1}$$

where $k = 1 \dots N$ represents the index of activities (transactions) that can be considered as state transitions defined between the starting states v_s and the ending states v_e that are elements of the finite set $\mathbf{V} = \{v_s, v_e\}$, where the number of states is $N_v = |\mathbf{V}|$ and $v_s, v_e \in \{1, \dots, N_v\}$. The state transitions are characterized by starting at t_s and ending at t_e times ($v_s(t_s) \rightarrow v_e(t_e)$), and the required resources $r_l \in R$, where R represents the set of resources. The key idea of this work is that the r_l labels of the resources will be used to form layers of state transition networks. The networks of state transitions will be further enriched by l_m labels that provide additional information on the utilization of related resources; for example, in our case, the label $l_m = \alpha$ is used to represent the value-added activity and $l_m = \beta$ for the non-value-added activity. As many labels as desired can be used to define layers; in our case, two $l_m \in \{\alpha, \beta\} = \{1 \dots M\}$, where M is the number of layers used.

The proposed method transforms the raw data presented into valuable information represented by a multilayer network, $G = (\mathbf{V}, E, D)$ where $\mathbf{V}$ stands for the set of vertices (states of the system), $E = \mathbf{V} \times \mathbf{V}$ is the set of labeled edges which represent the state transitions, and $(r_l, l_m) \in D$ represents the set of dimensions (labels of resources and utilizations as value-added or non-value-added). $E^{(r_l,l_m)}$ represents the set of state transitions based on r_l resource separated by the l_m labels, in our case by the value-added and the non-value added activities.

The layers of the multilayer networks are defined according to these dimensions, where the intra-layer connections are represented by the elements of the $A^{(r_l,l_m)}$ adjacency matrices defined as

$$a_{i,j}^{(r_l,l_m)} = \begin{cases} > 0, & \text{if } (v_i^{(r_l,l_m)}, v_j^{(r_l,l_m)}) \in E^{(r_l,l_m)} \\ 0, & \text{otherwise} \end{cases} \tag{4.2}$$

$a_{i,j}^{(r_l,l_m)}$ represents the weight of the state transitions of the r_l resource from the i state to j according to the l_m type of activity. The weights of the network represent the costs of the activities. When adequate information on these costs is not available, these costs can be approximated by information related to activity times, or based on some distance measure defined between nodes. An example of this will be presented in the case study, in which the cost will be terminated based on the statistical evaluation of the logged activity times.

The interlayer connections represent the weight of the change in activity type. According to the possible transitions between value-added and non-value-added activities, a multiplex network can be formed, where the nodes of the layers are identical for all r_l resources, $\mathbf{V} = \mathbf{V}^\alpha = \mathbf{V}^\beta$, and can be interconnected only with the same nodes in other layers, so $a_{i,j}^{(r_l,\alpha),(r_l,\beta)} = 0, \forall i \neq j$. Figure 4.1 illustrates how connections define the value-added and non-value-added transitions, respectively, while the interlayer edges $a^{\alpha,\beta}_{i,i}$ stand for the transitions when a value-added activity is followed by a non-value-added activity or the other way around ($a^{\beta,\alpha}_{i,i}$).

This work proposes a method that transforms transactional data into valuable information represented by the presented multilayer network. The five steps of the method are depicted in Figure 4.2.

- In the first step, the network of value-added activities is identified based on mobility patterns.
- In the second step, the network of non-value-added flows is reconstructed by tracking the movement of the resources between two value-added activities.
- In the third step, the demands and shortages of resources are calculated.
- In the fourth step, based on the calculated balances, the optimal flows that can rebalance resources are calculated based on solving a network transportation problem.
- Finally, the efficiency and resilience of the network are calculated based on the comparison of this layer and the layer of the non-value-added flows.

In the following subsections, the details of these steps will be introduced.

Time-dependent flow-based representation of value-added activities

One significant novelty of the proposed network-based model is that it interprets the $a^{\alpha}_{i,j}(t)$ edge weights as number of finished state transitions in a given time interval per unit of time that can be integrated at any time interval to calculate the number of the expected state transitions:

$$Q^{\alpha}_{i,j}(t_1, t_2) = \int_{t_1}^{t_2} a^{\alpha}_{i,j}(t)dt \tag{4.3}$$

E.g., in the discussed case, $Q^{\alpha}_{i,j}(t_1, t_2)$ represents the number of taxi cabs that go from zone i to zone j with passengers.

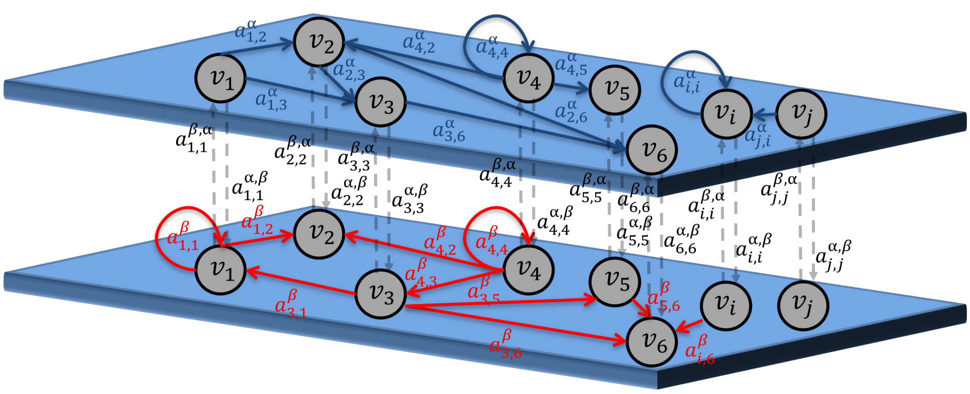


Figure 4.1: Multilayer network representation of the value-added (upper) and non-value-added (lower) activities (the r_l index of the resource is not shown for clarity).

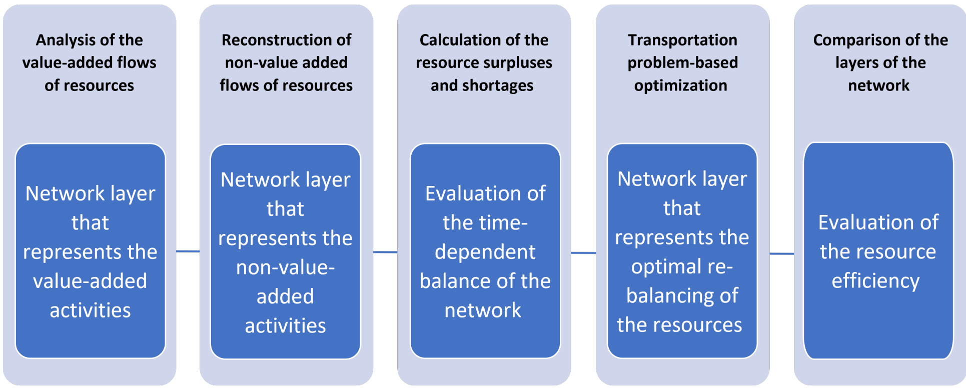


Figure 4.2: The five main steps of the proposed method developed to explore the efficiency of complex processes.

The difference of the $k^{\alpha}_{i,in}$ in-degree and $k^{\alpha}_{i,out}$ out-degree of a node defines how the flows around the i-th node are locally balanced at a given time instant,

$$d^{\alpha}_{i}(t) = k^{\alpha}_{i,in}(t) - k^{\alpha}_{i,out}(t) = \sum_{j=1}^{N_v} a^{\alpha}_{j,i}(t) - \sum_{j=1}^{N_v} a^{\alpha}_{i,j}(t)\,. \tag{4.4}$$

The $d^{\alpha}_{i}(t)$ difference can be normalized to get a node-level asymmetry value, $-1 \leq z_i \leq 1$:

$$z^{\alpha}_{i}(t) = \frac{k^{\alpha}_{i,in}(t) - k^{\alpha}_{i,out}(t)}{k^{\alpha}_{i,in}(t) + k^{\alpha}_{i,out}(t)} \tag{4.5}$$

If $z^{\alpha}_{i}(t)$ is close to zero, the flows around node i are balanced. A positive value z_i means that there is an oversupply at this node, while a negative value z_i shows that the demand at the node is higher than the supply.

When the nodes are considered as inventories, the inventory levels can be calculated as:

$$Q^{\alpha}_{i}(t) = \int_{t_0}^{t} d^{\alpha}_{i}(t)dt + Q^{\alpha}_{i}(t_0) \tag{4.6}$$

$$Q^{\alpha}_{i}(t_0) = 0 \tag{4.7}$$

The network is well balanced and resilient when $Q^{\alpha}_{i}(t) > 0, \forall t > t_0$. For example, in the studied example, $Q^{\alpha}_{i}(t)$ represents the number of taxis in zone i at time t, so when there are no available taxis in zone i, it is necessary to redistribute taxis from other zones, which trips cannot be considered as value-added activities. This example demonstrates that the local unbalances of the value-added activities are balanced by the flows of the network of non-value-added activities.

Network layer of the optimal non-value-added activities

The costs of the non-value-added activities are related to the distances of the non-value-added movements or times where the resources are not utilized. When $c_{i,j}(t)$ represents the cost of the $i-j$ state transition, the $C^{\beta}(t)$ total cost can be formalized as the sum of these costs:

$$C^{\beta}(t) = \sum_{i=1}^{N_v}\sum_{j=1}^{N_v} c_{i,j}(t)a^{\beta}_{i,j}(t) \tag{4.8}$$

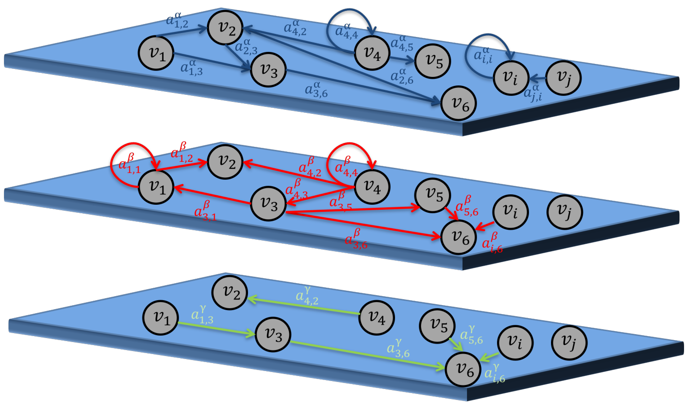


Figure 4.3: The α value-added layer and the β non-value-added layer are extended with a γ layer determined as the optimal non-value-added layer.

To evaluate the efficiency of the process, the non-value-added state transitions needed to ensure the balance of the network are calculated by minimizing the cost function.

$$\min_{a^{\gamma}_{i,j}(t),\forall i,j,t} C^{\gamma}(t) = \sum_{i=1}^{N_v}\sum_{j=1}^{N_v} c_{i,j}(t)a^{\gamma}_{i,j}(t) \tag{4.9}$$

by optimizing the $a^{\gamma}_{i,j}(t)$ values and comparing them to the reconstructed layer of the non-value-added activities (as it is depicted in Figure 4.3).

The $c_{i,j}(t)$ cost of transportation is time dependent, so in this study, this cost was approximated as the average of the transportation times recorded in a t_w time window.

$$c_{i,j}(t) = \frac{1}{m_{i,j}} \sum_{\forall k | t^k_{s_i} - t \leq t_w} \left(t^k_{e_j} - t^k_{s_i}\right), \tag{4.10}$$

where $t^k_{e_j}$ and $t^k_{s_i}$ stand for the end and start times, respectively, and $m_{i,j}$ represents the number of transports that are in the time window defined by the parameter t_w.

Nodes with positive $d^{\alpha}_i(t)$ values are considered as supplies and negative $d^{\alpha}_i(t)$ as demands. As highlighted, the goal of the non-value-added activities is to balance the network. To obtain the optimal rebalancing strategy that has minimal cost, a set of non-value-added activities must be found, $a^{\gamma}_{i,j}(t)$, that minimizes the cost represented by Equation 4.9. The resulting transportation problem is one of the typical applications of linear programming (LP). Equation 4.9 represents a cost function that is linear in the $a^{\gamma}_{i,j}(t)$ decision variables. The optimization problem is constrained by the following linear equality and inequality constraints.

- At each location, the resource transfers must be greater than or equal to zero (no negative values).

$$a^{\gamma}_{i,j}(t) \geq 0 | \forall i,j \in \{1,\ldots,N_v\} \tag{4.11}$$

- The flow-conservation at each node has to be ensured, $d^{\alpha}_i(t) + d^{\gamma}_i(t) \geq 0$

$$d^{\alpha}_i(t) + \left(\sum_{j=1}^{N_v} a^{\gamma}_{j,i}(t) - \sum_{j=1}^{N_v} a^{\gamma}_{i,j}(t)\right) \geq 0 \tag{4.12}$$

The problem is feasible only if the transportation problem is balanced:

$$\sum_{i=1}^{N_v} d_i^{\alpha}(t) \geq 0, \forall t > 0\,, \tag{4.13}$$

To address infeasible problem instances, an always-feasible problem is defined by introducing an additional source:

$$d_0^{\gamma}(t) = -\sum_{i=1}^{N_v} d_i^{\alpha}(t) \tag{4.14}$$

This external source $d_0^{\gamma}(t)$ evaluates how many additional resources should be introduced to the system to meet all the demands (e.g., how many new taxi cabs should be additionally located in different zones). Accordingly, $c_{0,i}$ costs must also be defined, which represent the costs of assigning new resources to the system.

The solution of linear programming problems can be achieved through several available tools, including standalone software solvers specifically designed for linear programming, such as CPLEX or Gurobi. Another option is to utilize online tools, such as NEOS or LINDO, which provide remote access to linear programming solvers. Additionally, spreadsheet programs, such as Microsoft Excel or Google Sheets, feature linear programming solvers as part of their data analysis tools. General-purpose mathematical software packages, such as MATLAB or R, also offer linear programming solvers as part of their toolboxes. In this particular case, the MATLAB R2021b solver was used for linear programming, which provided fast solutions mostly around a second.

Based on the $C^{\gamma}(t)$ optimized and $C^{\beta}(t)$ reconstructed non-value-added state transitions costs, the efficiency of the networked process can be evaluated:

$$E(t) = \frac{C^{\gamma}(t)}{C^{\beta}(t)} \tag{4.15}$$

The layer overlap is an important similarity indicator of the optimized and reconstructed layer which can highlight edges/activities that are critical to balance:

$$O^{\beta,\gamma}(t) = \frac{1}{N_v^2} \sum_{i=1}^{N_v} \sum_{j=1}^{N_v} \frac{\min\left(a_{i,j}^{\beta}(t), a_{i,j}^{\gamma}(t)\right)}{\max\left(a_{i,j}^{\beta}(t), a_{i,j}^{\gamma}(t)\right)}, \tag{4.16}$$

The difference in the degree and the degree of centrality of the nodes can also highlight states that require special attention, which is straightforward, as the proposed values $d_i^{\alpha}(t)$, $d_i^{\beta}(t)$, and $d_i^{\gamma}(t)$ values clearly represent how the external demand of the resources are covered by the non-value-added activities.

4.3 Multilayer network-based performance assessment of taxis in New York City - A Case Study

The zone-based network analysis of taxi trips in New York City has been chosen as a case study because it possesses all the necessary characteristics since

- the nodes of the network that represents states are the taxi zones,
- the edges are the traffic intensities modeled as state transition probabilities,
- the taxi cabs are considered resources,
- passenger transport is interpreted as value-added activity (as the efficiency of the taxi cabs is desired to be maximized).

The January 2012 dataset is being worked with, as it contains additional information, such as the medallion (car ID) and hack license (Driver ID) [38], which is useful for reconstructing the non-value-added movements of the taxi cabs.

The hack license is a license given to an individual who meets the United States Residency, New York State Department of Motor Vehicle Licensing, New York State Department of Financial Services, New York State Tax and Finance, and Municipality's criteria for the privilege of driving a taxicab in the State of New York.

The state of New York is divided into 263 taxi zones (see Figure 4.4). As the studied dataset contains only the GPS coordinates of the pick-up and drop-off places, based on the shape file of the taxi zones, the data are transformed into the proposed network form. As some coordinates do not belong to any of the defined taxi zones, a new zone 264 was defined as the collector of trips outside the zones.

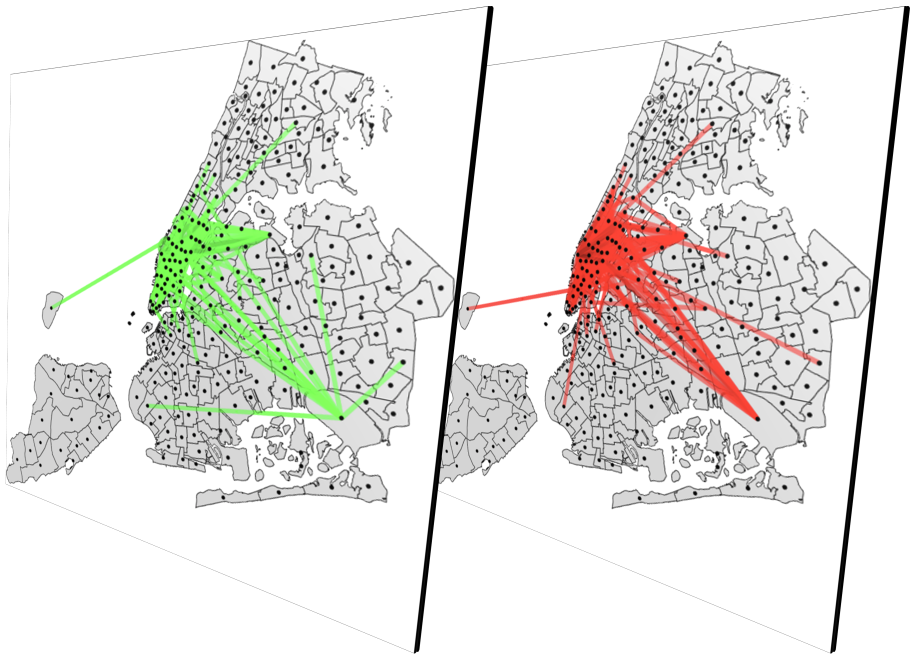


Figure 4.4: The reconstructed network of the value-added (green) and non-value-added (red) routes between New York taxi zones.

In the original dataset, only value-added activities are recorded. Our main idea is that the non-value-added activities/movements can be reconstructed as these are placed between two value-added trips (between the departure zone and the next pick-up zone). The flow intensities are modeled by kernel-density estimation to provide a compact and easy-to-interpret model. With the help of this technique, a large number of discrete events are transformed into probabilistic density functions. The kernel width (bandwidth) was selected to balance the accuracy, transparency (smoothness), and generalization ability of the model.

Based on the calculated edge-weights all the variables presented in the previous section can be calculated for any time-period. Figure 4.5 represents intensities between two randomly chosen zones and the node balance determined from the intensities for the entire network, which will give the node-level asymmetry.

The distance of each node can be determined, but it should also be considered that the traffic intensities of the given routes may change depending on the time. Therefore, it is more expedient for the time-dependent travel times between the individual nodes to be determined, allowing for the indirect calculation of traffic intensity during the optimization. The distances (times) between the specified nodes will be proportional to the transient resource requirement. The non-value-added time (idles times) is used to satisfy the time- and location-dependent resource demand. However, it can be optimized based on minimizing distances (times) as optimal re-balancing.

Figure 4.6 represents the map-based time-dependent value-added, reconstructed

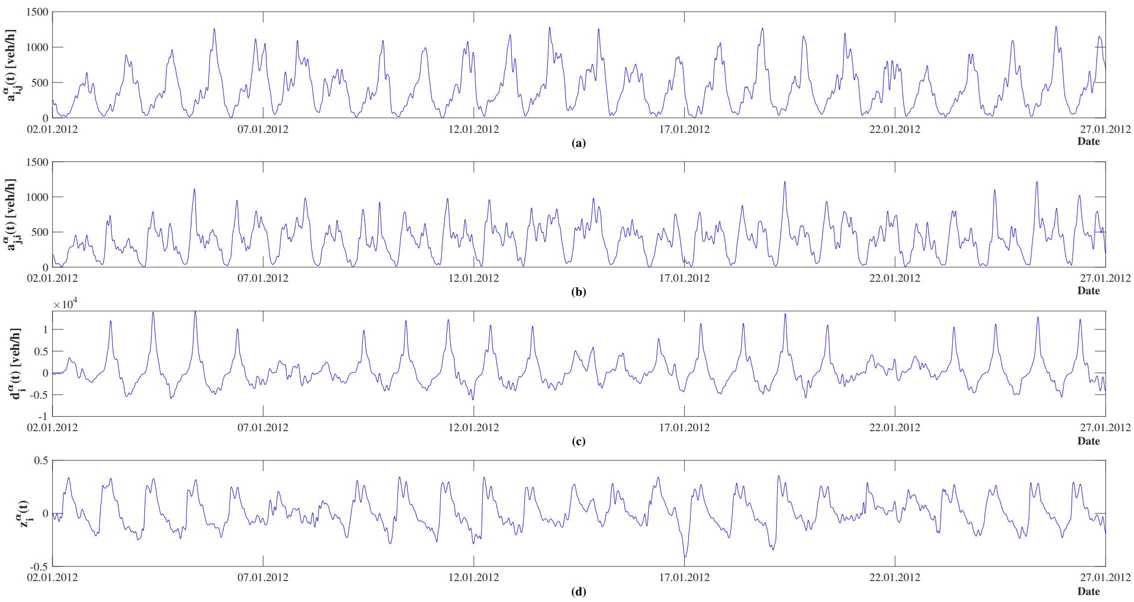
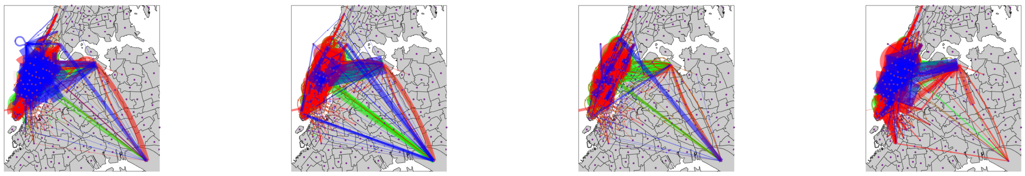
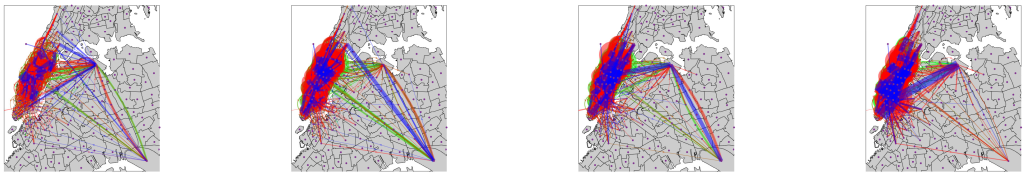


Figure 4.5: **(a)** Visualization of the value-added activity intensities from the 234th to the 48th zone between 2nd and 27th of January. The distribution is approximated with smooth kernel density, representing the number of state transitions. **(b)** Visualization of the value-added activity intensities from the 48th to the 234th zone between 2nd and 27th of January. The distribution is approximated with smooth kernel density, representing the number of state transitions. **(c)** Visualization of the differences of the in-degree and out-degree in the zone 234th between 2nd and 27th of January. The time-dependent values represented the node balance. **(d)** Visualization of the node-level asymmetry.



