Agent Oriented Smart Factory (AOSF): a MAS based framework for SMEs under Industry 4.0

A thesis submitted by

Fareed Ud Din, BCompSc (Hons), MCompSc (Research)

to fulfill the requirements for the degree of Doctorate in Philosophy (PhD) in Computer Science

Supervisor: Dr Joe Ryan

Co-Supervisor: Dr David Paul

August, 2019

This research was supported by an Australian Government Research Training Program (RTP) Scholarship

School of Electrical Engineering and Computing,

Faculty of Engineering and Built Environment,

The University of Newcastle, Australia

Statement of Originality

I hereby certify that the work embodied in the thesis is my own work, conducted under normal supervision. The thesis contains no material which has been accepted, or is being examined, for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made. I give consent to the final version of my thesis being made available worldwide when deposited in the University's Digital Repository, subject to the provisions of the Copyright Act 1968 and any approved embargo.

Fareed Ud Din

Acknowledgment of Authorship

I hereby certify that the work embodied in this thesis contains published papers/scholarly work of which I am a joint author. I have included as part of the thesis a written declaration endorsed in writing by my supervisors and co-author, attesting to my contribution to this thesis and joint publications/scholarly work.

Fareed Ud Din

By signing below I confirm that Fareed Ud Din was the principal contributor in conceptualisation, writing and revision of this complete thesis.

Joe Ryan

By signing below I confirm that Fareed Ud Din was the principal contributor in conceptualisation, preparation, design, analysis, writing and revision of the papers/publications entitled in the publications list below.

David Paul

List of Publications included as part of the thesis

- Ud Din F., Henskens F., Paul D., Wallis M. (2018) "Agent-Oriented Smart Factory (AOSF): An MAS Based Framework for SMEs Under Industry 4.0". In: Jezic G., Chen-Burger YH., Howlett R., Jain L., Vlacic L., Šperka R. (eds) Agents and Multi-Agent Systems: Technologies and Applications 2018. KES-AMSTA-18 2018. *Smart Innovation, Systems and Technologies*, vol 96. Springer, Cham
- Ud Din F., Henskens F., Paul D., Wallis M., (2018) "Formalisation of Problem and Domain Definition for Agent-Oriented Smart Factory (AOSF)", in *IEEE Region 10 Symposium (TenSymp), IEEE, 2019, pp. 265-270*
- Ud Din F., Henskens F., Paul D., Wallis M. and Hashmi M., (2019) "AOSR-WMS planner associated with AOSF framework for SMEs, under Industry 4.0", In review with *Cybernetics and Systems*

List of Achievements during candidature

- **Best Research Paper Award** at International Conference on Agents and Multi-Agent Systems: Technologies and Applications 2018, Gold Coast, Australia
- **Best Technical Research Poster Award**, at Poster Competition 2018, Faculty of Engineering and Built Environment, The University of Newcastle, Australia.
- **Best Academic Achiever of the Year 2019**, from the UNIS Society, The University of Newcastle, Australia.

Acknowledgments

This thesis would not have been completed without the support and encouragement of many people. Prominently among these people are my current and previous PhD supervisors; Prof. Frans Henskens, Dr Joe Ryan, Dr David Paul and Dr Mark Wallis. Special thanks to Dr David who has continued providing me with guidance throughout my candidature from beginning till end. I am much obliged that you have provided an extraordinary amount of time and effort to ensuring I produce a comprehensive body of work. Overall, I am indebted to all of you, for your advice, encouragement and positive attitude, which has proved invaluable. Thank you, for never letting me get distracted and keeping me focused and motivated all the time, in parallel to providing me with teaching opportunities to remain fresh with my skills.

Thank you to the School of Electrical Engineering and Computing at the University of Newcastle, for providing me with the generous scholarships; UNIPRS and UNRSC to cover my tuition fees and stipend, without it I was unable to bear the expense of studies. It's been a wonderful time, full of learning. Thanks for providing me the opportunity to teach at undergraduate level in Computer Science and Software Engineering disciplines. I look forward to support the faculty back in future, if I get a chance.

Finally, and most importantly, I offer my undying gratitude to my family, especially my parents, my brothers, my wife, and kids. My father would have been so happy to see the first PhD of the family; miss you Dad! And Mum, I'm not sure how it would have been possible to come through a long way without your prayers and kind wishes. And a massive thanks to you Mehwish; you've been a wonderful life-partner to support me throughout this process of PhD. I'm lucky to have you in my life; thank you for understanding and cooperating with me overall.

Contents

1 Introduction **1**

1.1 Industry 4.0 . . . 2
1.2 Motivation . . . 3
1.3 Research Objective . . . 6
1.4 Research Questions . . . 8
1.5 Research Methodology . . . 8
1.6 Thesis Structure . . . 10
1.7 Contributions . . . 11
1.8 Achievements and Recognition . . . 12

2 Theoretical Background **13**

2.1 Industry 4.0 and SMEs . . . 13
2.1.1 Cyber Physical Systems . . . 16
2.1.2 Supply Chain Architecture . . . 17
2.2 Warehouse Management Systems . . . 18
2.2.1 Standard Warehousing Strategies . . . 20
2.2.2 Robo-Warehousing . . . 22
2.3 Summary . . . 24

3 Overview of Agents and Related Technologies **25**

3.1 Multi-Agent Development Environments . . . 27
3.1.1 Agent-Based Development Systems . . . 28
3.1.2 Agent-Oriented Development Systems . . . 28
3.2 Agent Development Methodologies . . . 30
3.3 Interpreting Output in JADE . . . 32
3.4 Summary . . . 34

4 Agent-Oriented Smart Factory (AOSF) Framework **35**

4.1 Introduction . . . 35
4.2 Publication . . . 36
4.2.1 Erratum . . . 48
4.3 Further Details on Interaction within AOSF-Agents . . . 48

4.4 Enterprise Integration and AOSF 52
4.5 Summary 55

5 Problem and Domain Definition for AOSF 56
5.1 Introduction 56
5.2 Publication 58
5.2.1 Erratum 65

6 Agent-Oriented Storage and Retrieval-WMS Strategy 66
6.1 Introduction 66
6.2 Publication 67

7 How the AOSF-AOSR Strategy addresses the need of Industry 87
7.1 The Need of AOSF framework 87
7.2 The Need of AOSR Strategy 90