(a) (b) (c) (d)



(e) (f) (g) (h)

Figure 4.6: Value-added (green), reconstructed non-value-added (red) and optimized non-value-added (blue) activities over a given time interval on Thursday 7am **(a)**, 14pm **(b)**, 17pm **(c)** and 21pm **(d)** and on Saturday 7am **(e)**, 14pm **(f)**, 17pm **(g)** and 21pm **(h)**

non-value-added, and optimized non-value-added activities at different times of the day on weekdays and weekends. Interpreting the value-added, the reconstructed non-value-added, and the optimized non-value-added activities as separate layers, centrality indicators can be determined, based on which the time-dependent properties of these layers can be examined. Such centrality indicators are *authority*, *hub*, and *PageRank*. The *authority* shows the time-dependent dominant zones as target zone in the given time interval. The *hub* shows the dominant zones as starting zones in value-added or non-value-added activities. The *PageRank* taking into consideration the *authorities* and the *hubs*, as well as their direct and indirect connection, determines the time-dependent role of the given zones in the network.

Based on the *PageRank* (Figure 4.7), the popular passenger transportation zone is the Midtown Center (161) in the morning, but the LaGuardia Airport (138) and the JFK Airport (132) also affected. Reconstructed idle trips start in almost every zone in Manhattan. The East Village (79) is the most popular, but again the LaGuardia Airport (138) and the JFK Airport (132) are also affected. As the evening approaches, the importance of the two airports increases. Optimized idle trips are also started from the Manhattan and some Brooklyn zones. The Manhattan Valley (151) is the most popular, but the distribution is more centralized than reconstructed idle trips.

In the early and late afternoon, the popular zones for passenger transport according to *PageRank* are also in Manhattan, especially in Midtown Center (161), Union Square. (234), and Upper East Side North (236). However, the LaGuardia Airport (138) and the JFK Airport (132) are also gaining popularity. Between the reconstructed idle trips, the popular zones are in Manhattan, for example, in Midtown Center (161) and Union Square. (234), but LaGuardia Airport (138) is increasing significantly and JFK Airport (132) is increasing moderately. Centralization of optimized idle trips can be observed again. The affected zones are in Manhattan and Brooklyn. However, they have popular zones such as Midtown Center (161), Upper East Side North (236), Long Island City/Hunters Point (145), UN/Turtle Bay South (233ß), and Financial District South (88).

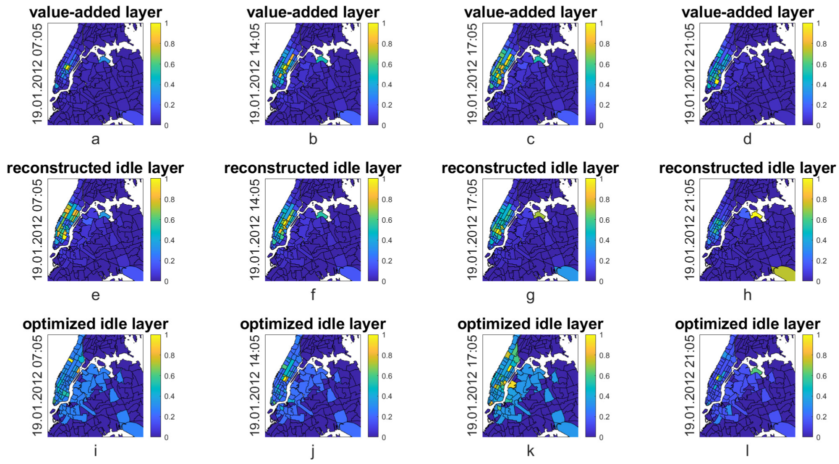


Figure 4.7: The figures show the distribution of passenger transport as value added activities **(a)**-**(d)** and reconstructed **(e)**-**(h)**/optimized **(i)**-**(l)** idle trips as non-value added activities according to the zones' centrality PageRank at different times of day (morning: **(a)**, **(e)**, **(i)**, early afternoon: **(b)**, **(f)**, **(j)**, late afternoon: **(c)**, **(g)**, **(k)**, and evening: **(d)**, **(h)**, **(l)**).

The popular nighttime passenger transport zones are in Manhattan, especially in the East Village (79). The reconstructed idle trips are also popular in Manhattan, but the most popular zones are the LaGuardia Airport (138) and the JFK Airport (132). The LaGuardia Airport (138) and Times SQ / Theater District (230) are popular zones in the optimized layer. However, Manhattan and Brooklyn are also affected by idle trips but are more centralized than reconstructed idle trips.

One parameter for comparing the reconstructed and optimized networks can be the total idle time during the time unit. Therefore, the C^{β} reconstructed and the C^{γ} optimized non-value-added state transitions costs can be represented with the time. The non-value-added activities, such as idle trips, are time-wasting. Recorded in 10-minute time intervals, the total travel time can be determined for the trip in the reconstructed and optimized activities mesh interval. Figure 4.8 shows the total idle

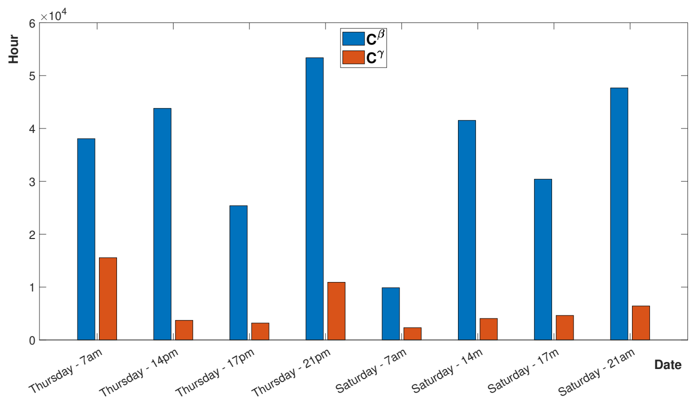


Figure 4.8: Comparison of the C^{β} reconstructed and the C^{γ} optimized cumulated idle time over a given time interval on Thursday 7am, 14pm, 17pm and 21pm and on Saturday 7am, 14pm, 17pm and 21pm

travel times in the given intervals of 10 minutes on weekdays and weekends, morning, afternoon, and evening based on the β reconstructed and γ optimized layers. The idle times are defined based on the network of non-value-added activities, and the cost is defined as the sum of the idle times in the given time interval. In the original dataset, the transport time of the value-added activities (passenger transport) was recorded, including, among others, the pick-up and drop-off zones of the passengers. By analyzing the entire database, the average travel time between zones can be determined. This analysis can be further specified, for example, by examining travel times during specific times of day, as these can be influenced by changes in traffic intensity. Using the average travel times between zones as a foundation for costs, the performance of the reconstructed and optimized idle routes was compared in the given time interval at different times of the day. It can be seen that a significant reduction can be achieved through the optimization of demand servicing.

A novel approach to addressing the challenges of supply chain optimization and resource allocation is introduced in this thesis. Unlike previous studies, the developed method utilizes a multilayered temporal network model to analyze network flows, distinguishing between value-added and non-value-added activities. The model enables a comprehensive understanding of resource flows and facilitates resource reallocation to minimize non-value-added flows.

4.4 Summary of the chapter

This thesis presents a comprehensive approach for evaluating the efficiency of network flows through the application of a multilayer network model. The developed solution offers valuable insights for the analysis of both supply chains and urban transportation systems. A key aspect of optimization is the identification and reduction of idle times, which are classified as non-value-added activities. The multilayer network framework classifies value-added and non-value-added activities, where nodes represent the states (starting and destination points), and edges represent the state transitions as network flows. By incorporating time-dependent probability distributions, this model provides a detailed analysis of the resource allocation challenges, optimizing supply-demand ratios for each location. Using continuous-time formalization and kernel density estimation, discrete data are transformed into a continuous representation, allowing for the optimization process to be performed within a discrete-time framework.

The application in this research involves the analysis of New York City taxi flows, demonstrating the multilayer network approach's ability to evaluate efficiency and identify optimization opportunities. Idle times were compared to optimal resource allocation solutions using transportation problem-solving techniques, revealing potential for significant reductions in non-value-added activities. The analysis showed how value-added and non-value-added activities could be reconstructed, providing a foundation for route optimization and improved resource utilization in urban transport systems. The findings offer insights for city mobility improvements, enabling data-driven decisions regarding route optimization, pricing strategies, fleet management, and transportation system enhancement.

The multilayer network approach offers a detailed, data-driven method for analyzing value-added and non-value-added activities, enabling the identification of bottlenecks and inefficiencies in resource management. The case study on New York taxi routes illustrated how this approach could be used to optimize network flows, reduce idle times, and enhance decision-making processes in urban mobility.

The proposed method also highlights the importance of managerial insights gained from multilayer network analysis. By understanding resource flows and activity classifications, companies can make better-informed decisions regarding resource allocation, route optimization, and operational efficiency. Key Performance Indicators (KPIs) such as the Resource Utilization Indicator help track and evaluate the efficiency of resource usage, identifying areas for further improvement. The multilayer network approach not only facilitates a deeper understanding of complex processes but also supports cross-industry applications in optimizing intralogistics operations, particularly those involving forklifts and other material handling equipment in warehouse settings.

Chapter 5

Clustering and network analysis for identifying value-added and non-value-added activities using position data in intralogistics environment: A DBSCAN-based Approach

5.1 Introduction to DBSCAN clustering-based activity analysis in logistic

Clustering techniques, particularly DBSCAN, play a pivotal role in logistics and production optimization tasks. In the context of Lean philosophy, applying clustering methods to categorize activities as value-added or non-value-added can provide companies with valuable insights into operational inefficiencies. By identifying idle times and process bottlenecks, organizations can streamline operations, minimize waste, and enhance resource utilization.

This research focuses on addressing these challenges by introducing a novel approach that uses position data from indoor positioning systems (IPS) and applies DBSCAN clustering to identify value-added and non-value-added activities. The results of this classification are further explored through multilayer network modeling to provide a comprehensive framework for process optimization.

The contributions of this research are as follows:

- Development of a novel clustering-based methodology using DBSCAN to identify and categorize value-added and non-value-added activities in logistics environments.
- Integration of movement trajectory analysis to enhance the accuracy of activity classification based on resource movements.
- Design of a multilayer network model that represents the relationships between value-added and non-value-added activities and provides actionable insights for process optimization.

- Validation of the methodology through a real-world case study, demonstrating its effectiveness in improving resource utilization and reducing idle times in internal logistics systems.

This methodology provides a unique approach to improving process efficiency by leveraging indoor positioning data and advanced clustering techniques. The research fills a gap in existing literature by applying these methods in a multilayer network framework, which has not been extensively explored before.

5.2 Methodology – DBSCAN clustering for IPS-based data analysis

In this section, a detailed methodological overview is provided based on the steps illustrated in Figure. 5.1.

The aim of the methodology is to classify process-related activities into value-added and non-value-added categories based on indoor positioning data. Our method combines clustering, time series analysis and movement trajectory analysis. Through clustering, the places where the resource performs value-added and non-value-added activities are identified using movement trajectory analysis. After that, state transitions can be identified through time series analysis, thereby revealing idle activities for optimization. Activities are discerned through the analysis of positional data, wherein the spatial coordinates serve as crucial indicators. The contextual background of the data plays a pivotal role in the identification of various states associated with these activities. To enhance the precision of this process, positional data undergoes clustering through the application of DBSCAN (16), which is meticulously tuned to delve into the typical positions and places where activities unfold. In this study, a modified version of DBSCAN is developed, focusing on identifying whether a resource is engaged in value-added or non-value-added activities based on positional data. Without this modification, the construction of multilayer-based clustering would not be possible, as the standard DBSCAN algorithm would not sufficiently differentiate between value-added and non-value-added activities based on positional data. This approach creates separate cluster groups depending on the activity category, and these clusters are represented as multilayers. This multilayer representation assists in uncovering inefficiencies and optimizing processes. Additionally, the modified DB-

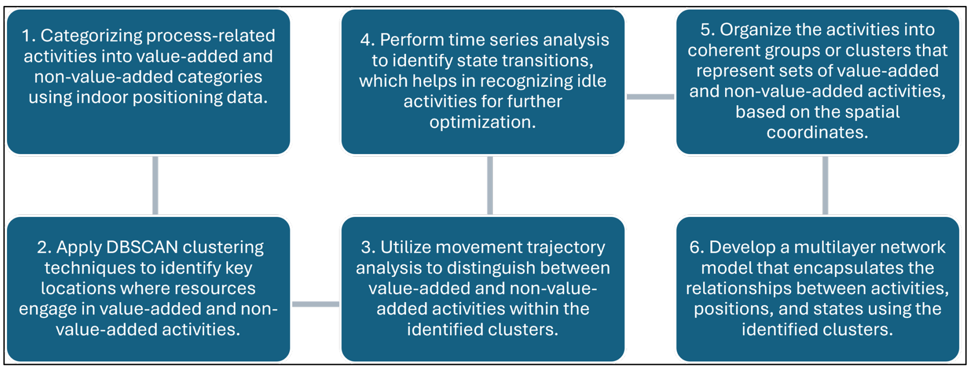


Figure 5.1: Classification and optimization of activities based on Indoor Positioning Data.

SCAN is specifically tailored to support the processing of positional data, enhancing the precision of activity classification. This clustering results in the formation of coherent groups or clusters, each representing the set of value-added and non-value-added activities. Subsequently, a multilayer network is constructed, drawing upon the tuples that encapsulate the relationships between activities, positions, and states. The two-step methodology, based on the provided text, includes the following steps:
1. Step - Identification and Grouping of Activities:

- Activities are discerned through the analysis of positional data using spatial coordinates.
- The contextual background of the data plays a pivotal role in identifying various states associated with these activities.
- Positional data undergoes clustering using DBSCAN (Density-Based Spatial Clustering of Applications with Noise), meticulously tuned to explore typical positions and places.

2. Step - Network Modeling and Optimization:

- The groups or clusters formed in the previous step represent activities, including both value-added and non-value-added activities.
- A multilayer network is constructed using tuples encapsulating relationships between activities, positions, and states.

The available T_k database is structured as follows:

$$T_k = (t_k, d_k, r_l, l_k)$$

where $k = 1 \dots N$ represents the index of the recorded position data, t_k is the time of the k-th recorded data, $\mathbf{d}_k = [d_x, d_y, d_z] \in \mathbf{D}$ is the k-th position data vector, r_l is the unique identification number of the $r_l \in \mathbf{R}$ resources and $l_k = \{\alpha, \beta\}$ is the activity qualifying flag, which identifies the value-added $(\alpha = 0)$ and non-value-added $(\beta = 1)$ activity of the r_l resource at the k-th position.

The next step is to define value-added and non-value-added activities based on position data with the DBSCAN algorithm. Figure 5.2 shows the steps of applying the algorithm:

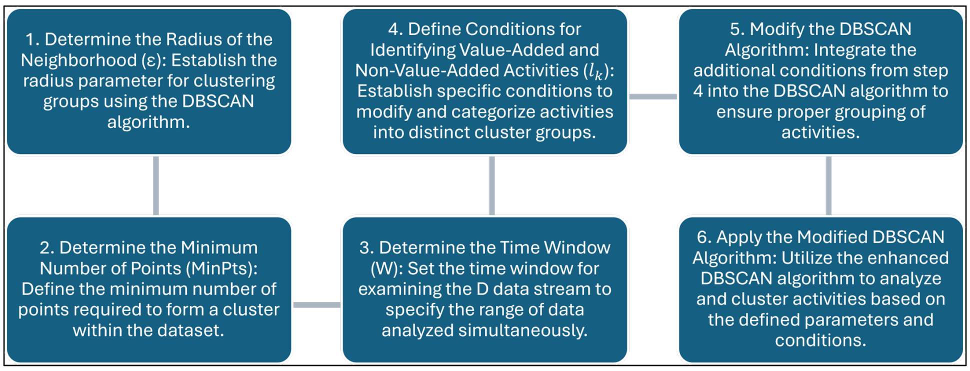


Figure 5.2: Steps of applying Modified DBSCAN algorithm.

When using the algorithm, a **W** vector is defined with a window size for the time-dependent **D** data stream for clustering. This is necessary because time series data is being dealt with, making it important to observe the sequence among the created clusters. It is possible for several clusters to be defined for the same position on the layout, but interaction with them occurs at different times in chronological order.

For the DBSCAN algorithm, the following parameters should be set:

- Data stream contains all of $\mathbf{d_k}$ positions in chronological order based on t_k: **D**
- Radius of the neighborhood: ϵ
- Vector used to collect data that meets the determined conditions: idx
- Minimum number of points required to form a C cluster: MinPts
- **W** vector with window size of the time-dependent **D** data stream for clustering: $i : w$
- $\mathbf{D}_{(i:w)}$ data stream within $i : w$ time window: $\mathbf{D}_{(i:w)}$
- Set of clusters: $C = \{C_1, C_2, \ldots, C_k\}$, where k is the number of the created cluster and $(\mathbf{d}^{(C_k)}, t^{(C_k)}) \in C_k$
- C^j tuple of the C clusters based on $l_k = \{\alpha, \beta\}$ activity qualifying flag

The method is the following: Algorithm 1. The pseudo code of the DBSCAN algorithm

for j=0:1 **do**
    Initialize C as an empty list and $i = 1$
    Initialize vector, Window W with size w for each new data point $d \in \mathbf{D}$
    Mark all $d \in \mathbf{D}$ as not visited
    if $|\mathbf{D}| < w$ **then**
        Add all $d \in \mathbf{D}_{(i:w)}$ to W
        $d_1 = W_1$
        if all elements of W are not visited **then**
            for $d \in W$ **do**
                if $|d - d_i| < \epsilon$ AND $(l_k = j)$ **then**
                    add d to idx
                end if
            end for
            if $|\text{idx}| \geq$ MinPts **then**
                Let $C =$ idx be a new cluster
                Add all idx related t time of the recorded data to C
                Mark all $d \in C$ in $\mathbf{D}$ as visited (no overlap)
                Add C to C^j
            else
                Mark d_i point as outlier
            end if
            Clear the elements of idx
        end if

Clear the elements of W
$w = w + 1$
$i = i + 1$
end if
end for

From the C^j qualified clusters - where the value-added activities are from $j = 0$ clusters and the non-value-added activities are from $j = 1$ - and by matching the position data-based cluster centers to the layout, the V state vector could be determined. The elements of the state vector were based on the positions of layout-based clusters to identify where activity occurred within the examined production or logistics environment under study. The number of unique states is the number of areas defined based on the layout $|V| = N_V$. Since the cluster itself identifies a value-added activity, for each state there is a state transition that points back to itself, which shows the activity in the position according to the given layout.

Based on the previous paragraph, the state of the resources is already known, indicating their layout-based position and the type of activity being performed (value-added or non-value-added). However, it is important to study the state transitions, as classifying these transitions allows for the characterization of transport activities and the minimization of idle time. This requires a multilayered network analysis.

Based on the t_k time data, the t_s^C start and t_e^C end times in the given state can also be determined. Therefore, the Time Series Analysis with focus on the state can create the $E = V \times V$ state transition matrix. Based on this information, a $G = (V, E, F)$ multilayer network can be created, where $(r, l) \in F$ represents the set of dimensions (labels of resources and utilizations). If one resource is focused on, the multilayer network based on the α dimension is as follows:

$$G^\alpha = (V^\alpha, E^\alpha)$$

where V^α are the nodes of the multilayers and E^α is the set of intra-layer connections represented by the elements of the A^α adjacency matrices defined as

$$A^\alpha = [a^\alpha_{p,q}]_{N_V \times N_V}$$

$$a^\alpha_{p,q} = \begin{cases} > 0, & \text{if } (v^\alpha_p, v^\alpha_q) \in E^\alpha \\ 0, & \text{otherwise} \end{cases}$$

where $a^\alpha_{p,q}$ represent the weight of the state transitions from the p state to q according to the type of activity. The $a^{(\alpha,\beta)}_{p,q}$ interlayer connections between the nodes of the layers are defined as:

$$a^{(\alpha,\beta)}_{p,q} = \begin{cases} \geq 1, & \text{if } (v^\alpha_p, v^\alpha_q) \in E^{(\alpha,\beta)} \\ 0, & \text{otherwise} \end{cases}$$

State transitions can be:

$$a_{p,q} = \begin{cases} a^{(\alpha,\alpha)}_{p,q} = a^\alpha_{p,q} : & \text{value-added activities} \\ a^{(\beta,\beta)}_{p,q} = a^\beta_{p,q} : & \text{non-value-added activities} \\ a^{(\alpha,\beta)}_{p,q} : & \text{value-added activities} \\ a^{(\beta,\alpha)}_{p,q} : & \text{non-value-added activities} \end{cases}$$

Figure 5.3 illustrates how connections define the value-added and non-value-added transitions. With the help of the method, the processes can be qualified, which is indirectly useful for process optimization. Resource utilization can also be determined, and non-value-added processes (as idle route) can be reduced to perform value-added tasks by reducing them with an effective optimization procedure.