8 Validation 93
8.1 Test Scenario and AOSF 93
8.1.1 Dataset and Test Cases 97
8.1.2 Results and Discussion 101
8.1.3 Aggregated Results with Random Values 104
8.2 Test Scenarios and AOSR 106
8.2.1 Algorithmic Heuristics of AOSR 107
8.2.2 Results and Discussion 111

9 Conclusion 125
9.1 Revisiting Research Questions 127
9.1.1 Research Question 1 127
9.1.2 Research Question 2 128
9.1.3 Research Question 3 129
9.2 Future Directions 130
9.2.1 Cloud Architecture for Multiple Sites 131
9.2.2 Big Data Analytics 131
9.2.3 Addition to FIPA-ACL 132
9.2.4 Corporate Cutting Edge Features 132
9.2.5 Industry Interests 133
Appendices - A - Overview of Algorithms 134

Appendices - B - Dataset with Random Variation 139

Bibliography **143**

Glossary

ABM	Agent Based Modelling
ACL	Agent Communication Language
ADN	Advance Delivery Notices
AOSF	Agent Oriented Smart Factory
AOSR	Agent Oriented Storage and Retrieval
AS/RS	Automated Storage and Retrieval System
ASNs	Advance Shipment Notices
AUML	Agent Unified Modelling Language
BDI	Belief Desire Intention
BPR	Business Process Re-engineering
BRM	Business Relationship Management
BNF	Backus-Naur-Form
CNP	Contract Net Protocol
COI	Cube Per Order Index
CPS	Cyber Physical System
CPPS	Cyber Physical Production System
CR	Critical Ratio
CRM	Customer Relationship Management
DCS	Distributed Control Systems
FCFS	First Come First Serve
FIFO	First In First Out
FIPA	Foundation of Intelligent and Physical Agents
EA	Expedition Area
ECU	Enterprise Central Unit
EDD	Earliest Due Date
EI	Enterprise Integration
EMBBO	Ensamble Multi-Objective Biography Based Optimisation
ERP	Enterprise Resource Planning Systems
IDE	Integrated Development Environments
IIC	Industrial Internet Consortium

IoT	Internet of Things
IS	Information Systems
IWN	Intra-Enterprise Wireless Network
JADE	Java Agent Development Environment
JIT	Just In Time
KQML	Knowledge Query and Manipulation Language
LIS	Logistical Information System
MA	Mediator Agent
MA-HTN	Multi-Agent Hierarchical Task Networking
MAP	Multi-Agent Planning
MAS	Multi-Agent Systems
MIS	Management Information System
OO	Object-Orientation
OLAP	Online Analytical Processing
PA	Planner Agent
PLC	Programmable Logic Controllers
RA	Receiving Area
RMI	Remote Method Invocation
S/R Machines	Storage or Retrieval Machines
SC	Supply Chain
SCM	Supply Chain Management
SDA	Smart Device Agents
SPT	Shortest Processing Time
SKU	Stock Keeping Unit
SME	Small to Medium Size Enterprises
SOA	Service Oriented Architecture
TA	Timestamped Actions
UA	User Agents
UML	Unified Modelling Language
WMS	Warehouse Management System
xAOSF	Extended Agent Oriented Smart Factory

Abstract

Recently, the inception of a fourth industrial revolution, termed Industry 4.0, gave a boost to the concept of the smart factory, which offers the advanced features of enterprise integration, automation, seamless information exchange, intelligent self-organisation of components and decentralised decision making. In order to accomplish these promises, a mature amalgamation of allied technologies e.g. Internet of Things (IoT), Cloud Computing, Big Data and Multi-Agent Systems (MAS) is incumbent. Recent research explains that the idea of Industry 4.0 focuses mainly on large enterprise but, for its compatibility with Small to Medium Size Enterprises (SMEs), there is still much research to be done.

This dissertation focuses on providing a comprehensive SC architecture for SMEs under the umbrella of Industry 4.0 to resolve the issue of compatibility, by presenting the MAS based Agent Oriented Smart Factory (AOSF) framework. This framework provides a general architecture for the whole value chain, incorporating concerns from both ends of a firm: Supply Chain Management (SCM) and Customer Relationship Management (CRM). In order to provide a complete solution, this thesis also includes the associated framework of Agent Oriented Storage and Retrieval (AOSR) system to alleviate the persisting problems of SMEs in warehouse management. The classification and categorisation of constituent agents of this two-fold system, with their negotiation and communication strategies, are also discussed. Problem and Domain definitions for AOSF are extracted using a multi-agent extension of Hierarchical Task Networking (MA-HTN). Heuristics and experimental results for the implementation and validation of this system are also presented in comparison with existing standard strategies. The results reflect improvements in overall efficiency within SME-oriented warehouses. Some of the possible future work recommendations, scalability of this system and industry interest for this proposed strategy are also discussed.

Chapter 1

Introduction

A significant proportion of the world's economy is based on the manufacturing industry [124]. These industrial setups have been evolving ever since their inception. This continuous growth is supported by incorporating process integration, mechanisation of operations and customised procedural manufacturing [51]. The industrial world is now moving towards virtualisation and seamless operations with the help of artificial intelligence [199]. Extensive research and development has provided the manufacturing industry with high-tech solutions to speed-up the process of production and delivery of end-products to customers by utilising concepts of Distributed Artificial Intelligence [110], Internet of Things (IoT) [115], Big Data [184], Multi-Agent Systems (MAS) [113], Cloud Computing [82] and Industry 4.0 [67]. Fourth industrial revolution (Industry 4.0) is also one of the recent state of the art frameworks, which is providing a constructive disruption and transformation of traditional manufacturing industrial setups with the help of smart technologies [74].

Small to Medium Size Enterprises (SMEs) are broadly defined as firms which maintain lower annual revenue/assets than a defined threshold [172]. They operate under tight financial limits but represent about 90% of businesses and more than 50% of employment worldwide [12]. This thesis explicitly focuses of manufacturing type of SMEs. The processes in SMEs are usually ad-hoc and need to be flexible to cope with frequently changing external factors, such as varying demand, lack of resources, unstable customers and fluctuating prices in the market [132]. Therefore, Supply Chains (SC) in SMEs are usually very flexible and kept adaptable to cater to their changing needs [142]. In an SME-oriented environment, business processes are usually not fixed, which is one of the reasons that process automation is more complicated than compared to large enterprises [127].