5.3 Multilayer network-based performance assessment of intralogistics processes using IPS data and DBSCAN clustering – a case study

Our use case originates from a real logistics environment, where forklifts are tracked by a UWB-based IPS during the storage process. The forklifts are equipped with two tags (as seen in Figure 3.2), the O-signed IPS sensor tag II. moves with the fork and five forklifts are monitored.

These tags continuously transmit real-time position data at specified intervals during movement. If the forklift is not moving, the data recording time intervals are longer. The Figure 3.3 represents the layout of warehouse environment where material flow is managed using the aforementioned forklifts and blue markings are the movement of one forklift based on the integrated IPS sensor tag data. This includes storage operations as from the delivery zone to the high warehouse, as well as the reverse process as picking out.

Definition of the used states for DBSCAN clustering

Based on d_k position data, two states should be determined: when the forklift is active (moving) and not active (waiting). These can be identified from the δ_k data recording frequency:

$$\delta_k = t_k - t_{(k-1)} = \begin{cases} \leq d_t \rightarrow d_k \in D_a \\ > d_t \rightarrow d_k \in D_n \end{cases}$$

$$D_a \cap D_n = \emptyset$$

where d_t is the limit of the active recording frequency, D_a is the vector of active position data and D_n is the vector of non-active position data. In our case d_t is three

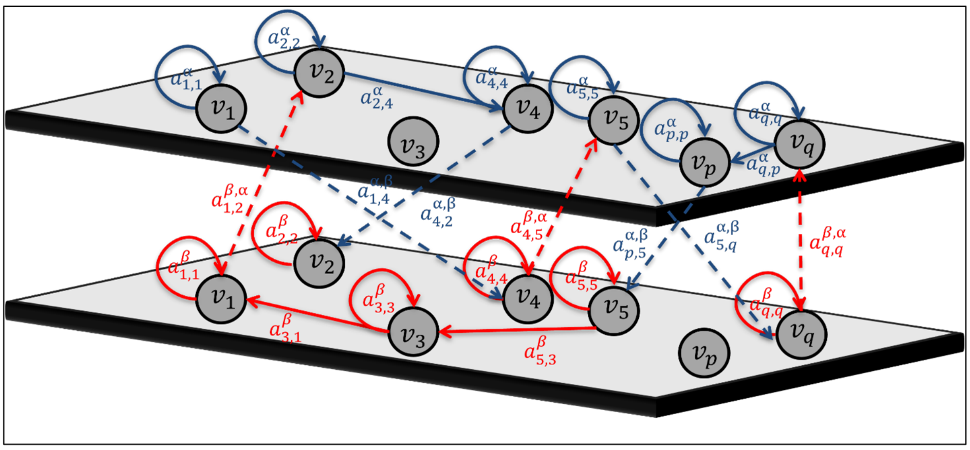


Figure 5.3: Multilayer network representation of the value-added (upper) and non-value-added (lower) activities of one resource.

seconds. The working of the forklift can be defined by the fact that the member attached to the fork moves in the z-coordinate direction:

$$d_k(d_z) = \begin{cases} > m & \rightarrow \text{working} \\ \leq m & \rightarrow \text{not working} \end{cases}$$

where m is the movement limit in z direction. The Table 5.1 summarizes, what states should be defined based on the position data, and the Table 5.2. represents the classification of the activities based on given state transitions. The methods of defining the states are presented in detail in the remaining parts of the chapter.

	Forklift is active ($d_t \leq 3$)	**Forklift is not active ($d_t > 3$)**
Forklift is working ($d_k(d_z) > m$)	Cluster 1: value-added activity	Cluster 3: not applicable
Forklift is not working ($d_k(d_z) \leq m$)	Cluster 2: non-value-added activity	Cluster 4: waiting

Table 5.1: Defined states based on position data driven clustering.

State transition	**Classification of the activities**
1 - 1	Value-added (on itself)
1 - 2	Value-added
1 - 3	Going to waiting
2 - 1	Non-value-added
2 - 2	Non-value-added (on itself)
2 - 3	Going to waiting
3 - 1	Going to do value-added activity
3 - 2	Going to do non-value-added activity
3 - 3	Waiting (on itself)

Table 5.2: State transitions based classification of the activities.

In the second step, the clusters of Table 5.3 must be determined using the DBSCAN algorithm.

	$D_1 = \mathbf{D}^a$: Forklift is active	$D_2 = \mathbf{D}^n$: Forklift is not active
$I_1 = A = \text{idx}(d_z) > m$: Forklift is working	$C_{1,1}$: Clusters of value-added activities	$C_{2,1}$: Empty clusters (not applicable as the forklift is not active and working state together)
$I_2 = B = \text{idx}(d_z) \leq m$: Forklift is not working	$C_{1,2}$: Clusters of non-value-added activities	$C_{2,2}$: Clusters of waiting

Table 5.3: Defined clusters based on the states.

For the DBSCAN algorithm, the following parameters should be set:

- Data stream contains all of active positions in chronological order based on t_k: $\mathbf{D}^a$
- Data stream contains all of non-active positions in chronological order based on t_k: $\mathbf{D}^n$
- Radius of the neighborhood: ϵ
- Movement limit in z direction: m
- Vector used to collect data that meets the determined conditions: idxp
- Minimum number of points required to form a C cluster: MinPts
- $\mathbf{W}$ vector with window size of the time-dependent $\mathbf{D}^a$ or $\mathbf{D}^n$ data stream for clustering: $i : w$
- $\mathbf{I}$ matrix contains the $\mathbf{D}^a$ and $\mathbf{D}^n$ data streams separated by m condition
- $\mathbf{D}^a$ or $\mathbf{D}^n$ data stream within $i : w$ time window: $\mathbf{D}^a_{(i:w)}$, $\mathbf{D}^n_{(i:w)}$
- Set of clusters: $C = \{C_1, C_2, \ldots, C_k\}$, where k is the number of the created clusters
- $C_{(f,b)}$ tuple of the C clusters based on
- Variables to handle the number of for loop iterations: f, b

The method of the cluster determination is the following: Algorithm 2. The pseudo code of the modified DBSCAN algorithm

Initialize C as an empty list
$D = [\mathbf{D}^a \quad \mathbf{D}^n]$
$A = D(d_z) > m$
$B = D(d_z) \leq m$
$I = \begin{bmatrix} A & B \end{bmatrix} = \begin{bmatrix} \mathbf{D}^a(d_z) > m & \mathbf{D}^a(d_z) \leq m \\ \mathbf{D}^n(d_z) > m & \mathbf{D}^n(d_z) \leq m \end{bmatrix}$
for $f = 1 : 2$ **do**
    for $b = 1 : 2$ **do**
        Initialize vector, Window W with size w for all each new data point $d \in I_{(b,f)}$
        Mark all $d \in I_{(b,f)}$ as not visited
        if $|I_{(b,f)}| < w$ **then**
            Add all $d \in I_{(b,f)_{(i:w)}}$ to W
            $d_1 = W_1$
            if all elements of W are not visited **then**
                for $d \in W$ **do**
                    if $|d - d_1| < \epsilon$ **then**
                        add d to idx
                    end if
                end for
                if $|\text{idx}| \geq \text{MinPts}$ **then**
                    Let $C = \text{idx}$ be a new cluster

Mark all $d \in C$ in $I_{(b,f)}$ as visited (no overlap)
Add C to $C_{(f,b)}$
else
Mark d_1 point as outlier
end if
Clear the elements of W
$w = w + 1$
$i = i + 1$
end if
end if
end for
end for

Result of the DBSCAN clustering

The cluster groups created based on DBSCAN are shown in Figures 5.4 and 5.5. The first layout in Figure 5.5 shows the $C_{1,1}$ clusters of value-added activities, the second the $C_{1,2}$ clusters of non-value-added activities, and the third the $C_{2,2}$ clusters of waiting.

With the center of the cluster groups, they can determine which area of the warehouse environment the given activity applies to (Figure 5.6):

With the help of the cluster groups, individual states of the forklift can be classified, and by defining the state transitions defined in Table 5.2, the classifications of the movement of the forklift can also be determined. The multilayer network of the non-value-added (red arrows) and value-added (green arrows) activities is seeable in Figure 5.7. The value 45 marks the locations forklifts arrive from outside the warehouse. Item for storage is temporarily stored in Aisle I (A1). As can be seen in the left-hand diagram, non-value-added routes typically lead from Aisles I (A1) and II (A2) to the high-bay area marked ,,P". This is understandable, as this is where forklifts depart to picking tasks. Conversely, there are movements in the opposite direction, albeit to a lesser extent, indicating instances where forklifts embark on storage tasks, also idle. Typical empty trips occur between Packing (Pa) and Aisle I (A1), indicating the idle before the packed products transport or idle after transport of awaiting packing products. There are also movements towards the Delivery zone (DZ), explaining forklifts travelling to pick up finished products.

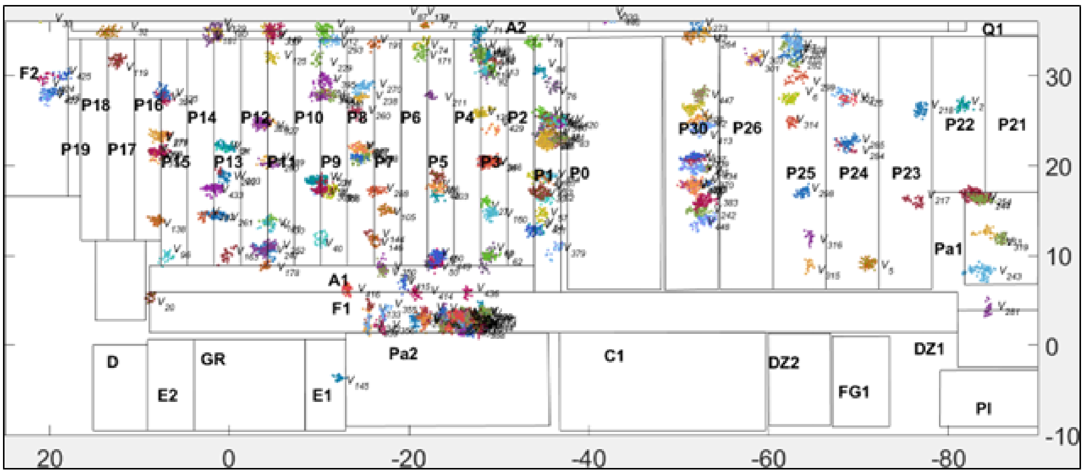


Figure 5.4: Indoor positioning data-based clustering in a warehouse environment.

Frequent value-added movements can be seen in the right-hand diagram, typically go from the high-bay area, marked ,,P", to Aisle I (A1), which makes sense as forklifts are carrying out storage operations. Of course, there are also movements in the opposite direction, representing picking operations. Value-added activities from the delivery zone are also storage activities.

By adopting a multilayered network approach, possibilities for optimizing both value-added and non-value-added processes are opened. Through techniques such as resource reallocation, idle time can be effectively minimized, the number of required

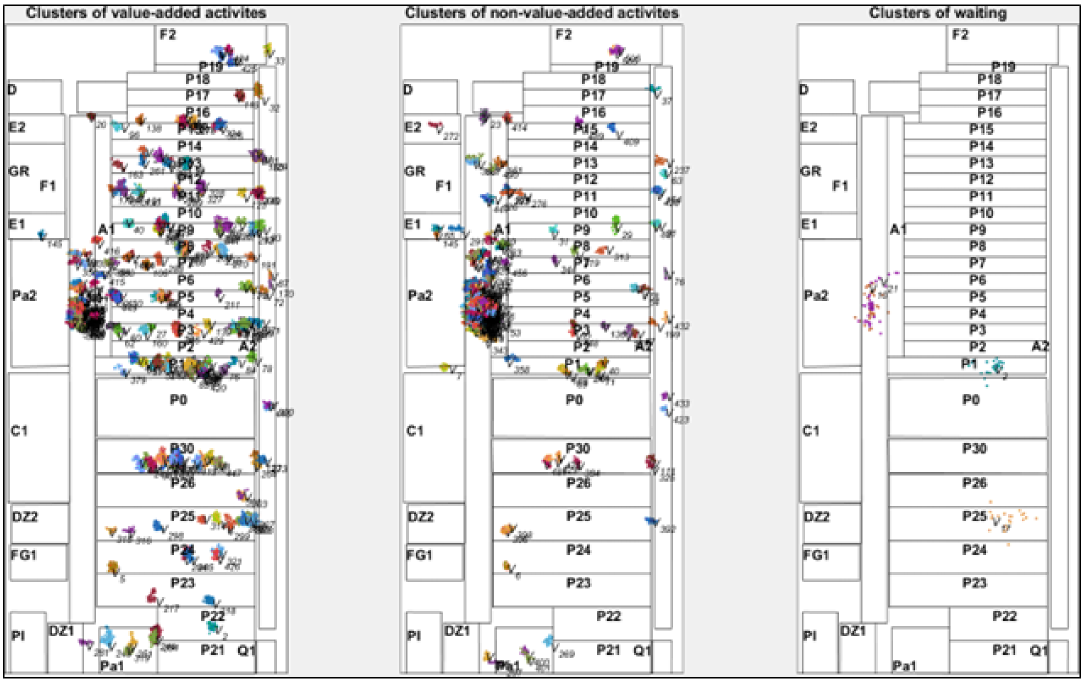


Figure 5.5: Created clusters based on Table 5.3 activities.

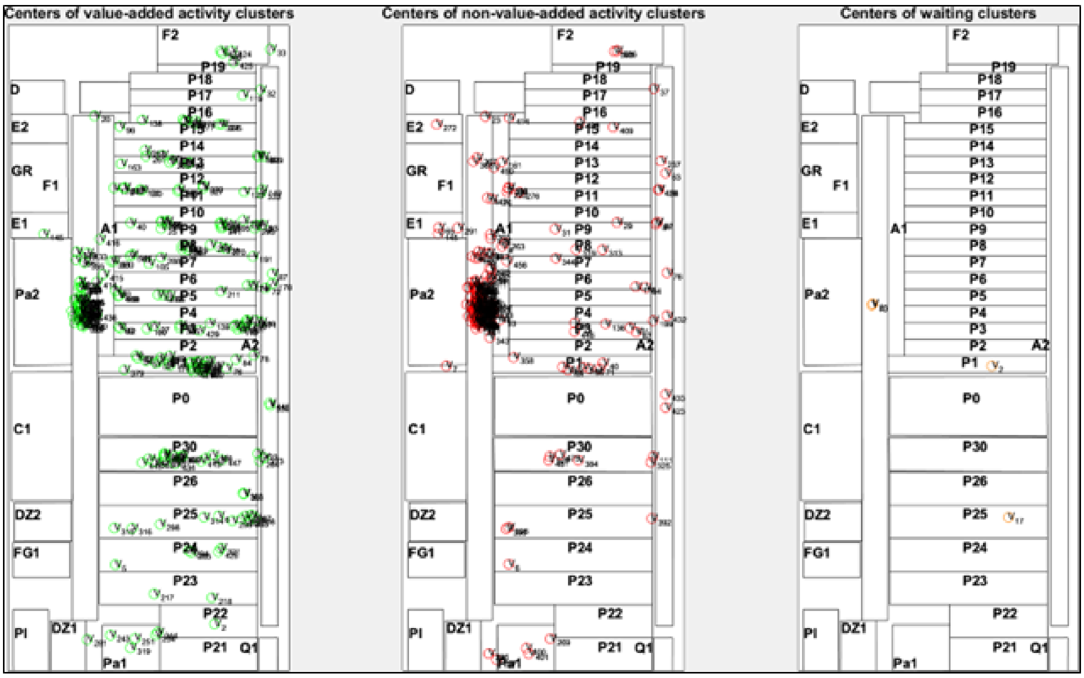


Figure 5.6: Centre of clusters for identifying the layout-based states.

resources can be reduced, and utilization can be improved. This comprehensive strategy allows ultimately leading to improved productivity and cost-effectiveness. Without the modified DBSCAN algorithm, only clusters could be identified. However, by monitoring the movement of the forklift forks, these clusters can be distinguished more effectively, as demonstrated in Figure 5.8. Naturally, the way these activities should be defined and incorporated into the algorithm as additional conditions depends on the specific case being examined. Any specific suggestions or adjustments can be tailored to better fit the context of the study. One of the main challenges in implementing this methodology in the warehouse environment was the accuracy of the position data. The accuracy of the data was inhomogeneous due to environmental factors such as noise and physical obstructions. To address this issue, a data reconciliation method was developed to improve the accuracy of the position data. This method involved the use of advanced filtering techniques and algorithms to reconcile discrepancies and improve data accuracy [[230]]. Five forklifts were equipped with IPS tags, which transmitted positional data. The sample case demonstrates the applicability of the method; however, it would fundamentally require the involvement of more equipment to effectively implement resource reallocation and optimization. Another significant challenge was clustering, especially when cluster groups were in close proximity. This situation occasionally resulted in "jumping" between clusters,

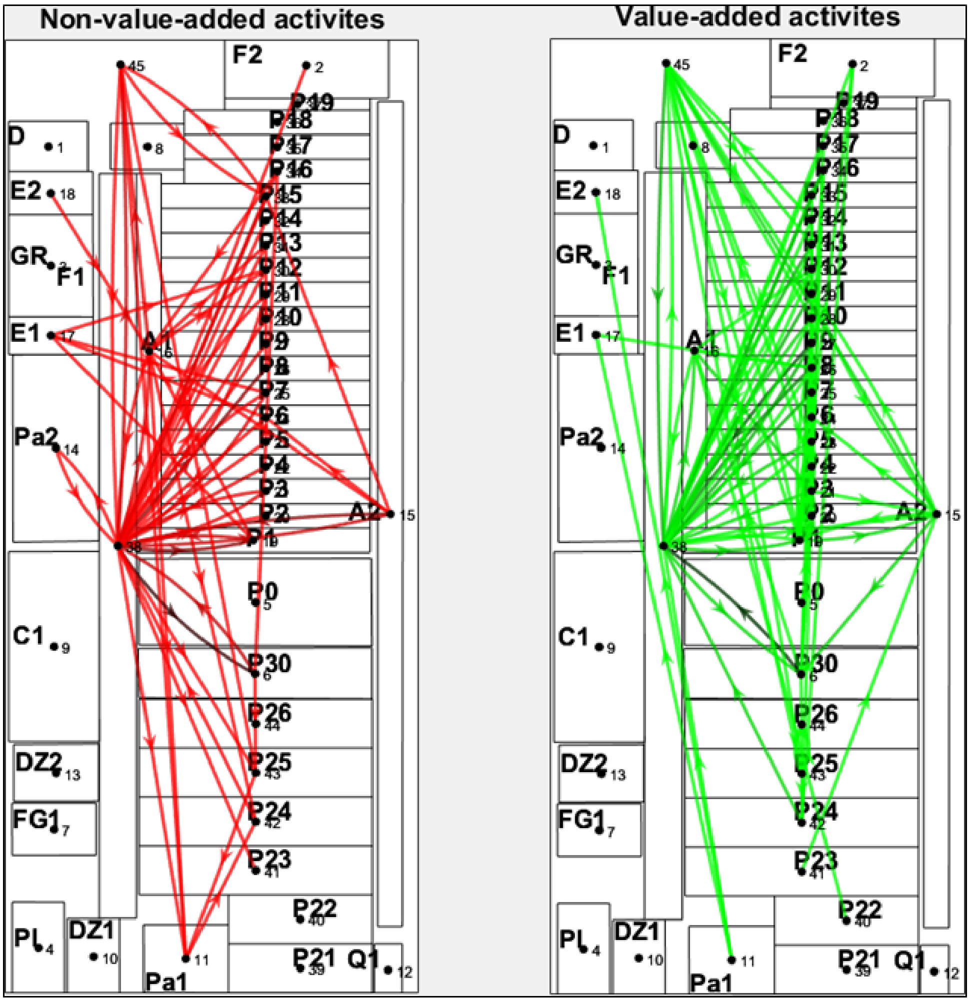


Figure 5.7: Representation the multilayer network of the forklift activities.

where data points inaccurately moved from one cluster to another. This issue was mitigated by refining the clustering algorithms and further improving the data accuracy through our reconciliation process. These combined efforts helped stabilize the cluster assignments and provided a more reliable representation of the position data, ultimately improving the overall effectiveness of the methodology. The algorithm processes position data from five forklifts, representing more than 800,000 data points collected over three days, in 27.877153 seconds in a MATLAB environment, including the generation of map-based visualizations. While the algorithm can be further optimized to reduce processing time, the current limit on the number of resources that can be analyzed is primarily determined by the computational power and time required to process the data.