The initiative of Industry 4.0 recommends an advanced infrastructural shift for incorporating intelligent machines within the manufacturing Supply Chain (SC), which can interact with each other, having nano or micro chips installed in them [173]. In order to build such a structure, high performance computing devices are required, which ultimately increase the infrastructural and operational cost. Although large setups can afford such solutions, Small to Medium Size Enterprises (SMEs), which are mostly centrally controlled and are not compatible with such an advanced system [5], may lag behind [133]. For large enterprises, Industry 4.0 standards require business process re-engineering, thorough digitisation, and inclusive inter-enterprise connectivity, which creates a mismatch for Industry 4.0 standards to be directly applied to SMEs [132]. Hence, in order to bring SMEs many of the benefits of Industry 4.0, this thesis presents a novel approach for a moderate-level semi-autonomous system for SMEs to apply a comprehensive SC framework under the umbrella of Industry 4.0, with implementation of a hybrid-logic based Warehouse Management System (WMS).

This chapter presents an overview of the concept of Industry 4.0 and highlights the gaps identified in literature for its implementation, particularly for SMEs, in Section 1.1. The rest of the chapter presents motivation, objectives, research questions, methodology and thesis structure as well as detailing the contributions of this work.

1.1 Industry 4.0

The industrial world has passed through a drastic change ever since its beginning when water-steam mechanical systems were incorporated in the 18th century. In literature, it is named the 'first industrial revolution' [37], which changed the nature of the mechanical industry throughout the globe. In the mid 20th century, after the introduction of mass production and the division of labour, came the 'second industrial revolution' [129], with a high-impact-change overall. By the 1970s Programmable Logic Controllers (PLCs) stepped into the market, giving a boost for employing auto-mechanical infrastructure, forcing the industry to adapt to the 'third industrial revolution' [51]. This rapid development led to the fourth industrial revolution, termed Industry 4.0 (also known as Industrie 4.0) [74]. The concepts of smart factory and the fourth industrial revolution were initiated by The German Federal Ministry of Education and Research in 2011 and they officially announced the term 'Industry 4.0' in 2013 [199]. Literature defines Industry

4.0 as an information revolution, fostering the concept of smart manufacturing, which is based on integrated, adaptive and optimised service-oriented inter-operable frameworks to provide high customisability and decentralised decision making [111]. Over the last few years, the development in state of the art technologies, such as IoT, MAS and Big Data technologies has fueled this concept even more, both in practice and academic research [113].

Warehouses, though they hold crucial importance for any small to medium size production organisation [1], are not getting the expected benefits of high-tech standards under the umbrella of Industry 4.0, as they are still facing issues in warehouse management [23]. A typical warehouse is a facility that provides support to the overall SC of a firm by offering storage and retrieval services as required [60]. Normally warehouses have capacity to store different categories of products in respective regions within the warehouse. Warehouses are usually subdivided into different areas to receive, store and ship products [36]. Defining these areas within the warehouse, and balancing the load in between, is itself a problem in the warehouse domain [23]. A common management information system (MIS), utilised to support day-to-day operations in a warehouse, is a Warehouse Management System (WMS) [179] (discussed in Chapter 2 in detail). Several Enterprise Resource Planning systems (ERPs) provide a separate software module for a WMS, but because such systems are based on predefined requirements and offer minimal levels of flexibility and customisation [161], the industry is still facing several warehousing problems, e.g. an increasing number of products in receiving and expedition areas, which cause delays in overall warehouse activities [9] or unmanaged storage/retrieval areas, which reduce the overall performance within warehouses [110]. Hence, a generic solution is needed under the umbrella of Industry 4.0 that can provide support to the overall supply chain (SC) for Small to Medium size Enterprises (SMEs) and attempt to reduce the issues of warehousing.

1.2 Motivation

Since the inception of the concept of Industry 4.0, this standard has been adopted as a strategic plan by the world's leading industrial economies e.g. Germany's *High-Tech Strategy 2020 Plan* [74], USA's *Advance Manufacturing Plan* [192] *Industrial Internet*

Consortium (IIC) [73], South Korea's *Manufacturing Innovation 3.0* [94], China's *Manufacturing 2025 plan* [107] and Taiwan's *Productivity 4.0 plan* [107]. The research evolution in the domain of Industry 4.0 has produced substantial results [53, 106, 111] in different dimensions such as implementing smart factories for large autonomous enterprises [183], flexibility to support supply chain scheduling [75] and compatibility with state of the art technologies such as IoT protocols [68, 184], Multi-Agent Systems (MAS) [2] and Big Data [184]. The successful implementation of Industry 4.0 in the textile [4] and thermal industries [83] has validated the claimed benefits, but there are still many open research questions about the exact implementation of Industry 4.0 such as business relationship management (BRM), supply chain management (SCM) and service-oriented architectures (SOA).

Recent research suggests that Industry 4.0 mainly focuses on large enterprises [8], and marginally on Small to Medium Size Enterprises (SMEs) [164]. High dependence on comprehensive Industrial Internet of Things (IIoT) and Business Process Re-engineering (BPR), is comparatively common in large enterprises as they tend to operate more on long-term goals and usually have established business processes, which can be reviewed and altered, if needed, in top-to-bottom fashion [127]. On the contrary, in SMEs bringing a new operation to support new technology can cause substantial disruption budget-wise [109]. However, the fact that SMEs play a crucial role in building the economy in any part of the world, can't be denied, and hence should not be overlooked. In Australia there are around 2.2 million SMEs which account for more than 57% of the annual GDP [134], in Germany they are 3.4 million [55], in UK there are 5.7 million [154] and in US they are around 30 million SMEs registered [139]. Thus, if the compatibility of the high-tech standards of Industry 4.0 with such a large proportion of the industry is not identified properly, there may be a chance that SMEs may lag behind or may not get the expected benefits to keep up with the current competitive market [133]. Similarly, process automation is another area where SMEs lag behind in comparison to large setups. For example, in warehousing there are some commercial solutions that offer complete warehouse automation (e.g. GrayOrange [186] and Unleashed [187]) but such solutions are not always the best choice for SMEs, not least because of the question of affordability, because of infrastructure and equipment cost [109]. For SMEs, solutions providing improved efficiency, supply chain flexibility, integrations and quality with low infrastructural cost have always been a preference [127]. Hence there is a need for a solution that offers a comprehensive framework to provide integration along the whole supply chain to overcome the aforementioned critical issues in SMEs (e.g. process compatibility, low-cost automation, improved efficiency, supply chain flexibility and integration), including major warehouse problems such as lack of real-time stock information, unmanaged inventory and human-operated functional delays [59].

Regardless of the size of the organisation, SC is a fundamental element that provides an organisation with process flow. For SMEs, the importance of SC networks becomes more crucial as they rely solely on the tightly-integrated sub-systems and components to maintain business processes. Within an SC network, a warehouse holds the central position to keep the flow of supply running from both ends: the front-end customer and the back-end supplier. Modern warehouses are equipped with storage and retrieval (S/R) machines to pick up products from an input/output (I/O) location and store them at their specific locations, and then to retrieve out-going products from other storage locations and deliver them to the I/O locations [9].

Warehouse scheduling is a typical NP-hard problem, and is one of the most challenging types of combinatorial optimisation problems [54]. Flowshop Algorithm [28], for controlling low-end operations in warehouses, and EMBBO Algorithm [114], for automation of warehouses, are important contributions in this regard, though SME-based warehouses are still facing warehouse management issues. One such issue in a typical warehouse is referred to as manual re-slotting, where the warehouse activities are performed based on manual or human-operated activities, such as stock-count and changing the location of products based on self judgment, which leads to issues of inaccurate stock count and misplaced or wandering items [23]. Unmanaged storage capacity and high turbulence in receiving/expedition areas, is also an issue in warehouses [110, 156]. Such issues arise if designated areas, such as expedition and receiving areas (discussed in Chapter 2 in detail), which are only meant to hold the products temporarily, hold the products too long; it can create overloading and mismanagement in a warehouse [156].

For SMEs, implementing a low cost but effective solution has always been a preference [109]. Thus, a general Industry 4.0 based SC framework for SMEs may help to bridge this gap to provide a solution for persisting issues. Agent technologies offer high level support for providing embedded intelligence with rationality. So the focus of this research, as depicted in Figure 1.1, is based on attempting to provide a MAS-based smart factory framework for SMEs under Industry 4.0 that may incorporate the foundations of Cyber-Physical System (CPS) [102] (detailed in Chapter 2).

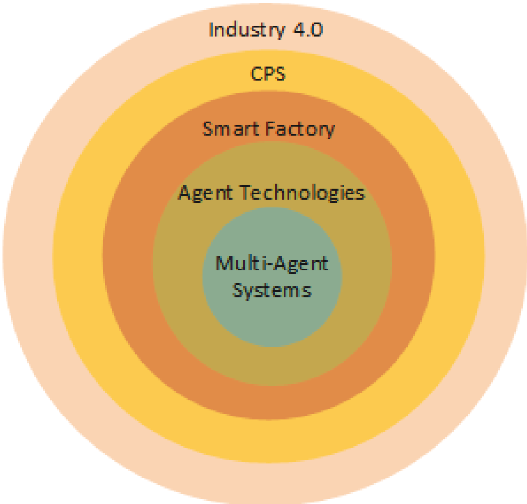


Figure 1.1: Research Focus

1.3 Research Objective

This research focuses on providing a solution to bridge the gap for a framework under the umbrella of Industry 4.0 that may suit SME-oriented environments to solve the problems of process integration and also extend support to manage the crucial aspect of warehouse management. This thesis thus aims to provide a two-fold solution to enhance the overall integration within the supply chain network and to reduce the number of issues within the warehouse shop-floor by providing:

- *a CPS-based SC framework under the umbrella of Industry 4.0 which may support overall enterprise integration within the SC network for SMEs*
- *an associated adaptable system to provide a solution for baseline warehouse issues in SMEs*

From the perspective of enabling technologies in Industry 4.0, this research is focused on agent orientation as a core technology, exploring the dimension of Multi-Agent Planning (MAP) in the domain of MAS. For the purpose of implementation and validation, this thesis includes a prototype for the warehouse side as a key area for utilising the proposed solution. The scope of this research does not include the concerns of process control in manufacturing automation, e.g. hardware interoperability, fault tolerance, resilience and industrial compliance, which require high-quality measures of commercialisation, testability, maintainability and scalability from the industry side.