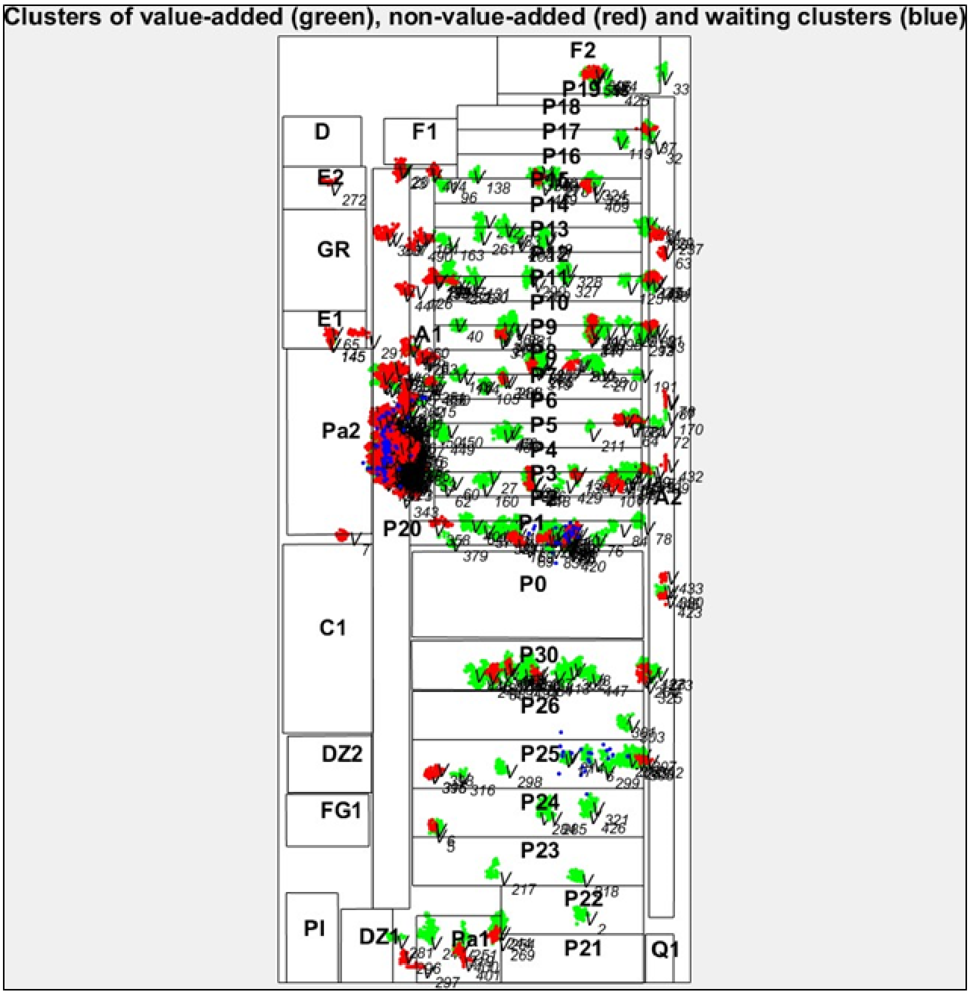


Figure 5.8: With the modified DBSCAN algorithm, the position data-based clusters can be classified and differentiated based on forklift fork movements, enabling the identification of distinct value-added, non-value-added, and waiting activities.

5.4 Summary of the chapter

This thesis proposes a method for classifying resource activities as value-added and non-value-added based on indoor positioning system data using the DBSCAN algorithm. The multilayer network-based solution presented here can facilitate supply chain analysis. By classifying position data and utilizing layout-based clustering, the method determines the activity status of resources within a given cluster group (performing value-added activity, non-value-added activity, or waiting). Additionally, it identifies state transitions that characterize transport activities, which can be visualized in a multi-layer network.

The methodology is illustrated through analysis of indoor positioning data from a logistics environment, monitoring five forklifts equipped with position sensors on their forks. These sensors allow the classification of activity states based on movement patterns and data recording frequencies. The method identifies state transitions, aiding in the classification of transport activities. Multi-layer network-based visualization of these activities provides insights into operational processes and highlights potential optimization opportunities.

The approach holds promise for application across various industries where resource activity tracking and optimization are essential. Future research may explore its implementation in manufacturing or warehousing environments to enhance operational efficiency. Unlike existing methods, this approach enables more detailed and dynamic classification of activities by leveraging the multilayer network representation and DBSCAN's clustering capabilities. This results in a nuanced analysis of state transitions and activity patterns, offering insights that may not be captured by static or simpler models.

The practical implications are significant for logistics and supply chain management. By accurately classifying activities and identifying value-added processes, companies can streamline operations, reduce waste, and improve overall efficiency. The multi-layer network visualization also clarifies complex processes, making it easier to identify bottlenecks or inefficiencies.

However, limitations exist, notably the dependency on the quality of positioning data and specific environmental characteristics. Future research may address these limitations by integrating advanced sensor technologies or by applying the method in diverse contexts to evaluate robustness and generalizability.

A key challenge in warehouse environments was data accuracy, affected by environmental noise and physical obstructions. To improve accuracy, a data reconciliation method was developed, incorporating filtering techniques to reduce discrepancies. This study demonstrates the method's applicability, though effective resource reallocation and optimization would require a broader equipment deployment.

Another challenge was clustering, especially with close-proximity clusters, where "jumping" between clusters could occur. This issue was mitigated by refining the clustering algorithms and improving data accuracy, stabilizing cluster assignments and enhancing the reliability of the position data, ultimately improving the methodology's effectiveness.

Chapter 6

Exploring micro-logistics and material handling process optimization using RTLS positioning data: a Reinforcement Learning and Markov Decision Process framework for AGV scheduling

6.1 Introduction to AGV scheduling optimization using Markov Decision Process

This research focuses on addressing the challenges associated with AGV scheduling in industrial practice, illustrated through a demonstrative case study. The case study explores the practical application of the MDP model as a framework for AGV scheduling, highlighting its potential to improve logistics efficiency. As logistics processes become more complex and data-intensive, the MDP model stands out as a promising approach to ensure optimal task execution, resource utilization, and predictive capabilities in modern manufacturing environments. In the following sections, the theoretical foundations of the MDP model, its practical implementation challenges, and the outcomes observed through its application in real-world scenarios will be explored in greater depth. The evolving role of AGVs in the logistics domain highlights the necessity for advanced scheduling mechanisms, and the MDP model emerges as a valuable tool to meet these demands.

The contributions of this work are the following.

- Assessment of current manufacturing and logistics processes.
- Exploration of potential opportunities for process improvement.
- Construction of a Markov Decision Process model.
- Validation of the method using an analysis of the real production environment.

According to these contributions, the Section 6 is structured as follows: The next Section 6.2 presents the structure of the proposed MDP model and the way it is developed. In Section 6.3, it is demonstrated that the validation of the method is inspired by a real production environment, where processes are evaluated to explore potential development opportunities. In this section, the analysis data from the case study will be presented, along with the construction of the MDP model illustrated through a manufacturing example.

6.2 Methodology – RTLS-driven framework for AGV scheduling using Reinforcement Learning and Markov Decision Processes in micro-logistics

The Markov Decision Process serves as a mathematical framework designed to model decision-making within dynamic systems. Its application in reinforcement learning facilitates the acquisition of an optimal policy through the agent's interactions with the environment. The formal structure of the MDP is as follows:

- Assessment of current manufacturing and logistics processes.
- States: S is the set of possible states, where $S_i \in S$ is the i-th state.
- Rewards: R is the set of rewards, where $R(S_i, S_j$ is the reward received after the transition of the state from i -th state to j -th state.
- Decisions / Actions: A is the set of actions, where $A_k \in A$ is the k th action.
- Transition Probabilities: $P(S_j|S_i, A_k)$ is the probability that the system is in S_i state, and when A_k action is taken, the system will go to the next S_j state.
- Initial state: S_0 is the initial state of the MDP model.

The optimization process begins by defining the relevant states, actions, and rewards that are important to the specific domain of the problem. States represent various situations or configurations achievable by the system, actions indicate executable decisions, and rewards measure the immediate benefit or cost associated with a state-action combination. Based on this, the SV state vector (6.1) can be created, with each element representing the current states according to the structure of the system's MDP model at time t.

$$SV(t) = [S_1(t), S_2(t), S_3(t), ..., S_i(t), ..., S_{N_S}(t)] \tag{6.1}$$

where N_S is the number of the states. Based on the $SV(t)$ state vector, the $R(t)$ rewards vector obtained at time instant t can also be determined using Equation 6.2:

$$R(t) = [RS_1(t), RS_2(t), ..., RS_i(t), ..., RS_{NS}(t)] \tag{6.2}$$

The state vector and the associated reward vector can be determined at any moment in time following the state transitions created by the actions. Of course, this requires knowledge of the MDP model-based representation of the entire system. Optimization algorithms, including a type known as Reinforcement Learning, are used to improve decision strategies by using feedback in the form of rewards.

Through continuous interaction between the agent and the environment, the reinforcement learning algorithm acquires knowledge about associating specific actions with the maximization of cumulative rewards over time. This iterative learning process involves updating the agent's policy based on observed rewards and experiences. The MDP model enables the reinforcement learning algorithm to make informed decisions by considering probabilistic transitions between states and expected rewards associated with each action. The ultimate goal of optimization is to converge towards an optimal policy that maximizes the expected cumulative reward over time. The MDP model essentially serves as the foundational framework that guides the reinforcement learning algorithm throughout the optimization process. Provides a formal representation of the decision-making environment, allowing the algorithm to learn and adjust its policy to achieve optimal outcomes based on defined states and their corresponding rewards.

6.3 Exploring the application of RTLS data and reinforcement learning for AGV scheduling in micro-logistics – a case study

The framework for optimizing logistic processes using the Markov Decision Process is explored through a case study involving micro-logistic tasks performed by an Automated Guided Vehicle (AGV), highlighting its potential for improving efficiency in production environments.

Use case description

As illustrated in Figure 6.1, the production line consists of three automated production cells: welding, molding, and assembly. In addition, there is an automated warehouse serving as a buffer, a packaging area for transporting finished products, and a dedicated charging station for the AGV. The empty crates comprise 10 universally adaptable trays designed for storing semifinished and finished products in any condition. Within each production cell, there is a buffer in both the infeed and outfeed, each consisting of a single unit equivalent to 10 trays. Other raw materials, not transported by AGV, are stored in bulk within the machines and only need to be replenished once per shift. The following subsections delineate the components of the production process from the perspective of logistics services provided by AGV.

Within the welding production cell, there exists an inbound section (W_{i1}) and an outbound section (W_{o1}) designated for the storage of empty crates and the retrieval of welded semi-finished parts, respectively. The average cycle time of the machine is 5.8 seconds. The projected cycle times for the products are shown in Figure 6.2. The consecutive times of 0 and about 12 seconds in the figure are due to the technology where the machine handles two products at the same time.

Similarly, the molding production cell has an input area (M_{i1}) and an output area (M_{o1}) for the storage of welded semi-finished products and the retrieval of molded semi-finished products. The machine handles the transfer of the empty crate itself. The machine handles one product at a time during the production process, with an average cycle time of 4.6 seconds. The projected cycle times for the products are shown in Figure 6.3.

Inside the assembly production cell, there are two input areas and two output areas designated for the storage of semi-finished molded products (A_{i1}) and the retrieval of finished products (A_{o1}), respectively. The assembly cell does not have the capacity to store empty containers internally. Consequently, the AGV is required to transfer empty containers from the second output position (A_{o2}) to the second input position (A_{i2}). The average cycle time of the machine is 5.8 seconds. The projected

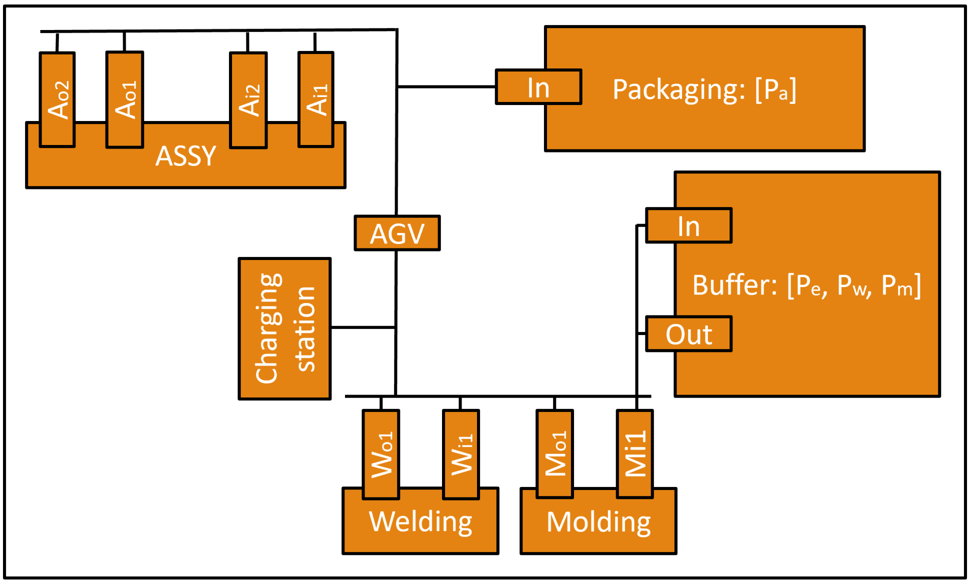


Figure 6.1: Schematic of the layout with automatic production cells, filling station, packer and buffer included in the logistics optimisation.

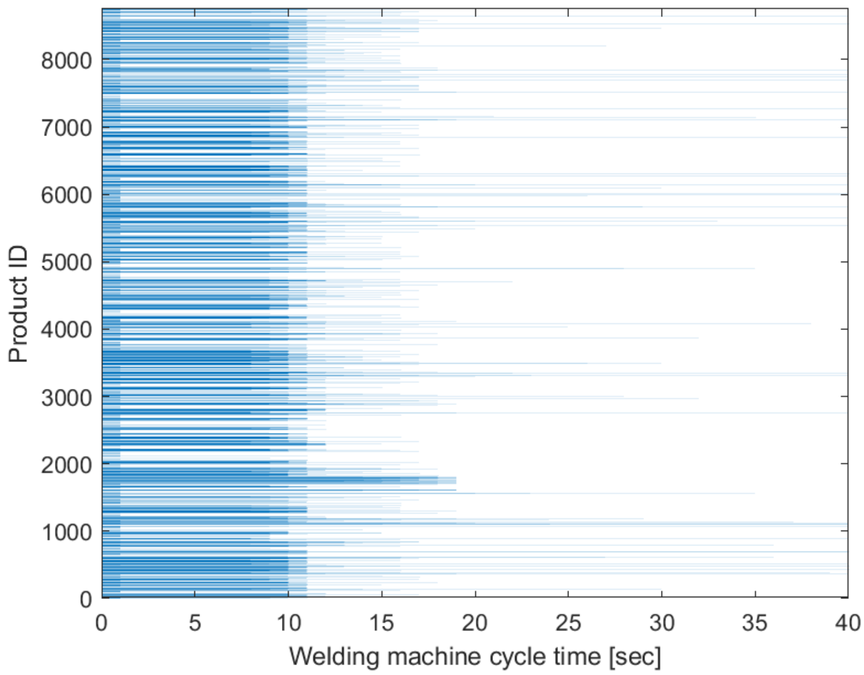


Figure 6.2: The automatic welding production cell product cycle time.

cycle times for the products are shown in Figure 6.4. The three consecutive times of 0 and about 21 seconds in the figure are due to the technology where the machine handles four products at the same time.

The process also includes a buffer that functions as an automated warehouse for the storage of empty crates (P_e) and semi-finished welded and molded products (P_w,

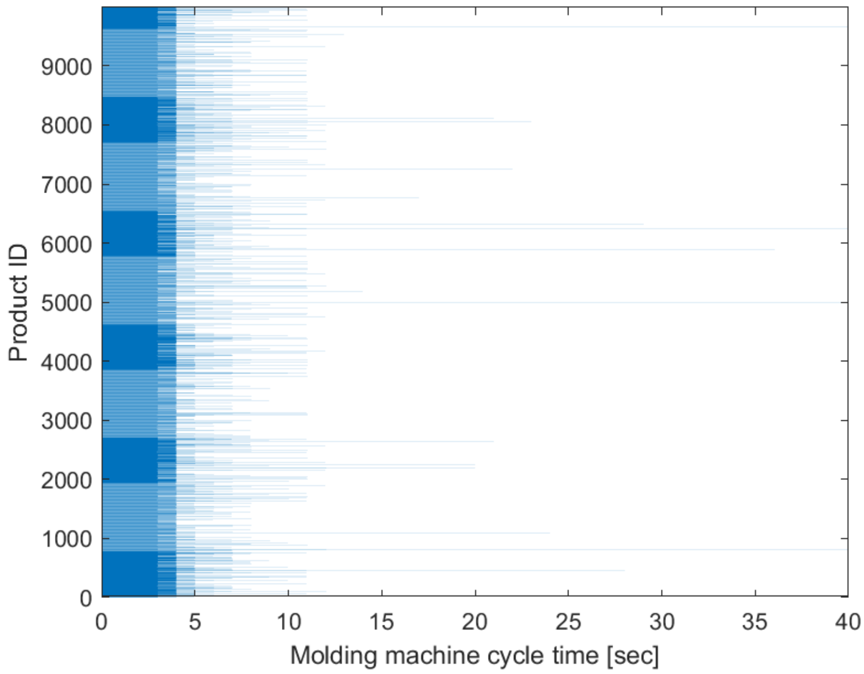


Figure 6.3: The automatic molding production cell product cycle time.

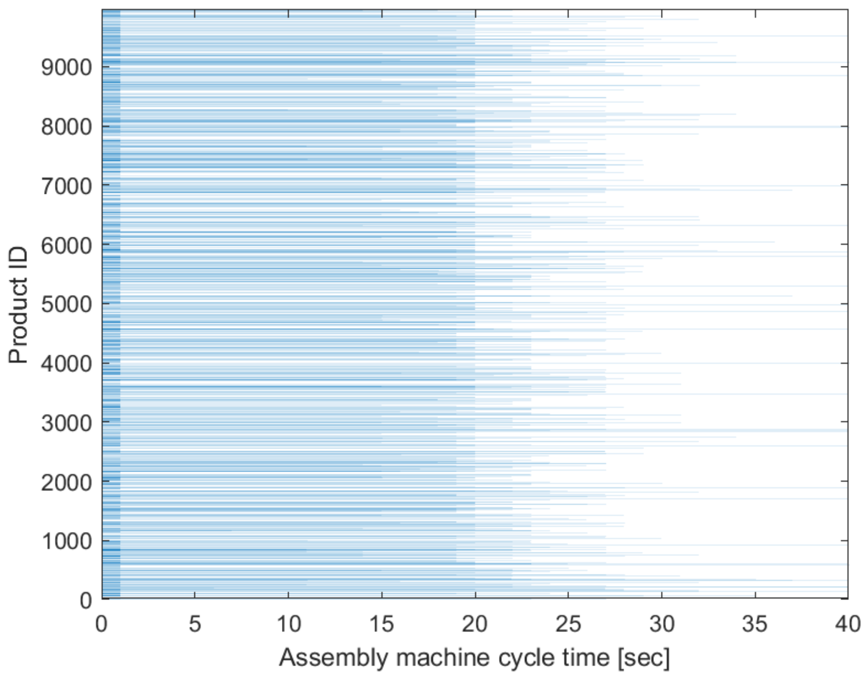


Figure 6.4: The automatic assembly production cell product cycle time.

P_m). There is also a packaging area for the transport of finished products (P_a), where the finished product crates are placed on the conveyor track at a fixed input point. During periods of inactivity, the AGV remains in a standby state, and when the state of charge (SOC) falls below a predetermined threshold, the AGV autonomously moves to the charging station.

The AGV performs the logistical material supply between these production elements in a scheduled manner, with the aim of eliminating production stoppages in any production cell and reducing nonvalue-added paths (idling). As mentioned in the introduction, it is worthwhile to study the logistics processes of the current system, as potential optimizations can simplify the AGV task distribution. Figure 6.5 shows the total lead times for each product on the line, including wait times. From this analysis, it is clear that a reduction in wait times is justified, and for the molded semi-finished product, it is advisable to maintain only a safety stock (which can be determined based on maintenance and historical data related to downtime), since the machine cycle time is fastest at this stage. The culmination of these analyzes supports a more efficient and more optimal inventory strategy.

Figure 6.6 shows the lead times separately, without waiting times. A closer look at the outlying time values revealed that during the period under study, an ad hoc machine maintenance shutdown occurred, which caused the semifinished products to remain in the declared state for a longer period of time. After filtering out these values, it can be said that the machining times are uniform (with acceptable variation due to manufacturing) throughout the production process.

In addition to identifying potential development opportunities that affect the MDP model, the time values obtained during the analysis of real production data can be used as variable parameters for the model's development. With knowledge

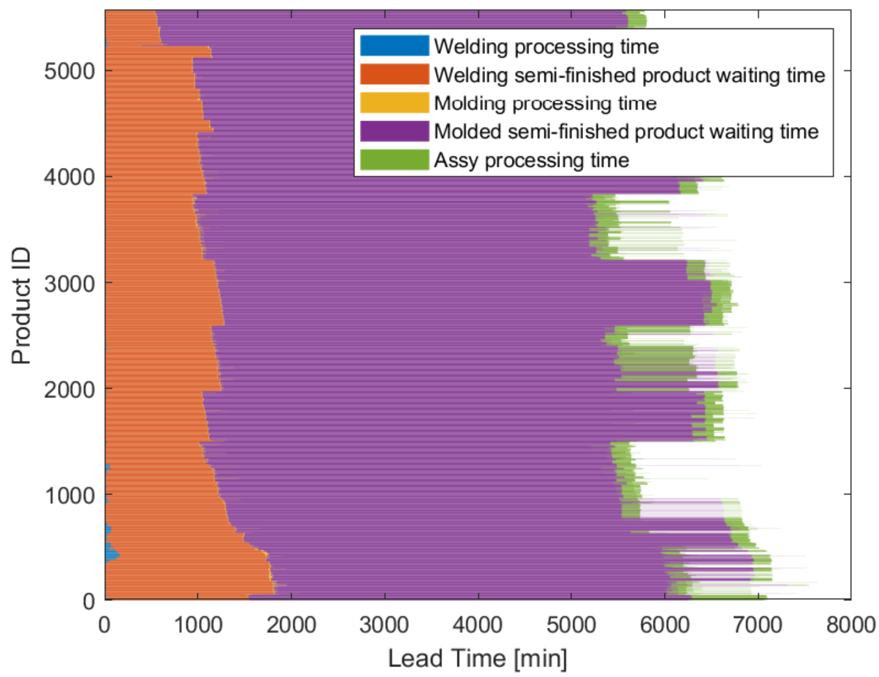


Figure 6.5: The lead time and the waiting time for the entire production process for each product.

of the production data and its variations (such as waste product generation and unexpected breakdowns), uncertainty can also be introduced into the system.