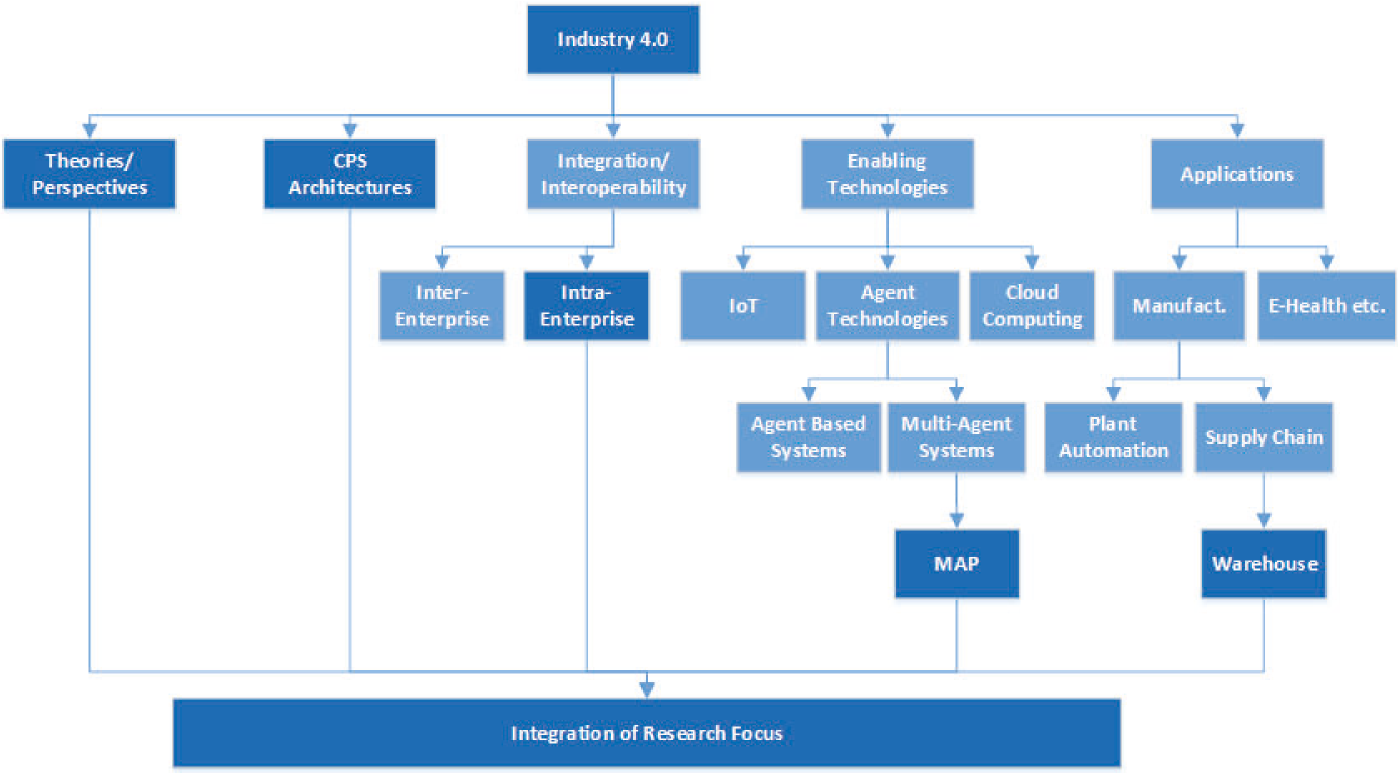


Figure 1.2: Research focus in Industry 4.0

As depicted in Figure 1.2, research in the domain of Industry 4.0 is subdivided into five major categories [53, 106, 111] including Theories/Perspectives [183, 184], CPS architecture [102], Interoperability/Integration [16], Enabling Technologies [2, 4, 83] and its Applications [30]. The research presented in this thesis incorporates the concepts from theories/perspective of Industry 4.0 and is based on CPS architecture. From the perspective of Enterprise Integration (EI), initially, this research is focused on intra-enterprise integration only, which may further be scaled to cater for inter-enterprise integration in the future. Some of the main enabling technologies to implement Industry 4.0 frameworks include IoT, Cloud Computing and Big Data, but the technological focus of this research is to provide a solution incorporating planning strategies from the MAS domain, usually referred to as MAP. From the widespread application domain of such solutions in manufacturing in general, which includes the implementation of plant-side automation, this research only focuses on the SC side, particularly warehouse implementation. Figure 1.2 summarises all this detail pictorially.

1.4 Research Questions

For many of the questions referring to the implementation and application of Industry 4.0, there exist many solutions, but from the perspective of compatibility with SMEs, some questions are still unanswered or only partially answered. The purpose of this research is to find a solution for the following research questions (RQ):

- **RQ1:** *Which framework, under the umbrella of Industry 4.0, is compatible with SMEs to improve the issues of warehousing? [5, 8, 119, 133, 151]* Do Industry 4.0 standards focus on SMEs explicitly?
- **RQ2:** *Can MAS technologies provide Industry 4.0 benefits to SMEs? [95, 111, 183]* What MAS solutions exist under the umbrella of Industry 4.0 to facilitate the standards?
- **RQ3:** *How can issues of warehousing be resolved for SMEs using Industry 4.0? [60, 110, 149, 156]* If SMEs cannot afford the high-tech robo-oriented warehousing system then how can the problems of warehouse management be resolved? Do the existing standard WMS strategies meet the management requirements of SMEs-oriented warehouses?

1.5 Research Methodology

The research for this thesis follows the elements of a structured research-problem-solving methodology [33]. It begins with research questions, identifying a potential research gap and then detailing the ideas with existing contributions. In order to propose a strong solution a comprehensive literature review is necessary with a subsequent phase of testing and validation. This research is segmented into eight distinctive but contiguous stages, as detailed below:

Phase 1: The first phase was to identify a research gap. For this purpose, the initial literature review was conducted. After identifying the achieved progress-milestone and contributions in the published literature, the potential research problem was finalised through discussion, criticism and mutual agreement among fellow researchers and peers.

Phase 2: For further investigation, the second phase of extensive literature review was conducted to get more responses and ideas from contributing researchers around the globe. This provided more details to fine-tune the problem statement and provided insight for a sound solution. This phase resulted in formulating the research questions.

Phase 3: Compiling the studied material and recommendations made by other researchers allowed formulation of a baseline solution for the identified problem. In order to provide a solution for RQ1, the idea of the Agent-Oriented Smart Factory (AOSF) framework was proposed. This proposition led to a book chapter [41], presented at the Springer-International Conference of Agents and Multi-Agent Systems and Technologies (AMSTA), held in Gold Coast, Australia in June 2018, which won the Best Research Paper Award.

Phase 4: To move on from the AOSF framework's recommended CPS-based SC network for SMEs under Industry 4.0, the next step was to validate the overall system by providing an implementation of it on the warehouse side. For implementing the proposed agent classification and negotiation strategy, it was necessary to design formal problem and domain definitions. For this purpose Multi-Agent Hierarchical Task Networking (MA-HTN) constructs were utilised [25]. This phase resulted in an IEEE-Conference publication on the AOSF framework's problem and domain definition [42].

Phase 5: Phases 3 and 4 provided a sound foundation for designing proper algorithmic heuristics for the AOSF-recommended Agent-Oriented Storage and Retrieval warehouse management strategy (AOSR-WMS). In Phase 5, the structural flow of the agent planner, on the warehouse side, was designed.

Phase 6: On the basis of these defined heuristics, the next step was the development of a prototype to test and validate the performance of the overall system. After exploring the possible available options of tools for developing an agent-oriented system, Java Agent Development Environment (JADE) [81] was selected as the main tool to develop this prototype.

Phase 7: After selection of the tool, the algorithmic heuristics were implemented, and results from a realistic simulation were acquired. This provided a good comparison of performance with existing systems.

Phase 8: The last stage was to analyse the performance of the system in compliance with the research questions identified at the beginning of the research. Revisiting the research questions ensured that the highlighted questions were fully answered with this research. The last phase also concluded the work with some future implications which are possible with this scalable system.

1.6 Thesis Structure

This thesis uses a hybrid structure: some of the included chapters have been submitted or accepted as publications. Chapter 4 contains an already published Book Chapter [41], Chapter 5 also includes a published IEEE conference paper [42] and Chapter 6, contains a journal article currently under review with Journal of Cybernetics and Systems. This thesis is structured in nine chapters, described below:

Chapter 1 provides an introduction to the preliminary concepts, such as Industry 4.0 and warehouses. It also includes research objectives and questions. The methodology utilised to complete this research and thesis structure is also included in this chapter.

Chapter 2 casts light on the background concepts, such as CPS, SC architecture, WMS and their interrelationships. This allows a better understanding of the proposed system in later chapters.

Chapter 3 provides the possible options of available tools for agent-oriented development and a comparison of existing systems, leading to the decision to use JADE for this research. It also includes details of agent-development methodologies that may help in the creation of an elegant agent-oriented prototype.

Chapter 4 includes a book chapter that provides details of the AOSF framework, its recommended agent classification and their communication strategies. It explains not only the details of three tiers of the framework but also the three-dimensional integration that the framework offers. This chapter also includes details about how the initial experimentation is conducted and how the AOSF framework provides three dimensional enterprise integration.

Chapter 5 includes a conference publication, presented in IEEE Region 10 Symposium [42], on the topic of how the problem and domain definition can help in developing the AOSR system in an effective way. It provides details of MA-HTN based Backus-Naur-Form (BNF) constructs and the decision trees for agents' primitive and non-primitive tasks, to help them reach their goal states easily and accurately.

Chapter 6 includes another journal publication, submitted to the International Journal of Cybernetics and Systems, which includes an overview of algorithmic heuristics of the AOSR system. It also presents the detailed *6-Feature* strategy recommended by this system, in an attempt to reduce the overall warehousing issues for SMEs.

Chapter 7 presents a synopsis of the previous chapters (Chapters 4, 5 and 6) and connects the ideas together before progressing to a thorough validation of the whole system in Chapter 8. Chapter 7 also provides details to justify the need of the proposed system and summarises the contributions made so far. In the light of recent research literature, it explains the importance and novelty of the work contributed by this thesis.

Chapter 8 presents a complete validation of the whole system. It provides test scenarios within the supply chain of a firm and relates it with different cases of information exchange from the front-end customer side to the back-end supplier side. This chapter also includes a complete overview of AOSR algorithmic heuristics with test data sets and their implementation results taken from the JADE prototype.

Chapter 9 includes a methodical conclusion of this research thesis by revisiting the research questions highlighted in Chapter 1. It summarises the whole work and links it with each of the research question to provide the right perspective. It also includes suggestions for possible future works to provide insight about scaling or upgrading this system in several dimensions.

1.7 Contributions

This thesis presents a CPS-based end-to-end comprehensive SC architecture for SMEs under the umbrella of Industry 4.0, with its associated moderate level semi-autonomous warehouse management strategy in order to reduce warehousing issues in SMEs. The work conducted for this research has resulted in the following publications, produced as part of this thesis:

- Ud Din F., Henskens F., Paul D., Wallis M. (2018) "Agent-Oriented Smart Factory (AOSF): An MAS Based Framework for SMEs Under Industry 4.0". In: Jezic G., Chen-Burger YH., Howlett R., Jain L., Vlacic L., Šperka R. (eds), *Agents and Multi-Agent Systems: Technologies and Applications 2018*, KES-AMSTA-18 2018. Smart Innovation, Systems and Technologies, vol 96. Springer, Cham

- Ud Din F., Henskens F., Paul D., Wallis M., (2018) "Formalisation of Problem and Domain Definition for Agent-Oriented Smart Factory (AOSF)", in *IEEE Region 10 Symposium (TenSymp), IEEE, 2019, pp. 265-270*.
- Ud Din F., Henskens F., Paul D., Wallis M. and Hashmi M., (2019), "AOSR-WMS planner associated with AOSF framework for SMEs, under Industry 4.0", In review with *Cybernetics and Systems*.

1.8 Achievements and Recognition

- **Best Research Paper Award** at International Conference on Agents and Multi-Agent Systems: Technologies and Applications 2018, Gold Coast, Australia
- **Best Technical Research Poster Award**, at Poster Competition 2018, Faculty of Engineering and Built Environment, The University of Newcastle, Australia.
- **Best Academic Achiever of the Year 2019**, from the UNIS Society, The University of Newcastle, Australia.

Chapter 2

Theoretical Background

In order to provide necessary background information, this chapter presents an overview of some concepts related to Industry 4.0 and SMEs in Section 2.1, which includes details of the CPS architecture as a basis for Industry 4.0 standards in Section 2.1.1 and the details related to SC architectures in Section 2.1.2. In order to maintain the whole business value-chain, particularly in manufacturing organizations, the role of warehouses is crucial. Details of purpose-built software that supports the whole warehousing process, are mentioned in Section 2.2. There are certain standard warehousing strategies employed by industry to manage products within a warehouse; some of these strategies are discussed in Section 2.2.1. Automated solutions, including robots and conveyor belts, are also utilised in industries incorporating high-tech infrastructure. Section 2.2.2 discusses some of these robo-oriented solutions. All these background concepts will help aid understanding of the hybrid logic utilised by the system presented in subsequent chapters and its comparison with standard baseline approaches.

2.1 Industry 4.0 and SMEs

With technological evolution in industrial architectures, enterprise setups are becoming more complex. There exist several solutions for enterprise integration, which are based on object-orientation (OO) [7, 46] and component-based architectures [87]. However, intelligent agents and multi-agent systems (MAS) based enterprise integration applications have also been proposed by many researchers [84,88,89,177]. Finding a universally acceptable solution is very difficult and even modelling agent architecture to resolve the complexity is not easy; particularly in the domain of enterprise information systems, where integration needs agility to meet current competitive demands [113]. MAS technologies provide better fault tolerance with rationality by providing decentralised decision making [123]. Before the inception of Industry 4.0 different ideas have been contributed to provide seamless operations in the area of automation of production systems based on agent technologies, such as the works contributed in [3, 50, 112, 117], but with a focus on plant automation. The core observation in the scenario of making a factory "smart" is to make a larger set of components interact and perform seamlessly to provide ease of access and improved quality of processing. For SMEs, automation is not the only concern; solutions providing improved efficiency, integrations and quality with low infrastructural cost have always been a preference [127].

For managing enterprise resources, enterprise application software (e.g. ERPs) provide support to manage business resources. Some also provide a specific solution to manage medium size businesses e.g, SAP Business One [163] and Oracle SMB [140], but, because of less flexibility in the requirements and rigidity in customisation [161], the issues of compatibility with SMEs still exist. A report published by Massachusetts Institute of Technology (MIT) Global SCALE: Supply Chain and Logistics Excellence Network [152], highlights the importance of optimisation strategies in managing an enterprise's resources. It concluded with the stance that there might be more research needed in the domain of predictive analysis in ERP solutions, such as including a planning part in order to improve decision making.

The concept of Industry 4.0 is no longer new, and provides a more flexible and less expensive solution than traditional ERP systems [127]. Since its inception in 2011, extensive research has been conducted in different dimensions in this area. Recent research claims that, in order to implement the idea of Industry 4.0, it is necessary to connect three integration levels in an enterprise [183]:

- horizontal integration, connecting all the sub-units of an enterprise together;
- vertical integration, coordinating along the hierarchical chain within the units of an enterprise; and
- end-to-end integration, linking the selective units for customised production chains.