Results and discussion

Taking into account the production and logistics processes, the state of the system can be identified, a crucial step in the development of the MDP model. In this context, it is advisable to evaluate the state of each production cell based on the conditions of the input and output zones. These conditions play a key role in determining whether a production cell will stop operating, either due to lack of raw materials in the input section such as W_{i1}, M_{i1}, A_{i1} and A_{i2} or when the storage area in the output section such as W_{o1}, M_{o1}, A_{o1} and A_{o2} reaches full capacity. It is recommended to further refine these states according to the specific nuances of the current problem. The states for the input and output of each production cell can be defined in a manner similar to the summarized information provided in Table 6.1 and Table 6.2.

Furthermore, it is crucial to monitor the buffer inventory, the quantity of manufactured products, and the level of charge of the AGVs (see Table 6.3). These states play a key role in identifying feasible tasks (actions) that can be executed at any given moment.

Based on these factors and Equations 6.1 and 6.2, a $SV(t)$ state vector (6.3) and the $R(t)$ rewards vector 6.4 can be formulated to characterize the current state of the system at time t time:

$$\begin{aligned} SV(t) = [&SW_{i1}(t), SW_{o1}(t), SM_{i1}(t), SM_{o1}(t), \\ &SA_{i1}(t), SA_{i2}(t), SA_{o1}(t), SA_{o2}(t), \\ &SP_a(t), SP_w(t), SP_e(t), S_{AGV}(t)] \end{aligned} \tag{6.3}$$

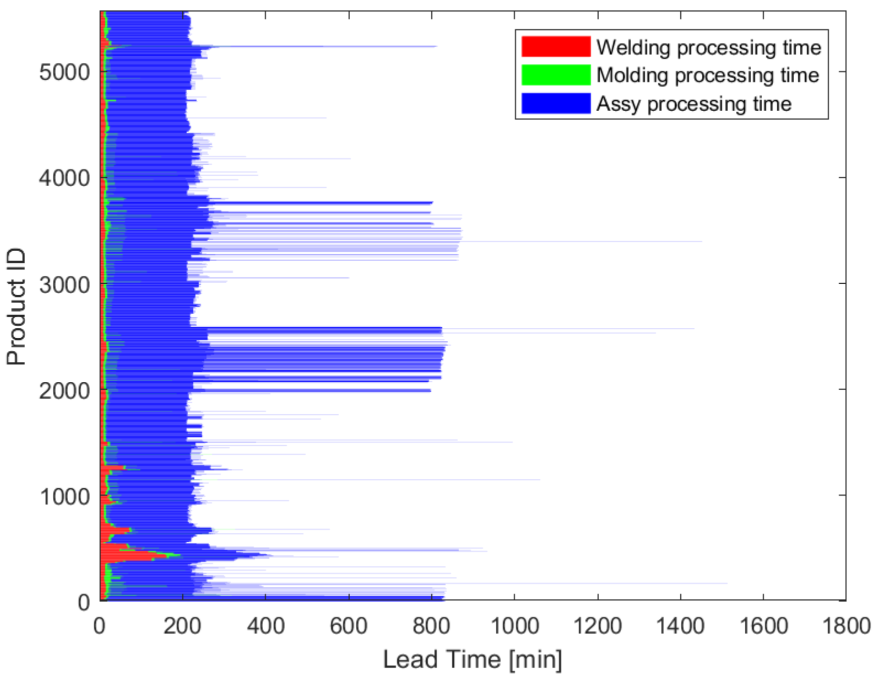


Figure 6.6: The lead time for the entire production process for each product.

State	Description	Reward
0	t = 0: production halted because no raw materials available	extra negative
1	0 < t <= 10: production stops within 10 minutes because no raw materials will be available within 10 minutes	negative
2	10 < t <= takt time: production is in the initial phase because raw materials will be available more than 10 minutes	positive
3	takt time < t <= 2 × takt time: production is ongoing and buffer is available	extra positive

Table 6.1: The states of the automatic production cells input zone (W_{i1}, M_{i1}, A_{i1} and A_{i2}) based on the availability of the raw material with the corresponding rewards.

State	Description	Reward
0	t = 0: production halted because picking out zone is not empty	extra negative
1	0 < t <= 10: production stops within 10 minutes because picking out zone will not be empty within 10 minutes	negative
2	10 < t <= takt time: production is in the initial phase because picking out zone will be empty for more than 10 minutes	positive
3	takt time < t <= 2 × takt time: production is ongoing and picking out zone is empty	extra positive

Table 6.2: The states of the automatic production cells output zone (W_{o1}, M_{o1}, A_{o1} and A_{o2}) based on the completion of the semi-finished or finished product with the corresponding rewards.

$$\begin{aligned} R(t) = [&RW_{i1}(t), RW_{o1}(t), RM_{i1}(t), RM_{o1}(t), \\ &RA_{i1}(t), RA_{i2}(t), RA_{o1}(t), RA_{o2}(t)] \end{aligned} \tag{6.4}$$

After establishing the states, it is crucial to analyze the possible state transitions and explore the methods by which they can be realized through different actions. The potential actions are listed in Table 6.4. When the action is completed, the AGV's location is determined along with the potential tasks it can perform at that specific location. For a complete understanding of the action network, please refer to Figure 6.7.

In the context of optimization, a profound understanding of the state of the system, its correlated rewards, and the potential actions that induce state transitions

State	Description
Pa	Number of finished product
Pw	Number of semi-finished welding products in the buffer
Pm	Number of semi-finished molding products in the buffer
Pe	Number of empty crates in the buffer
Agv	AGV SOC

Table 6.3: The states of other elements.

is essential. This fundamental understanding facilitates the creation of a state vector that encapsulates the system at a given time t, along with its associated reward. This plays a pivotal role in describing the dynamics of the system. The convergence of viable actions for AGVs depending on specific states delineates a repertoire of choices in a given scenario. The chosen action triggers a transition to a new state vector, facilitating the recalibration of the system's state and reward. This iterative process forms the crux of the reinforcement learning model within the Markov Decision Process framework. Through successive rounds of decisions, reward observations,

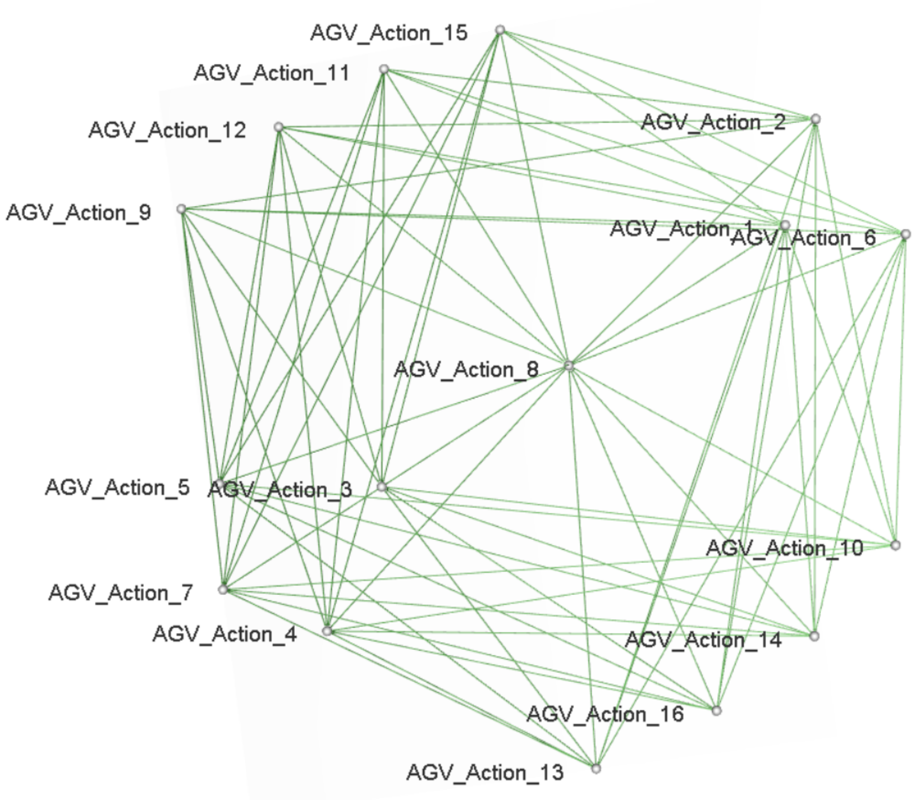


Figure 6.7: Actions connection diagram, which shows which action can theoretically be performed after a given action has been performed.

Action	**Description**
AGV Action 1	Picks up an empty container from the buffer storage.
AGV Action 2	Picks up a welding semi-finished product from the buffer storage.
AGV Action 3	Picks up a molding semi-finished product from the buffer storage.
AGV Action 4	Picks up a welding semi-finished product from the welding machine.
AGV Action 5	Picks up a molding semi-finished product from the molding machine.
AGV Action 6	Picks up an empty container from the assembly machine.
AGV Action 7	Picks up a finished product from the assembly machine.
AGV Action 8	The AGV stands empty, ready for loading.
AGV Action 9	Stores an empty container in the buffer storage.
AGV Action 10	Stores a welding semi-finished product in the buffer storage.
AGV Action 11	Stores a molding semi-finished product in the buffer storage.
AGV Action 12	Stores an empty container in the welding machine.
AGV Action 13	Stores a welding semi-finished product in the molding machine.
AGV Action 14	Stores an empty crate in the assembly machine.
AGV Action 15	Stores a molding semi-finished product in the assembly machine.
AGV Action 16	Stores a finished product in the packaging station.

Table 6.4: Actions of the AGV in the production process.

and policy updates, the reinforcement learning model systematically explores diverse scenarios. The ultimate goal is to converge towards an optimal policy that maximizes cumulative rewards across consecutive time steps. Furthermore, the incorporation of real-time indoor positioning data enriches this optimization paradigm, allowing continuous monitoring of production processes. These real-time data facilitate dynamic adjustments and enhancements in response to evolving conditions. The model's adaptability, rooted in the MDP structure, empowers it to optimize tasks based on specific criteria, such as distance metrics. Collaborative synergy between the MDP model and reinforcement learning holds promise in the approach to intricate optimization challenges, establishing a framework where intelligent decision making, guided by accumulated experience, contributes to the ongoing improvement and efficiency of logistics and manufacturing processes. In essence, the amalgamation of the MDP model and reinforcement learning not only provides a theoretical underpinning for decision making, but also imparts practical insights into real-time optimization, underscoring adaptability and continuous improvement in dynamic environments.

6.4 Summary of the chapter

The goal of this thesis is to explore advanced machine learning techniques, with a focus on reinforcement learning, to establish a framework for optimizing the integration and control of automated production and material supply in Industry 4.0 environments. By addressing the complexities of production processes and the heterogeneous nature of manufacturing environments, the research aims to provide a foundation for decision-making strategies that enhance efficiency and flexibility.

The principles and technologies of Industry 4.0 are transforming manufacturing paradigms by integrating digital technologies and automation into production processes. Automated production lines, in conjunction with supply chain management, improve efficiency and enable flexible responses to changing demands. However, the common integration and control of automated production and material supply is challenging due to complex production processes and heterogeneous environments. Machine learning methods, especially reinforcement learning algorithms, provide opportunities to address these challenges by developing optimal decision strategies using the available information.

The analysis of the production system provides insights into possible framework creation, such as an MDP model, for reinforcement learning-based optimization. The data analysis allows to create the states and state transitions caused by actions with associated rewards for building the model. The application from a case study point of view could help to understand the use of the method. The proposed approach can also be applied to different production optimization problems, which can be the subject of future research.

Chapter 7

Conclusion

The motivation for this thesis was to address the challenges of integrating and optimizing position data in manufacturing environments, with a particular focus on Real-Time Location Systems (RTLS). The research explored various methodologies to improve the accuracy of position data, identify value-added and non-value-added activities, and develop advanced decision-making strategies using machine learning algorithms such as reinforcement learning. The thesis presented four major approaches to improve data accuracy, optimize internal logistics, and facilitate AGV scheduling through reinforcement learning-based models. The contributions of each approach were discussed in detail, along with corresponding case studies that demonstrated their effectiveness.

The first approach focused on using RTLS positioning data in manufacturing environments to support process optimization and decision making. By applying data science techniques such as data mining and statistical analysis, the RTLS data provided insights into resource flows, operational inefficiencies, and process optimization opportunities. The results of this approach enabled manufacturing companies to better allocate resources, improve workflow efficiency, and enhance overall operational performance. This approach demonstrated the potential of RTLS data to drive data-driven decision making and contribute to increased productivity in manufacturing environments.

The second approach focused on improving the accuracy of position data in RTLS systems using a data reconciliation method. This method was applied to improve the reliability and accuracy of position data generated by UWB-based indoor positioning systems. By reconciling discrepancies in the data, the proposed method provided more accurate information for process monitoring and decision making. The effectiveness of this approach was validated through case studies in a warehouse environment, where the improvement in data accuracy led to better resource allocation and operational insights.

The third approach focused on identifying value-added and non-value-added activities through a multilayer network model applied to position data. This model, demonstrated in a taxi route optimization case study, is adaptable to different location data sets. By analyzing resource flows and activity patterns across multiple network layers, the methodology provides detailed insights into operational inefficiencies and potential improvements. The effectiveness of this approach was validated by the taxi data case study, which revealed opportunities to balance resource flows and reduce idle time, highlighting the potential of multilayer network analysis in optimizing transportation and logistics processes.

The fourth approach used a modified DBSCAN algorithm to classify activities based on position data, improving cluster separation by incorporating additional movement data. This methodology was applied to forklift movements in a real-world logistics environment, providing insight into operational patterns. The results showed significant opportunities to identify idle time and inefficiencies, laying the groundwork for potential improvements in resource utilization.

The fifth approach focused on establishing a framework for AGV scheduling in manufacturing environments using Reinforcement Learning and a Markov Decision Process framework. This RL-based approach aimed to provide a basis for the development of decision strategies capable of dynamically adapting to changes in production requirements and environmental conditions. The framework was explored in a case study that demonstrated the potential of RL models to improve task allocation, minimize idle time and improve overall efficiency in material handling processes.

Each of these approaches contributes to the broader goal of optimizing manufacturing processes through the use of advanced data analysis techniques and machine learning algorithms. The integration of RTLS positioning data, multilayer network models, the DBSCAN clustering algorithm, and reinforcement learning provides a robust framework for improving process efficiency and decision making in Industry 4.0 environments. Future research could extend these methods to other areas of production optimization, including the integration of additional machine learning models and the exploration of more complex manufacturing systems.

Chapter 8

Research questions and thesis findings

The thesis focuses on the utilization of positioning data in manufacturing environments, developing algorithms to support process optimization and data accuracy improvement. The main research questions are the following:

- *How can RTLS positioning data be utilized for process optimization and decision support in manufacturing environments?*
  In Chapter 2, I explore how RTLS positioning data can be leveraged using data science approaches to optimize manufacturing processes. This chapter highlights case studies where RTLS data supports better decision-making, improving operational efficiency through real-time tracking and analysis of assets.

- *How can position data accuracy be enhanced in RTLS applications for improved process insights?*
  In Chapter 3, I present a data reconciliation approach aimed at improving the accuracy of position data. This chapter details methods for mitigating errors in RTLS data, showcasing how enhanced data precision contributes to more reliable process monitoring and optimization in manufacturing environments.

- *How can algorithms identify value-added and non-value-added activities using position data?*
  Chapter 4 focuses on the development of algorithm that differentiate value-added and non-value-added activities using a multi-layer network approach. This methodology is illustrated through case study, a taxi route optimization, where position data is analyzed to balance and optimize resource flows. By applying a multi-layer network model, this approach provides insights into activity patterns and highlights opportunities for reducing inefficiencies and enhancing resource utilization.

- *How can clustering and network analysis classify value-added and non-value-added activities in intralogistics using position data?*
  Chapter 5 focuses on the development of a modified DBSCAN algorithm. This methodology is demonstrated through a case study in intralogistics, where position data is used to classify and optimize forklift movements. The approach enhances the identification of value-added and non-value-added activities,

enabling more efficient resource allocation and reducing idle times in logistics processes.

- *How can positioning data be utilized to optimize micro-logistics and material handling processes in manufacturing environments?*
  Chapter 6 explores the use of positioning data to provide a framework for optimizing micro-logistics and material handling processes. By applying reinforcement learning and Markov Decision Process (MDP) models, the chapter demonstrates the potential for improving AGV scheduling, resource utilization, and task execution in industrial environments.

I summarised the contributions to the development of algorithms and methods for utilizing RTLS positioning data in manufacturing, identifying value-added and non-value-added activities, and improving position data accuracy.

1. I developed a data science approach for utilizing RTLS positioning data in manufacturing environments for process optimization and decision support.

- I demonstrated how RTLS positioning data can be integrated into manufacturing workflows to provide real-time insights for process optimization and decision-making.
- I presented case studies where RTLS data has been applied to improve operational efficiency through better tracking and monitoring of assets.
- I introduced a framework for using data-driven decision support systems based on RTLS data to streamline processes and resource allocation.

Related publications:

1. András Rácz-Szabó, Tamás Ruppert, László Bántay, Andreas Löcklin, László Jakab, and János Abonyi. ”Real-time locating system in production management.” Sensors 20, no. 23 (2020): 6766. [[1]]

2. I developed a data reconciliation approach for enhancing the accuracy of RTLS position data for improved process insights.

- I demonstrated how data reconciliation techniques can improve the precision of RTLS position data in manufacturing environments.
- I presented examples of how more accurate position data leads to better process monitoring and error reduction in real-time operations.
- I introduced an approach for integrating data reconciliation into RTLS systems to continuously enhance data accuracy and reliability.

Related publications:

1. András Rácz-Szabó, Tamás Ruppert, and János Abonyi. Data reconciliation of indoor positioning data: Improve position data accuracy in warehouse environment. MethodsX. 2024 Dec 1;13:102838. [[230]]

3. I developed a multilayer-based algorithm for identifying value-added and non-value-added activities using position data.

- I demonstrated how position data can be used to distinguish between value-added and non-value-added activities through a multi-layer network approach.
- I presented case studies analyzing urban mobility patterns and traffic flows, which provided insights for balancing and optimizing resource flows.
- I introduced an analysis of urban mobility patterns and traffic flows to better understand and optimize resource utilization.

Related publications:

1. András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Multilayer Network-Based Evaluation of the Efficiency and Resilience of Network Flows." Complexity 2025, no. 1 (2025): 6940097. [[92]]

4. I developed a DBSCAN-based approach for identifying value-added and non-value-added activities using position data in an intralogistics environment.

- I demonstrated how position data, processed through a DBSCAN-based clustering algorithm, can differentiate between value-added and non-value-added activities within intralogistics operations.
- I presented case study focusing on the movement patterns of forklifts in a warehouse setting, providing insights into optimizing resource allocation and reducing idle times.
- I introduced a modified DBSCAN algorithm for refining the identification of non-value-added activities through position data analysis. This algorithm improved clustering precision by incorporating additional movement data and conditions, facilitating better task scheduling and resource utilization in manufacturing systems.

Related publications:

1. András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Clustering and network analysis of mobility patterns as an analysis tool for lean project." Emerging Science Journal 2025 Volume 9, Issue 01 [[231]]

5. I developed a framework for utilizing RTLS positioning data in manufacturing environments to support micro-logistics and material handling processes.

- I demonstrated how RTLS positioning data can be used as a basis for improving the scheduling and coordination of automated guided vehicles (AGVs) in complex manufacturing systems.
- I presented case studies that explored the potential of reinforcement learning and Markov Decision Process (MDP) models for enhancing AGV scheduling, resource utilization, and task allocation.

- I introduced a framework that integrates RTLS data with advanced algorithmic techniques, providing a foundation for actionable insights and improved decision-making in material handling operations.

Related publications:

1. András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Enhancing material supply for an automated production line by implementing a Markov Decision Process model for AGV-based material handling." In 2024 IEEE 22nd World Symposium on Applied Machine Intelligence and Informatics (SAMI), pp. 000193-000198. IEEE, 2024. [[232]]

2. András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Improving micro-logistics processes by indoor positioning system and the tools of data science."In PCS Science, ISBN 978-615-01-9235-2, published by Eclipse Solutions Ltd, 2023. Available at: http://pcsmeeting.hu/science/. [[86]]

Chapter 9

List of notations

Chapter 2: Utilizing RTLS positioning data in manufacturing: insights into process efficiency and operational applications

p - The current product.

z - The current zone.

T_p^z - The time spent by the current product (p) in the current zone (z).

$T_p^z(l)$ - The last timestamp of the current product (p) position data in the current zone (z).

$T_p^z(f)$ - The first timestamp of the current product (p) position data in the current zone (z).

Chapter 3: Enhancing position data accuracy in RTLS: A data reconciliation approach for improved process insights

$O = \{o_1, o_2, \ldots, o_{N_o}\}$ - Set of objects, where N_o is the number of objects.

$S = \{s_1, s_2, \ldots, s_{N_s}\}$ - Set of sensor tags, where N_s is the number of tags.

$N_l^{(o_j)}$ - Number of tags assigned to object o_j.

$\mathbf{A}$ - Matrix containing the relationships between sensors (S) and objects (O): 1 if a sensor belongs to an object, 0 otherwise.

$\mathbf{D}$ - Matrix describing the physical distances between sensors.

$\mathbf{P^l} \subseteq \mathbf{T^l} \times \mathbf{X^l}$ - Database containing position data for sensors.

$\mathbf{T^l} = \{t_1^l, t_2^l, \ldots, t_{N_d}^l\}$ - Timestamps of the l-th sensor.

$\mathbf{X^l} = \{\mathbf{x_1^l}, \mathbf{x_2^l}, \ldots, \mathbf{x_k^l}, \ldots, \mathbf{x_{N_d}^l}\}$ - 3D coordinate-based position data.

$\mathbf{x_k^l} = [x_{k,1}^l, x_{k,2}^l, x_{k,3}^l \quad \in \mathbb{R}^3]$ - Position data of the k-th measurement.

N_d^l - Number of recorded position data points for the l-th sensor.

$\mathbf{X}^{(l_{\text{interp}})}$ - Interpolated position data matrix for the l-th sensor at unique timestamps.

$\mathbf{X}_{\text{all}}$ - Position data matrix for all sensors.

$\mathbf{X}_{\text{all}}^{(\mathbf{o}_j)}$ - Position data of sensors belonging to object o_j.

$\mathbf{r}_n^{(\mathbf{o}_j)}$ - Error between measured and expected distances for object o_j at position n.

$\mathbf{R}_{\text{all}}$ - Matrix containing errors for all objects and positions.

$\mathbf{R}_x$ - Error cartography matrix.

p, q - x-axis and y-axis layout boundaries in the cartography matrix.

$N_{(p,q)}$ - Number of calculated errors for each $\mathbf{x}_{(p,q)}$ position.

$\mathbf{x}_{\mathbf{k}}^{(\mathbf{o_j})}$ - $M \times 1$ vector of measured position data for object o_j at k.

$\hat{\mathbf{x}}_k^{(\mathbf{o}_j)}$ - $M \times 1$ vector of reconciled position data for object o_j at k.

$\mathbf{V}_{\mathbf{x}_k}$ - $M \times M$ covariance matrix of the measurement errors.

f - $C \times 1$ vector representing the model's equality constraints (e.g., distance constraints).

$\mathbf{J}(\hat{\mathbf{x}}_k^{(\mathbf{o}_j)})$ - Objective function for minimizing errors during data reconciliation.

$d_{(x_k)}^{(o_j)}$ - Distance constraint for position x_k of object o_j.

$d_{(o_j)}$ - Measured distance between relevant sensors of object o_j.

Chapter 4. Algorithm development for identifying value-added and non-value-added activities using position data: a multi-layer network approach

T_k - Ordered list of activity attributes: $(v_s, v_e, t_s, t_e, r_l, l_m)$.

k - Index of activities (transactions), $k = 1 \ldots N$.

v_s, v_e - Starting and ending states of the activities, elements of the finite set $\mathbf{V}$.

$\mathbf{V}$ - Set of states of the system, $\mathbf{V} = \{v_s, v_e\}$.

N_v - Number of states, $N_v = |\mathbf{V}|$.

t_s, t_e - Starting and ending times of the state transitions.

r_l - Resource labels used to form layers of state transition networks, $r_l \in R$.

R - Set of resources required for activities.

l_m - Labels providing additional information on resource utilization (e.g., α for value-added, β for non-value-added).

M - Number of layers in the network, $l_m \in \{\alpha, \beta\} = \{1 \ldots M\}$.

G - Multilayer network, $G = (\mathbf{V}, E, D)$.

E - Set of labeled edges representing state transitions, $E = \mathbf{V} \times \mathbf{V}$.

D - Set of dimensions, (r_l, l_m) representing resource and utilization labels.

$E^{(r_l,l_m)}$ - Set of state transitions separated by r_l resources and l_m labels.

$A^{(r_l,l_m)}$ - Adjacency matrix representing intra-layer connections.

$a_{i,j}^{(r_l,l_m)}$ - Weight of state transitions for resource r_l from state i to j based on activity type l_m.

$Q_{i,j}^{\alpha}(t_1, t_2)$ - Number of finished state transitions in a given time interval per unit of time.

$k_{i,in}^{\alpha}, k_{i,out}^{\alpha}$ - In-degree and out-degree of a node in the value-added network.

$d_i^{\alpha}(t)$ - Difference between in-degree and out-degree at a given time, $d_i^{\alpha}(t) = k_{i,in}^{\alpha}(t) - k_{i,out}^{\alpha}(t)$.

$z_i^{\alpha}(t)$ - Node-level asymmetry value, $z_i^{\alpha}(t) = \frac{d_i^{\alpha}(t)}{k_{i,in}^{\alpha}(t)+k_{i,out}^{\alpha}(t)}$.

$Q_i^{\alpha}(t)$ - Inventory level of a node over time.

$C^{\beta}(t)$ - Total cost of non-value-added state transitions at time t.

$C^{\gamma}(t)$ - Total cost of optimized non-value-added state transitions.

$a_{i,j}^{\gamma}(t)$ - Optimized non-value-added state transition flows.

$O^{\beta,\gamma}(t)$ - Overlap of optimized and reconstructed non-value-added layers.

$c_{i,j}(t)$ - Cost of transportation for state transitions.

$d_i^{\gamma}(t)$ - Additional demand for balancing the network at node i.

$d_0^{\gamma}(t)$ - External source required to balance the network.

Chapter 5. Clustering and network analysis for identifying value-added and non-value-added activities using position data in intralogistics environment: A DBSCAN-based Approach

T_k - Database of recorded position data.

t_k - Time of the k-th recorded data.

$\mathbf{d}_k = [d_x, d_y, d_z \quad \in \mathbf{D}]$ - Position data vector of the k-th record.

r_l - Unique identification number of the resource.

$l_k \in \{\alpha, \beta\}$ - Activity flag for value-added ($\alpha = 0$) or non-value-added ($\beta = 1$) activities.

ϵ - Radius of the neighborhood in DBSCAN clustering.

MinPts - Minimum number of points required to form a cluster.

$\mathbf{W}$ - Vector defining the window size for the time-dependent data stream.

$\mathbf{D}_{(i:w)}$ - Data stream within the time window $i : w$ for clustering.

C - Set of clusters, where $C = \{C_1, C_2, \ldots, C_k\}$ and k is the number of created clusters.

C^j - Tuple of clusters based on l_k activity flag, separating value-added ($j = 0$) and non-value-added ($j = 1$) activities.

d_t - Limit of the active recording frequency.

D_a - Vector of active position data.

D_n - Vector of non-active position data.

D^a - Data stream contains all of active positions in chronological order based on t_k.

D^n - Data stream contains all of non-active positions in chronological order based on t_k.

m - Movement limit in the z-coordinate direction.

V - State vector based on layout-based clusters.

N_V - Number of unique states in the state vector.

E - State transition matrix.

$G = (V, E, F)$ - Multilayer network, where F represents the labels of resources and utilizations.

$G^\alpha = (V^\alpha, E^\alpha)$ - Multilayer network of value-added activities.

$A^\alpha = [a^\alpha_{p,q} \quad {}_{N_V \times N_V}]$ - Adjacency matrix of intra-layer connections for value-added activities.

$a^\alpha_{p,q}$ - Weight of state transitions from state p to q for value-added activities.

$a^{(\alpha,\beta)}_{p,q}$ - Interlayer connections between value-added and non-value-added activities.

$d_k(d_z)$ - Movement of the forklift in the z direction, determining whether the forklift is working.

δ_k - Data recording frequency difference: $\delta_k = t_k - t_{(k-1)}$.

Chapter 6. Exploring micro-logistics and material handling process optimization using RTLS positioning data: a Reinforcement Learning and Markov Decision Process framework for AGV scheduling

S - Set of possible states in the MDP model.

S_i - The i-th state in the set of states.

S_0 - Initial state of the MDP model.

A - Set of possible actions in the MDP model.

A_k - The k-th action in the set of actions.

R - Set of rewards associated with state transitions.

$R(S_i, S_j)$ - Reward received after transitioning from state S_i to state S_j.

$P(S_j|S_i, A_k)$ - Probability of transitioning to state S_j from state S_i after taking action A_k.

$SV(t)$ - State vector at time t, representing all current states of the system.

N_S - Number of states in the MDP model.

$R(t)$ - Reward vector at time t, corresponding to the rewards for each state.

W_{i1}, W_{o1} - Input and output zones for the welding production cell.

M_{i1}, M_{o1} - Input and output zones for the molding production cell.

$A_{i1}, A_{o1}, A_{i2}, A_{o2}$ - Input and output zones for the assembly production cell.

P_a - Packaging area for finished products.

P_w - Buffer for welded semi-finished products.

P_m - Buffer for molded semi-finished products.

P_e - Buffer for empty crates.

AGV - Automated Guided Vehicle responsible for logistics tasks.

SOC - State of Charge of the AGV.

$c_{i,j}(t)$ - Cost of transitioning from state i to state j at time t.

$SV(t)$ - State vector describing the system at time t.

$R(t)$ - Reward vector corresponding to the state transitions at time t.

Chapter 10

Bibliography

Publications by the Author

[[1]] András Rácz-Szabó, Tamás Ruppert, László Bántay, Andreas Löcklin, László Jakab, and János Abonyi. "Real-time locating system in production management". In: *Sensors* 20.23 (2020), p. 6766.

[[86]] András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Improving micrologistics processes using indoor positioning systems and data science tools". In: *PCS Science* (2023), pp. 15–20. URL: http://pcsmeeting.hu/science/.

[[92]] András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Multilayer Network-Based Evaluation of the Efficiency and Resilience of Network Flows". In: *Complexity* 2024.1 (2024), p. 6940097.

[[230]] András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Data reconciliation of indoor positioning data: Improve position data accuracy in warehouse environment". In: *MethodsX* 13 (2024), p. 102838.

[[231]] András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Clustering and network analysis of mobility patterns as an analysis tool for lean project". In: *Emerging Science Journal* 9.1 (2025).

[[232]] András Rácz-Szabó, Tamás Ruppert, and János Abonyi. "Enhancing material supply for an automated production line by implementing a Markov Decision Process model for AGV-based material handling". In: *2024 IEEE 22nd World Symposium on Applied Machine Intelligence and Informatics (SAMI)*. IEEE. 2024, pp. 000193–000198.

Bibliography

[2] Zhixin Yang, Pengbo Zhang, and Lei Chen. "RFID-enabled indoor positioning method for a real-time manufacturing execution system using OS-ELM". In: *Neurocomputing* 174 (2016), pp. 121–133.

[3] Guido Schroeer. "A real-time UWB multi-channel indoor positioning system for industrial scenarios". In: *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2018, pp. 1–5.

[4] Yanying Gu, Anthony Lo, and Ignas Niemegeers. "A survey of indoor positioning systems for wireless personal networks". In: *IEEE Communications surveys & tutorials* 11.1 (2009), pp. 13–32.

[5] Yu-Chi Pu and Pei-Chun You. "Indoor positioning system based on BLE location fingerprinting with classification approach". In: *Applied Mathematical Modelling* 62 (2018), pp. 654–663.

[6] Mohammad Shaifur Rahman, Md Mejbaul Haque, and Ki-Doo Kim. "High precision indoor positioning using lighting LED and image sensor". In: *14th International Conference on Computer and Information Technology (ICCIT 2011)*. IEEE. 2011, pp. 309–314.

[7] Yuehua Cai, Suleman Khalid Rai, and Hao Yu. "Indoor positioning by distributed machine-learning based data analytics on smart gateway network". In: *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2015, pp. 1–8.

[8] Carlos Medina, José Carlos Segura, and Angel De la Torre. "Ultrasound indoor positioning system based on a low-power wireless sensor network providing sub-centimeter accuracy". In: *Sensors* 13.3 (2013), pp. 3501–3526.

[9] Chung-Hao Huang, Lun-Hui Lee, Chian C Ho, Lang-Long Wu, and Zu-Hao Lai. "Real-time RFID indoor positioning system based on Kalman-filter drift removal and Heron-bilateration location estimation". In: *IEEE Transactions on Instrumentation and Measurement* 64.3 (2014), pp. 728–739.

[10] Suining He and S-H Gary Chan. "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons". In: *IEEE Communications Surveys & Tutorials* 18.1 (2015), pp. 466–490.

[11] Tsung-Nan Lin and Po-Chiang Lin. "Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks". In: *2005 international conference on wireless networks, communications and mobile computing*. Vol. 2. IEEE. 2005, pp. 1569–1574.

[12] Seoung-Hyeon Lee, Il-Kwan Lim, and Jae-Kwang Lee. “Method for improving indoor positioning accuracy using extended Kalman filter”. In: *Mobile Information Systems* 2016.1 (2016), p. 2369103.

[13] Ryoji Ban, Katsuhiko Kaji, Kei Hiroi, and Nobuo Kawaguchi. “Indoor positioning method integrating pedestrian Dead Reckoning with magnetic field and WiFi fingerprints”. In: *2015 Eighth international conference on mobile computing and ubiquitous networking (ICMU)*. IEEE. 2015, pp. 167–172.

[14] Jaegeol Yim, Seunghwan Jeong, Kiyoung Gwon, and Jaehun Joo. “Improvement of Kalman filters for WLAN based indoor tracking”. In: *Expert Systems with Applications* 37.1 (2010), pp. 426–433.

[15] Jijun Geng, Linyuan Xia, Jingchao Xia, Qianxia Li, Hongyu Zhu, and Yuezhen Cai. “Smartphone-based pedestrian dead reckoning for 3D indoor positioning”. In: *Sensors* 21.24 (2021), p. 8180.

[16] Wonho Kang and Youngnam Han. “SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization”. In: *IEEE Sensors journal* 15.5 (2014), pp. 2906–2916.

[17] Sinem Bozkurt, Gulin Elibol, Serkan Gunal, and Ugur Yayan. “A comparative study on machine learning algorithms for indoor positioning”. In: *2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA)*. IEEE. 2015, pp. 1–8.

[18] Yuan Zhuang, Zainab Syed, Jacques Georgy, and Naser El-Sheimy. “Autonomous smartphone-based WiFi positioning system by using access points localization and crowdsourcing”. In: *Pervasive and mobile computing* 18 (2015), pp. 118–136.

[19] Arto Perttula, Helena Leppäkoski, Martti Kirkko-Jaakkola, Pavel Davidson, Jussi Collin, and Jarmo Takala. “Distributed indoor positioning system with inertial measurements and map matching”. In: *IEEE Transactions on Instrumentation and Measurement* 63.11 (2014), pp. 2682–2695.

[20] Ahasanun Nessa, Bhagawat Adhikari, Fatima Hussain, and Xavier N Fernando. “A survey of machine learning for indoor positioning”. In: *IEEE access* 8 (2020), pp. 214945–214965.

[21] Po-Yen Tseng, Jacob J Lin, Ying-Chieh Chan, and Albert Y Chen. “Real-time indoor localization with visual SLAM for in-building emergency response”. In: *Automation in Construction* 140 (2022), p. 104319.

[22] Giuseppe Caso, Luca De Nardis, Filip Lemic, Vlado Handziski, Adam Wolisz, and Maria-Gabriella Di Benedetto. “ViFi: Virtual fingerprinting WiFi-based indoor positioning via multi-wall multi-floor propagation model”. In: *IEEE Transactions on Mobile Computing* 19.6 (2019), pp. 1478–1491.

[23] Pavel Davidson, Jussi Collin, and Jarmo Takala. “Application of particle filters for indoor positioning using floor plans”. In: *2010 Ubiquitous Positioning Indoor Navigation and Location Based Service*. IEEE. 2010, pp. 1–4.

[24] Moustafa Abbas, Moustafa Elhamshary, Hamada Rizk, Marwan Torki, and Moustafa Youssef. “WiDeep: WiFi-based accurate and robust indoor localization system using deep learning”. In: *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE. 2019, pp. 1–10.

[25] Fazli Subhan, Halabi Hasbullah, Azat Rozyyev, and Sheikh Tahir Bakhsh. “Indoor positioning in Bluetooth networks using fingerprinting and lateration approach”. In: *2011 International Conference on Information Science and Applications*. IEEE. 2011, pp. 1–9.

[26] Kamer Özgün, Melih Günay, Barış Doruk Başaran, Batuhan Bulut, Ege Yürüten, Fatih Baysan, and Melisa Kalemsiz. “Analysis of public transportation for efficiency”. In: *The International Conference on Artificial Intelligence and Applied Mathematics in Engineering*. Springer. 2020, pp. 680–695.

[27] Marcos Medina-Tapia, Francesc Robusté, and Miquel Estrada. “Modeling public transportation networks for a circular city: the role of urban subcenters and mobility density”. In: *Transportation Research Procedia* 47 (2020), pp. 353–360.

[28] Haisong Huang, Liguo Yao, and Chieh-Yuan Tsai. “Transportation service quality improvement through closed sequential pattern mining approach”. In: *Cybernetics and Information Technologies* 16.3 (2016), pp. 185–194.

[29] Jiaxin Chen, Wenzhu Liao, and Chengwei Yu. “Route optimization for cold chain logistics of front warehouses based on traffic congestion and carbon emission”. In: *Computers & Industrial Engineering* 161 (2021), p. 107663.

[30] Napong Wanichayapong, Wasawat Pruthipunyaskul, Wasan Pattara-Atikom, and Pimwadee Chaovalit. “Social-based traffic information extraction and classification”. In: *2011 11th International Conference on ITS Telecommunications*. IEEE. 2011, pp. 107–112.

[31] Rami Ibrahim and M Omair Shafiq. “Detecting taxi movements using Random Swap clustering and sequential pattern mining”. In: *Journal of Big Data* 6.1 (2019), pp. 1–26.

[32] AP Riascos and José L Mateos. “Networks and long-range mobility in cities: A study of more than one billion taxi trips in New York City”. In: *Scientific reports* 10.1 (2020), pp. 1–14.

[33] Dabin You and Ha Yoon Song. “Urban mobility model generation with public taxi transportation data”. In: *Proceedings of the 15th International Conference on Advances in Mobile Computing & Multimedia*. 2017, pp. 13–21.

[34] Yanchi Liu, Chuanren Liu, Nicholas Jing Yuan, Lian Duan, Yanjie Fu, Hui Xiong, Songhua Xu, and Junjie Wu. “Intelligent bus routing with heterogeneous human mobility patterns”. In: *Knowledge and Information Systems* 50.2 (2017), pp. 383–415.

[35] Angela Di Febbraro and Simona Sacone. “An online information system to balance traffic flows in urban areas”. In: *Proceedings of the 36th IEEE Conference on Decision and Control*. Vol. 5. IEEE. 1997, pp. 4772–4773.

[36] Jian Wen, Jinhua Zhao, and Patrick Jaillet. “Rebalancing shared mobility-on-demand systems: A reinforcement learning approach”. In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. Ieee. 2017, pp. 220–225.

[37] Zhiguo Zheng, Yunfeng Chen, Debao Zhu, Huijun Sun, Jianjun Wu, Xing Pan, and Daqing Li. “Extreme unbalanced mobility network in bike sharing system”. In: *Physica A: Statistical Mechanics and its Applications* 563 (2021), p. 125444.

[38] Javier Alonso-Mora, Samitha Samaranayake, Alex Wallar, Emilio Frazzoli, and Daniela Rus. “On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment”. In: *Proceedings of the National Academy of Sciences* 114.3 (2017), pp. 462–467.

[39] Hao Wang, Kai Zhang, Junhua Chen, Zhifeng Wang, Guijun Li, and Yuqi Yang. “System dynamics model of taxi management in metropolises: Economic and environmental implications for Beijing”. In: *Journal of environmental management* 213 (2018), pp. 555–565.

[40] Sukru Dursun, Celalettin Ozdemir, Hakan Karabork, and Saim Koçak. “Noise pollution and map of Konya city in Turkey”. In: *Journal of International Environmental Application & Science* 1.1 (2006), pp. 63–72.

[41] Inesa Mikhno, Viktor Koval, Galyna Shvets, Oksana Garmatiuk, and Rima Tamošiūnienė. “Green economy in sustainable development and improvement of resource efficiency”. In: *Central European Business Review (CEBR)* 10.1 (2021), pp. 99–113.

[42] R Sundar, AN Balaji, and RM Satheesh Kumar. “A review on lean manufacturing implementation techniques”. In: *Procedia Engineering* 97 (2014), pp. 1875–1885.

[43] Rene T Domingo. “Identifying and eliminating the seven wastes or muda”. In: *Asian Institute of Management, godina nepoznata* (2015).

[44] Sandeep J Pavnaskar, John K Gershenson, and Anil B Jambekar. “Classification scheme for lean manufacturing tools”. In: *International Journal of Production Research* 41.13 (2003), pp. 3075–3090.

[45] Peide Liu, Ayad Hendalianpour, Mohammadreza Feylizadeh, and Witold Pedrycz. “Mathematical modeling of vehicle routing problem in omni-channel retailing”. In: *Applied Soft Computing* 131 (2022), p. 109791.

[46] Peide Liu, Ayad Hendalianpour, and Mohammad Hamzehlou. “Pricing model of two-echelon supply chain for substitutable products based on double-interval grey-numbers”. In: *Journal of Intelligent & Fuzzy Systems* 40.5 (2021), pp. 8939–8961.

[47] Ayad Hendalianpour. “Optimal lot-size and price of perishable goods: a novel game-theoretic model using double interval grey numbers”. In: *Computers & Industrial Engineering* 149 (2020), p. 106780.

[48] Peide Liu, Ayad Hendalianpour, Jafar Razmi, and Mohamad Sadegh Sangari. “A solution algorithm for integrated production-inventory-routing of perishable goods with transshipment and uncertain demand”. In: *Complex & Intelligent Systems* 7 (2021), pp. 1349–1365.

[49] Ayad Hendalianpour, Mohammad Hamzehlou, Mohammad Reza Feylizadeh, Naiming Xie, and Mohammad Hossein Shakerizadeh. "Coordination and competition in two-echelon supply chain using grey revenue-sharing contracts". In: *Grey Systems: Theory and Application* 11.4 (2021), pp. 681–706.

[50] Shinan Zhao, Ayad Hendalianpour, and Peide Liu. "Blockchain Technology in Omnichannel Retailing: A Novel Fuzzy Large-Scale Group-DEMATEL & Ordinal Priority Approach". In: *Expert Systems with Applications* (2024), p. 123485.

[51] Richard B Detty and Jon C Yingling. "Quantifying benefits of conversion to lean manufacturing with discrete event simulation: a case study". In: *International journal of production research* 38.2 (2000), pp. 429–445.