The inclusion of a cloud-based network in Industry 4.0 standard provides a smart architecture to overcome the limits of hierarchical mediation. A possible implementation of such an architecture, focusing on large setups, is discussed in [184], which presents a MAS-based solution to provide a thorough process-control mechanism. Literature also includes a broader domain of Industrial Automation and Control Systems (IACS) networks, e.g. Cisco's Ethernet-to-the-Factory (EttF) architecture [32] and Rockwell

Automation's Integrated Architecture (AIA) [157]. Both EttF and AIA are manufacturing control system architectures based on de-facto Ethernet-based networking standards to provide value within industrial operations. Another more general but comprehensive control system architecture is called the Converged Plantwide Ethernet (CPwE) architecture [10], which connects these two architectures together. The main issue is the mismatch in compatibility of such large setups with SMEs, not only on the Cyber-Physical Systems (CPS) side (e.g., robotics, RFID, environmental sensors), but on the software technology side as well because the solutions for large setups are usually rigid in requirements [161] and incur high cost to customise. There exist several other contributions regarding Industry 4.0 and enterprise setups, including the concepts of supply chain based implementations e.g., [75] and [115], but because of the requirement for large structural change and affordability, the issues for SMEs still exist [133]. Hence, as highlighted in Chapter 1, this thesis does not focus on control system automation but attempts to provide a general SC architecture to reduce the baseline problems of SMEs such as warehouse management.

Brettle [22] highlighted three main focus areas in Industry 4.0 with respect to the SC implementation:

1. Individualised Production, catering to the requirements of mass customisation;
2. Horizontal Collaboration, inter-unit interaction in different hierarchies; and
3. End-to-end Digital Integration, providing digital/virtual access to the entire value chain.

Individualised Production, is an alternative name for mass customisation, which is a production strategy to promote personalised products on a massive level by incorporating all stakeholders into the value chain. This concept is successfully implemented in [34] and [44]. In order to bring improvement in decision making, horizontal integration within collaborative networks in the value chain, is also important. Thus, to increase performance efficiency, particularly for SMEs, an intra-enterprise communication mechanism that may support a random resource planning strategy is crucial [44]. The details of how these recommendations are catered for by the system proposed in this thesis are addressed in Chapter 4.

2.1.1 Cyber Physical Systems

The concept of Industry 4.0 relies heavily on the application of Cyber-Physical Systems (CPS), which is an interface of human-machine interaction. CPS is a five-layered architecture [102] which provides a controlling and monitoring mechanism to link all subsystems together. Figure 2.1 depicts the hierarchy of different layers in a CPS. An extended CPS in production and manufacturing industries is normally termed as Cyber-Physical Production System (CPPS) [131]. In CPS/CPPS, the smart devices and embedded systems are usually connected with machines at the first level of the hierarchy. Conversion and correlation of data are performed at the second level, which maintains a secure connection to the cyber layer at the third level. Configuration is performed on the basis of cognition that is provided from the lower three layers. On top of this architecture, there are electronic interfaces, which are controlled via software modules to provide an overall layout to the computerisation system for communication and collaboration with other subsystems [22].

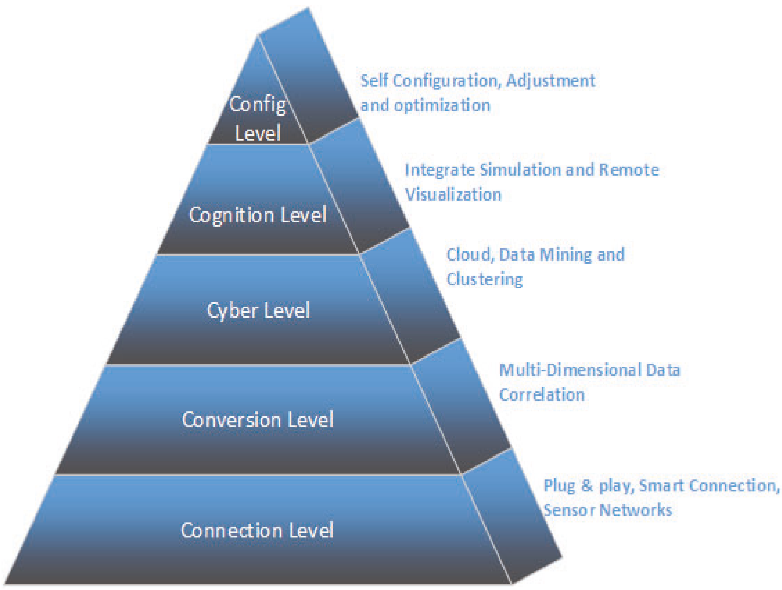


Figure 2.1: Hierarchy of a CPS architecture [102]

In an Industry 4.0 setup for manufacturing organisations, plug & play devices such as RFID/barcode scanners, weighing/pressure sensors and threshold detectors are usually installed at the Connection Level (the base level). For collecting data from these devices and categorising it as needed, correlation and conversion are performed at the Conversion

Level, which passes the meaningful information e.g., total stock values or orders in a queue to the Cyber Level. At the Cyber Level, cloud architecture is maintained, with data mining and clustering of features to provide decisive information. At the Cognition Level, predictive simulation and visualisation provide support for better decision making e.g. to increase or decrease production rate. The top layer of Configuration provides auto-adjustments and optimisation based on the feedback received from several different subsystems in the unit. All these levels can provide their benefits in a production system only when they are efficiently attached to the SC network. In SMEs, there are high chances of maintaining ad-hoc operations, hence flexibility within the CPS architecture and compatibility with the SC network hold a crucial role.

2.1.2 Supply Chain Architecture

For any industry, the supply chain (SC) holds critical importance as it links the constituent entities of an organization together. Literature defines SC as an integrated manufacturing/transformation of raw materials into finished/semi-finished products and delivering them to end-customers via distribution channels [197]. It originates from the back-end supplier side and encapsulates all the intermediary stages incorporating the front-end customer side. Our previous work [40], related to SC concepts and ERPs, explains in detail about SC architecture and subdivides it into three parts:

1. Inbound SC, including the SC elements and their interaction from the supplier side;
2. In-plant SC, including process and operations related to production system within the whole value chain; and
3. Outbound SC, including the SC elements and their interaction from the customer side.