[52] Amr Mahfouz, John Crowe, and Amr Arisha. "Integrating current state and future state value stream mapping with discrete event simulation: a lean distribution case study". In: Technological University Dublin, 2011.

[53] Shitong Peng, Tao Li, Jiali Zhao, Yanchun Guo, Shengping Lv, George Z Tan, and Hongchao Zhang. "Petri net-based scheduling strategy and energy modeling for the cylinder block remanufacturing under uncertainty". In: *Robotics and Computer-Integrated Manufacturing* 58 (2019), pp. 208–219.

[54] MK Tiwari, Niraj Sinha, Shailendra Kumar, Rahul Rai, and SK Mukhopadhyay. "A Petri net based approach to determine the disassembly strategy of a product". In: *International Journal of Production Research* 40.5 (2002), pp. 1113–1129.

[55] Yanyan Gu and Yandong Wang. "Using weighted multilayer networks to uncover scaling of public transport system". In: *Environment and Planning B: Urban Analytics and City Science* (2022), p. 23998083211062905.

[56] Alberto Aleta, Sandro Meloni, and Yamir Moreno. "A multilayer perspective for the analysis of urban transportation systems". In: *Scientific reports* 7.1 (2017), pp. 1–9.

[57] Riccardo Gallotti and Marc Barthelemy. "The multilayer temporal network of public transport in Great Britain". In: *Scientific data* 2.1 (2015), pp. 1–8.

[58] Diego Bogado Tomasiello, Mariana Giannotti, Renato Arbex, and Clodoveu Davis. "Multi-temporal transport network models for accessibility studies". In: *Transactions in GIS* 23.2 (2019), pp. 203–223.

[59] Mingyi He, Shivam Pathak, Urwa Muaz, Jingtian Zhou, Saloni Saini, Sergey Malinchik, and Stanislav Sobolevsky. "Pattern and anomaly detection in urban temporal networks". In: *arXiv preprint arXiv:1912.01960* (2019).

[60] Wen-Bo Du, Xing-Lian Zhou, Oriol Lordan, Zhen Wang, Chen Zhao, and Yan-Bo Zhu. "Analysis of the Chinese Airline Network as multi-layer networks". In: *Transportation Research Part E: Logistics and Transportation Review* 89 (2016), pp. 108–116.

[61] Xinglong Wang, Jinhan Peng, Junqing Tang, Qiuchen Lu, and Xiaowei Li. "Investigating the impact of adding new airline routes on air transportation resilience in China". In: *Transport Policy* 125 (2022), pp. 79–95.

[62] Alessio Cardillo, Massimiliano Zanin, Jesús Gómez-Gardenes, Miguel Romance, Alejandro J García del Amo, and Stefano Boccaletti. "Modeling the multi-layer nature of the European Air Transport Network: Resilience and passengers re-scheduling under random failures". In: *The European Physical Journal Special Topics* 215 (2013), pp. 23–33.

[63] Rui Ding, Norsidah Ujang, Hussain bin Hamid, Mohd Shahrudin Abd Manan, Yuou He, Rong Li, and Jianjun Wu. "Detecting the urban traffic network structure dynamics through the growth and analysis of multi-layer networks". In: *Physica A: Statistical Mechanics and its Applications* 503 (2018), pp. 800–817.

[64] Lu Xu, Funing Yang, Zhezhou Yu, Qiuyang Huang, Xuan Ji, and Yuanbo Xu. "Analysis of public transport system efficiency based on multi-layer network". In: *Artificial Intelligence and Security: 6th International Conference, ICAIS 2020, Hohhot, China, July 17–20, 2020, Proceedings, Part I 6*. Springer. 2020, pp. 798–809.

[65] Mehmet Yildirimoglu and Jiwon Kim. "Identification of communities in urban mobility networks using multi-layer graphs of network traffic". In: *Transportation Research Part C: Emerging Technologies* 89 (2018), pp. 254–267.

[66] Yanni Ju, Haihua Yuan, Zongping Li, Mi Gan, and Yufan Chen. "Multi-layer structures and resilience evaluation for multimode regional rail transit system". In: *IET Intelligent Transport Systems* 16.7 (2022), pp. 843–859.

[67] Aniekan Essien, Ilias Petrounias, Pedro Sampaio, and Sandra Sampaio. "Improving urban traffic speed prediction using data source fusion and deep learning". In: *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE. 2019, pp. 1–8.

[68] Gaurav Meena, Deepanjali Sharma, and Mehul Mahrishi. "Traffic prediction for intelligent transportation system using machine learning". In: *2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE)*. IEEE. 2020, pp. 145–148.

[69] Wen-Bo Du, Xing-Lian Zhou, Marko Jusup, and Zhen Wang. "Physics of transportation: Towards optimal capacity using the multilayer network framework". In: *Scientific reports* 6.1 (2016), p. 19059.

[70] Camill Harter, Otto Koppius, and Rob Zuidwijk. "Vulnerability of transport networks under disruption of their collaborations: A multi-layer network model". In: *Available at SSRN 4032613* (2022).

[71] Pengcheng Zhang, Srinivas Peeta, and Terry Friesz. "Dynamic game theoretic model of multi-layer infrastructure networks". In: *Networks and Spatial Economics* 5 (2005), pp. 147–178.

[72] Abdullahi Adinoyi Ibrahim, Daniela Leite, and Caterina De Bacco. "Sustainable optimal transport in multilayer networks". In: *Physical Review E* 105.6 (2022), p. 064302.

[73] Apichat Sopadang, Sooksiri Wichaisri, and Aicha Sekhari. "The conceptual framework of lean sustainable logistics". In: *Materiały z konferencji,International Conference on Transportation and Logistics*. 2014.

[74] Vinicius Mitsuo Kojima Campos, Syntia Lemos Cotrim, Edwin Vladmir Cardoza Galdamez, and Gislaine Camila Lapasini Leal. "Introduction of lean manufacturing philosophy by kaizen event: case study on a metalmechanical industry". In: *Independent Journal of Management & Production* 7.1 (2016), pp. 151–167.

[75] Fabiana Dafne Cifone, Kai Hoberg, Matthias Holweg, and Alberto Portioli Staudacher. "'Lean 4.0': How can digital technologies support lean practices?" In: *International Journal of Production Economics* 241 (2021), p. 108258.

[76] David Romero, Paolo Gaiardelli, Daryl Powell, Thorsten Wuest, and Matthias Thürer. "Digital lean cyber-physical production systems: the emergence of digital lean manufacturing and the significance of digital waste". In: *Advances in Production Management Systems. Production Management for Data-Driven, Intelligent, Collaborative, and Sustainable Manufacturing: IFIP WG 5.7 International Conference, APMS 2018, Seoul, Korea, August 26-30, 2018, Proceedings, Part I*. Springer. 2018, pp. 11–20.

[77] Mohd Javaid, Abid Haleem, Ravi Pratap Singh, Shanay Rab, Rajiv Suman, and Shahbaz Khan. "Exploring relationships between Lean 4.0 and manufacturing industry". In: *Industrial Robot: the international journal of robotics research and application* 49.3 (2022), pp. 402–414.

[78] Sajede Aminzadegan, Mohammad Tamannaei, and Morteza Rasti-Barzoki. "Multi-agent supply chain scheduling problem by considering resource allocation and transportation". In: *Computers & Industrial Engineering* 137 (2019), p. 106003.

[79] Peter Szabó, Samuel Janík, Miroslava MĹkva, and György Czifra. "Waste Elimination in the Assembly Process using Lean Tools". In: *Acta Polytechnica Hungarica* 21.4 (2024).

[80] Sisay G Gebeyehu, Muluken Abebe, and Amdework Gochel. "Production lead time improvement through lean manufacturing". In: *Cogent Engineering* 9.1 (2022), p. 2034255.

[81] Nuno Correia, Leonor Teixeira, and Ana Luísa Ramos. "Implementing an AGV system to transport finished goods to the warehouse". In: *Advances in Science, Technology and Engineering Systems* 5.2 (2020), pp. 241–247.

[82] Yang-Hua Lian and Hendrik Van Landeghem. "An application of simulation and value stream mapping in lean manufacturing". In: *Proceedings 14th European Simulation Symposium*. c) SCS Europe BVBA. 2002, pp. 1–8.

[83] Yolandi Schoeman, Paul Oberholster, and Vernon Somerset. "Value stream mapping as a supporting management tool to identify the flow of industrial waste: A case study". In: *Sustainability* 13.1 (2020), p. 91.

[84] K Venkataraman, B Vijaya Ramnath, V Muthu Kumar, and C Elanchezhian. "Application of value stream mapping for reduction of cycle time in a machining process". In: *Procedia Materials Science* 6 (2014), pp. 1187–1196.

[85] Changhong Pan and Meng Liu. “Optimization of intelligent logistics supply chain management system based on wireless sensor network and RFID technology”. In: *Journal of Sensors* 2021.1 (2021), p. 8111909.

[87] David Romero, Paolo Gaiardelli, Thorsten Wuest, Daryl Powell, and Matthias Thürer. “New forms of Gemba walks and their digital tools in the digital lean manufacturing world”. In: *Advances in Production Management Systems. Towards Smart and Digital Manufacturing: IFIP WG 5.7 International Conference, APMS 2020, Novi Sad, Serbia, August 30–September 3, 2020, Proceedings, Part II.* Springer. 2020, pp. 432–440.

[88] ZahirAbbas N Khaswala and Shahrukh A Irani. “Value network mapping (VNM): visualization and analysis of multiple flows in value stream maps”. In: *Proceedings of the Lean Management Solutions Conference.* Vol. 614. 2001, pp. 1–18.

[89] Mojtaba A Farahani, MR McCormick, Ramy Harik, and Thorsten Wuest. “Time-series classification in smart manufacturing systems: An experimental evaluation of state-of-the-art machine learning algorithms”. In: *Robotics and Computer-Integrated Manufacturing* 91 (2025), p. 102839.

[90] Konstantin Muehlbauer, Maximilian Wuennenberg, Sebastian Meissner, and Johannes Fottner. “Data driven logistics-oriented value stream mapping 4.0: A guideline for practitioners”. In: *IFAC-PapersOnLine* 55.16 (2022), pp. 364–369.

[91] Andreas Löcklin, Falk Dettinger, Maurice Artelt, Nasser Jazdi, and Michael Weyrich. “Trajectory prediction of workers to improve AGV and AMR operation based on the manufacturing schedule”. In: *Procedia CIRP* 107 (2022), pp. 283–288.

[93] Wilfried Lepuschitz, Vaclav Jirkovsky, Petr Kadera, and Pavel Vrba. “A multi-layer approach for failure detection in a manufacturing system based on automation agents”. In: *2012 Ninth International Conference on Information Technology-New Generations.* IEEE. 2012, pp. 1–6.

[94] Jie Zhou. “Logistics inventory optimization method for agricultural e-commerce platforms based on a multilayer feedforward neural network”. In: *Pakistan Journal of Agricultural Sciences* 60 (2023), pp. 487–496.

[95] Ilias P Vlachos, Rodrigo Martinez Pascazzi, George Zobolas, Panagiotis Repoussis, and Mihalis Giannakis. “Lean manufacturing systems in the area of Industry 4.0: A lean automation plan of AGVs/IoT integration”. In: *Production planning & control* 34.4 (2023), pp. 345–358.

[96] G Saravanan, Shailesh S Parkhe, Chetan M Thakar, Vaibhav V Kulkarni, Hari Govind Mishra, and G Gulothungan. “Implementation of IoT in production and manufacturing: An Industry 4.0 approach”. In: *Materials Today: Proceedings* 51 (2022), pp. 2427–2430.

[97] Giuseppe Fragapane, Dmitry Ivanov, Mirco Peron, Fabio Sgarbossa, and Jan Ola Strandhagen. “Increasing flexibility and productivity in Industry 4.0 production networks with autonomous mobile robots and smart intralogistics”. In: *Annals of operations research* 308.1 (2022), pp. 125–143.

[98] Krisztina Demeter and Zsolt Matyusz. "The impact of lean practices on inventory turnover". In: *International journal of production economics* 133.1 (2011), pp. 154–163.

[99] Norbert Szántó, Adám Csapó, and Ildikó Horváth. "Information basis of digital twins: A quantifiable metric for spatio-temporal expressivity". In: *Acta Polytechnica Hungarica* 20.6 (2023), pp. 151–171.

[100] Hendrik Sebastian Birkel and Evi Hartmann. "Impact of IoT challenges and risks for SCM". In: *Supply Chain Management: An International Journal* 24.1 (2019), pp. 39–61.

[101] Tomáš Lojka, Martin Miškuf, and Iveta Zolotová. "Industrial iot gateway with machine learning for smart manufacturing". In: *Advances in Production Management Systems. Initiatives for a Sustainable World: IFIP WG 5.7 International Conference, APMS 2016, Iguassu Falls, Brazil, September 3-7, 2016, Revised Selected Papers*. Springer. 2016, pp. 759–766.

[102] Juan Pablo Usuga Cadavid, Samir Lamouri, Bernard Grabot, Robert Pellerin, and Arnaud Fortin. "Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0". In: *Journal of Intelligent Manufacturing* 31 (2020), pp. 1531–1558.

[103] Marcel Panzer and Benedict Bender. "Deep reinforcement learning in production systems: a systematic literature review". In: *International Journal of Production Research* 60.13 (2022), pp. 4316–4341.

[104] Kyu Tae Park, Yoo Ho Son, Sang Wook Ko, and Sang Do Noh. "Digital twin and reinforcement learning-based resilient production control for micro smart factory". In: *Applied Sciences* 11.7 (2021), p. 2977.

[105] Martin L Puterman. "Markov decision processes". In: *Handbooks in operations research and management science* 2 (1990), pp. 331–434.

[106] Douglas J White. "A survey of applications of Markov decision processes". In: *Journal of the operational research society* 44.11 (1993), pp. 1073–1096.

[107] Walter L Boyajian, Jens Clausen, Lea M Trenkwalder, Vedran Dunjko, and Hans J Briegel. "On the convergence of projective-simulation–based reinforcement learning in Markov decision processes". In: *Quantum machine intelligence* 2 (2020), pp. 1–21.

[108] Kiran V Sagar and J Jerald. "Real-time automated guided vehicles scheduling with Markov decision process and double q-learning algorithm". In: *Materials Today: Proceedings* 64 (2022), pp. 279–284.

[109] Marco A Wiering and Martijn Van Otterlo. "Reinforcement learning". In: *Adaptation, learning, and optimization* 12.3 (2012), p. 729.

[110] Casey C Bennett and Kris Hauser. "Artificial intelligence framework for simulating clinical decision-making: A Markov decision process approach". In: *Artificial intelligence in medicine* 57.1 (2013), pp. 9–19.

[111] Shalabh Bhatnagar, Emmanuel Fernández-Gaucherand, Michael C Fu, Ying He, and Steven I Marcus. "A Markov decision process model for capacity expansion and allocation". In: *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304)*. Vol. 2. IEEE. 1999, pp. 1380–1385.

[112] Giuseppe Fragapane, Rene De Koster, Fabio Sgarbossa, and Jan Ola Strandhagen. “Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda”. In: *European Journal of Operational Research* 294.2 (2021), pp. 405–426.

[113] Ramon F Brena, Juan Pablo García-Vázquez, Carlos E Galván-Tejada, David Muñoz-Rodriguez, Cesar Vargas-Rosales, and James Fangmeyer. “Evolution of indoor positioning technologies: A survey”. In: *Journal of Sensors* 2017 (2017).

[114] Samer S Saab and Zahi S Nakad. “A standalone RFID indoor positioning system using passive tags”. In: *IEEE Transactions on Industrial Electronics* 58.5 (2010), pp. 1961–1970.

[115] Jacqueline Lee Fang Ang, Wai Kong Lee, and Boon Yaik Ooi. “GreyZone: A Novel Method for Measuring and Comparing Various Indoor Positioning Systems”. In: *2019 International Conference on Green and Human Information Technology (ICGHIT)*. IEEE. 2019, pp. 30–35.

[116] Yuwei Chen, Jian Tang, Changhui Jiang, Lingli Zhu, Matti Lehtomäki, Harri Kaartinen, Risto Kaijaluoto, Yiwu Wang, Juha Hyyppä, Hannu Hyyppä, et al. “The accuracy comparison of three simultaneous localization and mapping (SLAM)-based indoor mapping technologies”. In: *Sensors* 18.10 (2018), p. 3228.

[117] Anum Hameed and Hafiza Anisa Ahmed. “Survey on indoor positioning applications based on different technologies”. In: *2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*. IEEE. 2018, pp. 1–5.

[118] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu. “Survey of wireless indoor positioning techniques and systems”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37.6 (2007), pp. 1067–1080.

[119] Luka Batistić and Mladen Tomic. “Overview of indoor positioning system technologies”. In: *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE. 2018, pp. 0473–0478.

[120] Wisam Hasan Ali, Amnah Abas Kareem, and Manal Jasim. “Survey on Wireless Indoor Positioning Systems”. In: *Cihan University-Erbil Scientific Journal* 3.2 (2019), pp. 42–47.

[121] Abdulrahman Alarifi, AbdulMalik Al-Salman, Mansour Alsaleh, Ahmad Alnafessah, Suheer Al-Hadhrami, Mai A Al-Ammar, and Hend S Al-Khalifa. “Ultra wideband indoor positioning technologies: Analysis and recent advances”. In: *Sensors* 16.5 (2016), p. 707.

[122] Faheem Zafari, Athanasios Gkelias, and Kin K Leung. “A survey of indoor localization systems and technologies”. In: *IEEE Communications Surveys & Tutorials* 21.3 (2019), pp. 2568–2599.

[123] Wilson Sakpere, Michael Adeyeye-Oshin, and Nhlanhla BW Mlitwa. “A state-of-the-art survey of indoor positioning and navigation systems and technologies”. In: *South African Computer Journal* 29.3 (2017), pp. 145–197.

[124] Guowei Shi and Ying Ming. "Survey of indoor positioning systems based on ultra-wideband (UWB) technology". In: *Wireless Communications, Networking and Applications*. Springer, 2016, pp. 1269–1278.

[125] George Q Huang, YF Zhang, and PY Jiang. "RFID-based wireless manufacturing for walking-worker assembly islands with fixed-position layouts". In: *Robotics and Computer-Integrated Manufacturing* 23.4 (2007), pp. 469–477.

[126] Germán Martín Mendoza-Silva, Joaquín Torres-Sospedra, and Joaquín Huerta. "A meta-review of indoor positioning systems". In: *Sensors* 19.20 (2019), p. 4507.

[127] Jianga Shang, Shengsheng Yu, and Liangfeng Zhu. "Location-aware systems for short-range wireless networks". In: *2009 International Symposium on Computer Network and Multimedia Technology*. IEEE. 2009, pp. 1–5.

[128] B Gladysz and K Santarek. "An approach to RTLS selection". In: *DEStech Transactions on Engineering and Technology Research* (2017).

[129] Miguel Saez, Francisco P Maturana, Kira Barton, and Dawn M Tilbury. "Real-time manufacturing machine and system performance monitoring using internet of things". In: *IEEE Transactions on Automation Science and Engineering* 15.4 (2018), pp. 1735–1748.

[130] Davide Macagnano, Giuseppe Destino, and Giuseppe Abreu. "Indoor positioning: A key enabling technology for IoT applications". In: *2014 IEEE World Forum on Internet of Things (WF-IoT)*. IEEE. 2014, pp. 117–118.

[131] Shuo Li and Rashid Rashidzadeh. "Hybrid indoor location positioning system". In: *IET Wireless Sensor Systems* 9.5 (2019), pp. 257–264.

[132] Karl Hammerin and Ramona Streitenberger. *RTLS–the missing link to optimizing Logistics Management?* 2019.

[133] Sverre Holm. "Hybrid ultrasound-RFID indoor positioning: Combining the best of both worlds". In: *2009 IEEE International Conference on RFID*. IEEE. 2009, pp. 155–162.

[134] Sunkyu Woo, Seongsu Jeong, Esmond Mok, Linyuan Xia, Changsu Choi, Muwook Pyeon, and Joon Heo. "Application of WiFi-based indoor positioning system for labor tracking at construction sites: A case study in Guangzhou MTR". In: *Automation in Construction* 20.1 (2011), pp. 3–13.

[135] J Park, Yong K Cho, and Diego Martinez. "A BIM and UWB integrated mobile robot navigation system for indoor position tracking applications". In: *Journal of Construction Engineering and Project Management* 6.2 (2016), pp. 30–39.

[136] Peter Stephan, Ines Heck, P Krau, and Georg Frey. "Evaluation of Indoor Positioning Technologies under industrial application conditions in the SmartFactoryKL based on EN ISO 9283". In: *IFAC Proceedings Volumes* 42.4 (2009), pp. 870–875.

[137] Walter CSS Simões, Guido S Machado, André Sales, Mateus M de Lucena, Nasser Jazdi, and Vicente F de Lucena. "A Review of Technologies and Techniques for Indoor Navigation Systems for the Visually Impaired". In: *Sensors* 20.14 (2020), p. 3935.

[138] Tomas Lennvall, Stefan Svensson, and Fredrik Hekland. "A comparison of WirelessHART and ZigBee for industrial applications". In: *2008 IEEE International Workshop on Factory Communication Systems*. IEEE. 2008, pp. 85–88.

[139] Wang Longkang, Nie Baisheng, Zhang Ruming, Zhai Shengrui, and Li Hailong. "ZigBee-based positioning system for coal miners". In: *Procedia Engineering* 26 (2011), pp. 2406–2414.

[140] Marcin Uradzinski, Hang Guo, Xiaokang Liu, and Min Yu. "Advanced indoor positioning using zigbee wireless technology". In: *Wireless Personal Communications* 97.4 (2017), pp. 6509–6518.

[141] Jiaqiao Shi. "Laser guided four-wheel drive AGV trolley". In: *AIP Conference Proceedings*. Vol. 2073. AIP Publishing LLC. 2019, p. 020060.

[142] Gürsel Alici and Bijan Shirinzadeh. "A systematic technique to estimate positioning errors for robot accuracy improvement using laser interferometry based sensing". In: *Mechanism and Machine Theory* 40.8 (2005), pp. 879–906.

[143] Martin Kirch, Olaf Poenicke, and Klaus Richter. "RFID in logistics and production–Applications, research and visions for smart logistics zones". In: *Procedia Engineering* 178 (2017), pp. 526–533.

[144] Chuanying Zhai, Zhuo Zou, Qin Zhou, Jia Mao, Qiang Chen, Hannu Tenhunen, Lirong Zheng, and Lida Xu. "A 2.4-GHz ISM RF and UWB hybrid RFID real-time locating system for industrial enterprise Internet of Things". In: *Enterprise Information Systems* 11.6 (2017), pp. 909–926.

[145] Hyun-Soo Lee, Kwang-Pyo Lee, Moonseo Park, Yunju Baek, and SangHyun Lee. "RFID-based real-time locating system for construction safety management". In: *Journal of Computing in Civil Engineering* 26.3 (2012), pp. 366–377.

[146] Paul Martin, Bo-Jhang Ho, Nicholas Grupen, Samuel Muñoz, and Mani Srivastava. "An iBeacon primer for indoor localization: demo abstract". In: *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*. 2014, pp. 190–191.

[147] Hanhui Yue, Xiao Zheng, Juan Wang, Li Zhu, Chunyan Zeng, Cong Liu, and Meng Liu. "Research and Implementation of Indoor Positioning Algorithm for Personnel Based on Deep Learning". In: *International Conference on Emerging Internetworking, Data & Web Technologies*. Springer. 2018, pp. 782–791.

[148] Timothy Otim, Luis E Díez, Alfonso Bahillo, Peio Lopez-Iturri, and Francisco Falcone. "Effects of the body wearable sensor position on the UWB localization accuracy". In: *Electronics* 8.11 (2019), p. 1351.

[149] Asvin Gohil, Hardik Modi, and Shobhit K Patel. "5G technology of mobile communication: A survey". In: *2013 international conference on intelligent systems and signal processing (ISSP)*. IEEE. 2013, pp. 288–292.

[150] Klaus Witrisal, Stefan Hinteregger, Josef Kulmer, Erik Leitinger, and Paul Meissner. "High-accuracy positioning for indoor applications: RFID, UWB, 5G, and beyond". In: *2016 IEEE International Conference on RFID (RFID).* IEEE. 2016, pp. 1–7.

[151] Seppo Horsmanheimo, Sergio Lembo, Lotta Tuomimaki, Sami Huilla, Petri Honkamaa, Matti Laukkanen, and Paul Kemppi. "Indoor positioning platform to support 5G location based services". In: *2019 IEEE International Conference on Communications Workshops (ICC Workshops).* IEEE. 2019, pp. 1–6.

[152] Javier Rodas, Valentín Barral, and Carlos J Escudero. "Architecture for multi-technology real-time location systems". In: *Sensors* 13.2 (2013), pp. 2220–2253.

[153] Jin-Shyan Lee, Yu-Wei Su, and Chung-Chou Shen. "A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi". In: *IECON 2007-33rd Annual Conference of the IEEE Industrial Electronics Society.* Ieee. 2007, pp. 46–51.

[154] Jiun-Ren Lin, Timothy Talty, and Ozan K Tonguz. "On the potential of bluetooth low energy technology for vehicular applications". In: *IEEE Communications Magazine* 53.1 (2015), pp. 267–275.

[155] Xiaoning Jiang and Shaoju Chen. "Design of Wireless Point of Sale Based on ZigBee Technology". In: *Sensors & Transducers* 164.2 (2014), p. 120.

[156] Amna Abdullah, Asma Ismael, Aisha Rashid, Ali Abou-ElNour, and Mohammed Tarique. "Real time wireless health monitoring application using mobile devices". In: *International Journal of Computer Networks & Communications (IJCNC)* 7.3 (2015), pp. 13–30.

[157] Olaide O Kazeem, Olubiyi O Akintade, and Lawrence O Kehinde. "Comparative Study of Communication Interfaces for Sensors and Actuators in the Cloud of Internet of Things". In: *Int. J. Internet Things* 6.1 (2017), pp. 9–13.

[158] Yansong Cui and Jiayu Zhao. "Real-time location system and applied research report". In: *International Conference on Internet of vehicles.* Springer. 2015, pp. 49–57.

[159] AV Astafiev, AL Zhiznyakov, and DG Privezentsev. "Development of Indoor Positioning Algorithm Based on Bluetooth Low Energy beacons for Building RTLS-Systems". In: *2019 International Russian Automation Conference (RusAutoCon).* IEEE. 2019, pp. 1–5.

[160] Piotr Nowotarski, Jerzy Pasławski, Maciej Skrzypczak, and Radosław Krygier. "RTLS systems as a Lean Management tool for productivity improvement". In: *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction.* Vol. 34. IAARC Publications. 2017.

[161] Adriano Moreira, Maria João Nicolau, Filipe Meneses, and António Costa. "Wi-Fi fingerprinting in the real world-RTLS@ UM at the EvAAL competition". In: *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN).* IEEE. 2015, pp. 1–10.

[162] Fazeelat Mazhar, Muhammad Gufran Khan, and Benny Sällberg. “Precise indoor positioning using UWB: a review of methods, algorithms and implementations”. In: *Wireless Personal Communications* 97.3 (2017), pp. 4467–4491.

[163] Bin Ding, Li Chen, Dianlong Chen, and Haitao Yuan. “Application of RTLS in warehouse management based on RFID and Wi-Fi”. In: *2008 4th International conference on wireless communications, networking and mobile computing*. IEEE. 2008, pp. 1–5.

[164] T. Ruppert and J. Abonyi. “Industrial Internet of Things based cycle time control of assembly lines”. In: *2018 IEEE International Conference on Future IoT Technologies (Future IoT)*. 2018, pp. 1–4.

[165] Dávid Gyulai, András Pfeiffer, and Júlia Bergmann. “Analysis of asset location data to support decisions in production management and control”. In: *Procedia CIRP* 88 (2020), pp. 197–202.

[166] Behnam Badihi, Jianyu Zhao, Siyan Zhuang, Olli Seppänen, and Riku Jäntti. “Intelligent Construction Site: On Low Cost Automated Indoor Localization Using Bluetooth Low Energy Beacons”. In: *2019 IEEE Conference on Wireless Sensors (ICWiSe)*. IEEE. 2019, pp. 29–35.

[167] Meimanat Soleimanifar and Ming Lu. “Streamlining an indoor positioning architecture based on field testing in pipe spool fabrication shop”. In: *Proceedings of the Winter Simulation Conference 2014*. IEEE. 2014, pp. 3260–3271.

[168] ZXa Guo, EWTa Ngai, Can Yang, and Xuedong Liang. “An RFID-based intelligent decision support system architecture for production monitoring and scheduling in a distributed manufacturing environment”. In: *International journal of production economics* 159 (2015), pp. 16–28.

[169] Jennie J Gallimore, Laurie Quill, Ron Cagle, Jessica Gruenke, Chris Hosman, Elizabeth Matthews, Paul Faas, Jason Seyba, and Ian Young. *User Feedback on RFID and Integrated Flightline Data for Maintenance Decisions*. Tech. rep. DAYTON UNIV OH RESEARCH INST, 2006.

[170] Tamas Ruppert and Janos Abonyi. “Software Sensor for Activity-Time Monitoring and Fault Detection in Production Lines”. In: *Sensors* 18.7 (2018), pp. 1–18.

[171] Mirco Peron, Giuseppe Fragapane, Fabio Sgarbossa, and Michael Kay. “Digital Facility Layout Planning”. In: *Sustainability* 12.8 (2020), p. 3349.

[172] Yujie Zang and Lihua Wu. “Application of RFID and RTLS technology in supply chain enterprise”. In: *2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*. IEEE. 2010, pp. 1–4.

[173] CKM Lee, Yaqiong Lv, KKH Ng, William Ho, and KL Choy. “Design and application of Internet of things-based warehouse management system for smart logistics”. In: *International Journal of Production Research* 56.8 (2018), pp. 2753–2768.

[174] Xiangguo Ma and Tongjuan Liu. “The application of Wi-Fi RTLS in automatic warehouse management system”. In: *2011 IEEE International conference on automation and logistics (ICAL)*. IEEE. 2011, pp. 64–69.

[175] Zuin Silvia, Calzavara Martina, Sgarbossa Fabio, and Persona Alessandro. “Ultra Wide Band Indoor Positioning System: analysis and testing of an IPS technology”. In: *IFAC-PapersOnLine* 51.11 (2018), pp. 1488–1492.

[176] Csaba Hegedűs, Attila Frankó, and Pál Varga. “Asset and Production Tracking through Value Chains for Industry 4.0 using the Arrowhead Framework”. In: *2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*. IEEE. 2019, pp. 655–660.

[177] Abiola A Akanmu, Chimay J Anumba, and John I Messner. “An RTLS-based approach to cyber-physical systems integration in design and construction”. In: *International Journal of Distributed Sensor Networks* 8.12 (2012), p. 596845.

[178] A Akanmu, S Hammad Rasheed, and I Abdel Qader. “Spatial mapping approach to component tracking using RTLS system”. In: *AEI 2013: Building Solutions for Architectural Engineering*. Architectural Engineering Conference 2013, 2013, pp. 364–376.

[179] Attila Frankó, Gergely Vida, and Pal Varga. “Reliable Identification Schemes for Asset and Production Tracking in Industry 4.0”. In: *Sensors* 20.13 (2020), p. 3709.

[180] Gheorghe Sima, Ramona Lile, Glavan Dan, and Elena Muncut. “Management, traceability and control of industrial processes”. In: *Soft Computing Applications*. Springer, 2016, pp. 971–980.

[181] Shaoping Lu, Chen Xu, Ray Y Zhong, and Lihui Wang. “A RFID-enabled positioning system in automated guided vehicle for smart factories”. In: *Journal of Manufacturing Systems* 44 (2017), pp. 179–190.

[182] Thomas Kelepouris and Duncan McFarlane. “Determining the value of asset location information systems in a manufacturing environment”. In: *International Journal of Production Economics* 126.2 (2010), pp. 324–334.

[183] Hoon Choi, Yunju Baek, and Ben Lee. “Design and implementation of practical asset tracking system in container terminals”. In: *International Journal of Precision Engineering and Manufacturing* 13.11 (2012), pp. 1955–1964.

[184] Srideep Ghosh, D Anurag, and Somprakash Bandyopadhyay. “Use of wireless mesh network to improve mobile asset utilization in manufacturing industries”. In: *Procedia Computer Science* 5 (2011), pp. 66–73.

[185] Jiajun Jiang, Yu Guo, and Wenhe Liao. “Research on AGV guided by real-time locating system (RTLS) for material distribution”. In: *International Journal of Control & Automation* 8.7 (2015), pp. 213–226.

[186] Valentín Barral, Pedro Suárez-Casal, Carlos J Escudero, and José A García-Naya. “Multi-sensor accurate forklift location and tracking simulation in industrial indoor environments”. In: *Electronics* 8.10 (2019), p. 1152.

[187] Huan-Bang Li, Ryu Miura, Hisashi Nishikawa, Toshinori Kagawa, and Fumihide Kojima. "Tracking of warehouse forklifts using an indoor positioning system based on IR-UWB". In: *International Conference on indoor positioning and navigation (IPIN 2016)*. 2016.

[188] Juraj Slovák, Pavol Vašek, Matej Šimovec, Markus Melicher, and Dana Šišmišová. "RTLS tracking of material flow in order to reveal weak spots in production process". In: *2019 22nd International Conference on Process Control (PC19)*. IEEE. 2019, pp. 234–238.

[189] Cheng Zhang and Amin Hammad. "Multiagent approach for real-time collision avoidance and path replanning for cranes". In: *Journal of Computing in Civil Engineering* 26.6 (2012), pp. 782–794.

[190] Santiago Barro-Torres, Tiago M Fernández-Caramés, Héctor J Pérez-Iglesias, and Carlos J Escudero. "Real-time personal protective equipment monitoring system". In: *Computer Communications* 36.1 (2012), pp. 42–50.

[191] Alp Ulku. "The Next Generation In Personnel\/People Tracking: Active RFID Technology Has Allowed for Enhanced Security and Safety". In: *IEEE Consumer Electronics Magazine* 6.4 (2017), pp. 122–124.

[192] Enrico Köppe, Matthias Bartholmai, Achim Liers, and Jochen Schiller. "Radio-based multi-sensor system for person tracking and indoor positioning". In: *2012 9th Workshop on Positioning, Navigation and Communication*. IEEE. 2012, pp. 180–186.

[193] Thomas R Hellmich, Casey M Clements, Nibras El-Sherif, Kalyan S Pasupathy, David M Nestler, Andy Boggust, Vickie K Ernste, Gomathi Marisamy, Kyle R Koenig, and M Susan Hallbeck. "Contact tracing with a real-time location system: A case study of increasing relative effectiveness in an emergency department". In: *American Journal of Infection Control* 45.12 (2017), pp. 1308–1311.

[194] Ihsan Arkan and Hendrik Van Landeghem. "Evaluating the performance of a discrete manufacturing process using RFID: a case study". In: *Robotics and Computer-Integrated Manufacturing* 29.6 (2013), pp. 502–512.

[195] Bartlomiej Gladysz, Krzysztof Santarek, and Cezary Lysiak. "Dynamic spaghetti diagrams. A case study of pilot RTLS implementation". In: *International Conference on Intelligent Systems in Production Engineering and Maintenance*. Springer. 2017, pp. 238–248.

[196] Adewole Bamidele Awolusi, Oluwole Charles Akinyokun, and Gabriel Babatunde Iwasokun. "RFID and RTLS-Based Human Resource Monitoring System". In: *Journal of Advances in Mathematics and Computer Science* (2016), pp. 1–14.

[197] Hanley J Ho, Zoe Xiaozhu Zhang, Zhilian Huang, Aung Hein Aung, Wei-Yen Lim, and Angela Chow. "Use of a real-time locating system for contact tracing of health care workers during the COVID-19 pandemic at an infectious disease center in Singapore: validation study". In: *Journal of Medical Internet Research* 22.5 (2020), e19437.

[198] Tamas Ruppert and Janos Abonyi. “Integration of real-time locating systems into digital twins”. In: *Journal of Industrial Information Integration* 20 (2020), p. 100174.

[199] Kadir Alpaslan Demir, Gözde Döven, and Bülent Sezen. “Industry 5.0 and human-robot co-working”. In: *Procedia Computer Science* 158 (2019), pp. 688–695.

[200] Saeid Nahavandi. “Industry 5.0—A human-centric solution”. In: *Sustainability* 11.16 (2019), p. 4371.

[201] Margherita Peruzzini, Fabio Grandi, and Marcello Pellicciari. “Exploring the potential of Operator 4.0 interface and monitoring”. In: *Computers & Industrial Engineering* 139 (2020), p. 105600.

[202] Tamás Ruppert, Szilárd Jaskó, Tibor Holczinger, and János Abonyi. “Enabling technologies for operator 4.0: a survey”. In: *Applied sciences* 8.9 (2018), p. 1650.

[203] Andreas Löcklin, Tamás Ruppert, László Jakab, Robert Libert, Nasser Jazdi, and Michael Weyrich. “Trajectory Prediction of Humans in Factories and Warehouses with Real-Time Locating Systems”. In: *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. Vol. 1. IEEE. 2020, pp. 1317–1320.

[204] Mahbubeh Sattarian, Javad Rezazadeh, Reza Farahbakhsh, and Alireza Bagheri. “Indoor navigation systems based on data mining techniques in internet of things: A survey”. In: *Wireless Networks* 25.3 (2019), pp. 1385–1402.

[205] Ahmet Yazıcı, Sinem Bozkurt Keser, and Serkan Gunal. “Integration of classification algorithms for indoor positioning system”. In: *2017 International Conference on Computer Science and Engineering (UBMK)*. IEEE. 2017, pp. 267–270.

[206] Kamran Zia, Hifsa Iram, Muhammad Aziz-ul-Haq, and Aasim Zia. “Comparative study of classification techniques for indoor localization of mobile devices”. In: *2018 28th International Telecommunication Networks and Applications Conference (ITNAC)*. IEEE. 2018, pp. 1–5.

[207] He Xu, Ye Ding, Peng Li, Ruchuan Wang, and Yizhu Li. “An RFID indoor positioning algorithm based on Bayesian probability and K-nearest neighbor”. In: *Sensors* 17.8 (2017), p. 1806.

[208] Abebe Belay Adege, Hsin-Piao Lin, Getaneh Berie Tarekegn, Yirga Yayeh Munaye, and Lei Yen. “An indoor and outdoor positioning using a hybrid of support vector machine and deep neural network algorithms”. In: *Journal of sensors* 2018 (2018).

[209] Lummanee Chanama and Olarn Wongwirat. “A comparison of decision tree based techniques for indoor positioning system”. In: *2018 International Conference on Information Networking (ICOIN)*. IEEE. 2018, pp. 732–737.

[210] Zhefu Wu, Qiang Xu, Jianan Li, Chenbo Fu, Qi Xuan, and Yun Xiang. “Passive indoor localization based on csi and naive bayes classification”. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 48.9 (2017), pp. 1566–1577.

[211] Abdulraqeb Alhammadi, Saddam Alraih, Fazirulhisyam Hashim, and Mohd Fadlee A Rasid. “Robust 3D Indoor Positioning System Based on Radio Map Using Bayesian Network”. In: *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE. 2019, pp. 107–110.

[212] Ahmet Yazıcı, Sinem Bozkurt Keser, Serkan Günal, and Uğur Yayan. “A Multi-Criteria Decision Strategy to Select a Machine Learning Method for Indoor Positioning System”. In: *International Journal on Artificial Intelligence Tools* 27.05 (2018), p. 1850018.

[213] Zong-kai LIAN, Fei YUAN, and Wei QI. “Improved K-means Clustering BP Neural Network UWB Indoor Location Method”. In: *Modern Computer* 2017.21 (2017), p. 3.

[214] Yun Mo, Zhongzhao Zhang, Weixiao Meng, Lin Ma, and Yao Wang. “A spatial division clustering method and low dimensional feature extraction technique based indoor positioning system”. In: *Sensors* 14.1 (2014), pp. 1850–1876.

[215] Santosh Subedi, Hui-Seon Gang, Nak Yong Ko, Suk-Seung Hwang, and Jae-Young Pyun. “Improving indoor fingerprinting positioning with affinity propagation clustering and weighted centroid fingerprint”. In: *IEEE Access* 7 (2019), pp. 31738–31750.

[216] Matti Pastell and Lilli Frondelius. “A hidden Markov model to estimate the time dairy cows spend in feeder based on indoor positioning data”. In: *Computers and Electronics in Agriculture* 152 (2018), pp. 182–185.

[217] Ashraf Sayed Abdou, Mostafa Abdel Aziem, and Ashraf Aboshosha. “An efficient indoor localization system based on Affinity Propagation and Support Vector Regression”. In: *2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC)*. IEEE. 2016, pp. 1–7.

[218] Eric Hsueh-Chan Lu, Kuei-Hua Chang, and Jing-Mei Ciou. “Data Preprocessing Based on Convolutional Neural Network for Improving Precision of Indoor Positioning”. In: *Asian Conference on Intelligent Information and Database Systems*. Springer. 2020, pp. 545–552.

[219] Drew Gislason. *Zigbee wireless networking*. Newnes, 2008.

[220] Min-Hsiung Hung, Shih-Sung Lin, Jui-Yu Cheng, and Wu-Lung Chien. “A ZigBee indoor positioning scheme using signal-index-pair data preprocess method to enhance precision”. In: *2010 IEEE International Conference on Robotics and Automation*. IEEE. 2010, pp. 548–553.

[221] Changzhi Wang, Fei Wu, Zhicai Shi, and Dongsong Zhang. “Indoor positioning technique by combining RFID and particle swarm optimization-based back propagation neural network”. In: *Optik* 127.17 (2016), pp. 6839–6849.

[222] Xiaolong Yang, Yanmeng Wang, Mu Zhou, and Yiyao Liu. “Pedestrian motion learning based indoor WLAN localization via spatial clustering”. In: *Wireless Communications and Mobile Computing* 2018 (2018).

[223] Michael J Kuhn, Mohamed R Mahfouz, Jonathan Turnmire, Yazhou Wang, and Aly E Fathy. “A multi-tag access scheme for indoor UWB localization systems used in medical environments”. In: *2011 IEEE Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems*. IEEE. 2011, pp. 75–78.

[224] Zheng Gao, Yongtao Ma, Kaihua Liu, Xinlong Miao, and Yang Zhao. “An indoor multi-tag cooperative localization algorithm based on NMDS for RFID”. In: *IEEE Sensors Journal* 17.7 (2017), pp. 2120–2128.

[225] Faisal Jamil, Naeem Iqbal, Shabir Ahmad, and Do-Hyeun Kim. “Toward Accurate Position Estimation Using Learning to Prediction Algorithm in Indoor Navigation”. In: *Sensors* 20.16 (2020), p. 4410.

[226] He Xu, Manxing Wu, Peng Li, Feng Zhu, and Ruchuan Wang. “An RFID indoor positioning algorithm based on support vector regression”. In: *Sensors* 18.5 (2018), p. 1504.

[227] Thomas Moder, Petra Hafner, Karin Wisiol, and Manfred Wieser. “3d indoor positioning with pedestrian dead reckoning and activity recognition based on bayes filtering”. In: *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2014, pp. 717–720.

[228] Yong Gao, Dan Jiang, and Yan Xu. “Optimize taxi driving strategies based on reinforcement learning”. In: *International Journal of Geographical Information Science* 32.8 (2018), pp. 1677–1696.

[229] James Edmund Reeb and Scott A. Leavengood. “Transportation problem: a special case for linear programming problems”. In: *Performance Excellence in the Wood Products Industry* (2002).

List of Figures

List of Tables