The procedures involved in the receiving of materials from suppliers are considered a part of Inbound SC and the procedures which involve customer interaction are considered a part of Outbound SC. All the stages in-between these two ends are part of In-plant SC e.g., distributed production, product displacement/transportation and warehousing. Research has provided substantial support for improving SC architectures but they are still not fully optimised, particularly in SME-oriented environments because of limited resources and other challenges [182]. A descriptive case analysis of SC networks is recalled and discussed in detail with validation in Chapter 8.

Within the SC architecture, integration has been considered a vital characteristic in literature [143], in order to provide business growth and stability, but even after the inception of current high-tech solutions, integration in supply chain is not well advanced, particularly in SMEs [62]. The manufacturing industry has gone through increased globalisation of markets in recent years, which demands a high level of coordination and integration between SC elements. Recent SC networks are generally more complex than in the past and hence pose interesting challenges for effective supply chain performance. Large industrial setups usually employ comprehensive SC architectures effectively but it is not always true in the case of SMEs [6], which affects the overall business performance.

In modelling supply chain architectures, one of the major research concerns is the 'Ripple Effect', which affects large supply chains heavily [76]. The ripple effect is a propagation of a low frequency/high impact disruption throughout the whole supply chain. It is also known as the Domino Effect and Snow Ball Effect in supply chain literature [176]. There are multiple other disruptions in the supply chain e.g., 'Bull-Whip Effect', which is a high frequency/low impact disruption, which impacts more fatal disruptions to the overall supply chain [75]. MAS technologies, in parallel with CPS, and in coordination with other associated technologies e.g. IoT, Big Data and Cloud Computing, can provide support for developing more robust systems to alleviate such issues in SC architectures [184].

2.2 Warehouse Management Systems

Warehouses are the real backbone of the supply chain for any production/manufacturing organisation [1]. A general distribution-warehouse includes many operations such as receiving, storing, picking and shipping. Receiving is a process where the products and the information about products are received at the beginning of the overall warehousing procedure. The process of receiving products may include identification and inspection of products as per delivery notices. Usually the warehouses have a designated Receiving Area (RA) for this purpose. The area within the warehouse, where the products are loaded into the delivery trucks, is usually referred to as Shipment Area (SA). The storing process initiates with a validated receiving process, which involves finding a suitable storage location and then storing the products respectively. The storage area in a warehouse usually includes racks and floor buffers, which can be further segmented into different zones as per products' characteristics and business needs. A warehouse may also have some Expedition Areas (EAs), which are temporary floor buffer storage locations for products waiting to be stored or shipped. The shipping process includes finding required products from the storage area, picking them in the requested quantity as per shipment notices and taking them to the SA. Some warehouses also include an additional process of packaging and clinging too, to unpack and then pack the received products again to fit them into previously defined fixed capacity storage racks.

For managing warehouse activities, the role of the receiving process is critical, as in usual human-operated warehouses, the products are placed in EA first and then identified and matched with the corresponding shipment notice [59]. After that, they are packed as per the available storage capacity. The overall process is a time-consuming activity and leads to the issue of RA overloading. After the identification of products, the next step is to search for a suitable location with matching characteristics either in EA or in racks. For this purpose, efficient division of warehouse zones and racks is crucial. The process of storing products includes identification of empty racks and placement of products in preferred locations. In the case that no suitable storage location is available, products are placed in EA, but if the zones are not defined properly, a large number of products placed in EA may overload the area and hence increase the likelihood of lost and wandering items [23]. For any delivery order, the picking process is initiated on the basis of identification of products in storage locations and, after finding and retrieving the specific products from a designated location, the process of shipping is invoked, which may also include the packaging process.

A warehouse is a facility that provides the whole supply chain with a mechanism to maintain overall low transportation cost and increase economical achievement as well as reduce response time [9]. Warehouses can provide custom-made services for managing products whether the products are finished or raw (unfinished). Handling warehouse operations is practically impossible without a Warehouse Management System (WMS) [54]. A WMS is a software application that supports day-to-day operations in a warehouse. WMS programs enable centralised management of tasks such as tracking inventory levels and stock locations. ERPs typically include a separate WMS Module, which provides support with information processing features within the procedures of receiving, storing and shipping the products, but because of low provision of customisation in ERPs [161], the industry is still facing warehousing problems, e.g. the aforementioned issues of expedition area overloading, receiving area problems, definition of zones in warehouse and division of racks [9].

Previous research presents multiple projects, focusing on smart enterprises, particularly from the warehouse perspective, using agent technology to overcome the gap in traditional manufacturing systems e.g. the works presented in [24], [123] and [126]. The work mentioned in [96] is also related to warehouse management and a process control mechanism with an architecture of a specific warehouse of Lareal Company with a predefined number of categories of zones, which is similar to the ones presented in [52] and [89] proposing a macro level view of the whole supply chain, but the management problems of warehouses still persist [110].

2.2.1 Standard Warehousing Strategies

As discussed earlier in Section 2.2, WMS systems, though varying in requirements as per the type of industry, follow an already defined set of standard operating procedures such as receiving the products, placing them in storage areas and then picking them as per customer orders. This section discusses the details of some of the standard WMS strategies. The very first step to perform product slotting, or storage-assignment, is the definition of a Stock Keeping Unit (SKU). On the basis of the defined SKU, the product category/class is then assigned to both the product and storage locations. Choosing product-class assignment and classification of storage locations is dependent on the individual warehouse scenario and layout. A changing design layout makes product-class and storage-class definition more complicated. Product classification is affected by several different factors, for example, based on turnover metrics, where the products are defined by Cube Per Order Index (COI) [93]. As per COI rule, fast moving products (frequently ordered) or heavy products are placed near the shipment area to reduce the overall activity time. For SMEs, the chance of altering warehouse layout/requirements is high, hence they need a generic and highly flexible solution [98].

There are five broader commonly used storage-assignment strategies in industry [36]:

1. Randomised logic, where a random location from the available racks is selected;
2. Nearest logic, where the locations closest to RA or SA are selected;
3. Dedicated Storage logic, where products are stored only in fixed zones;
4. Turnover logic, where the location of a product is defined based on its frequency of occurrence in shipment or delivery notices; and
5. Class-based logic, where the products are stored as per a certain class/category based on products' characteristics.

As per Randomised logic, the products are stored in a randomly selected, available location [146]. It is normally utilised when there is only one product-class to be stored in a warehouse. The strategy for placing the products in the nearest available location in a computer-controlled environment is known as Nearest logic, which results in a scenario of having more products near RA and less on the back-end side [64]. Warehouse-layouts sometimes fix the locations for particular products, which is generally referred to as Dedicated Storage or Zoning logic [100]. Another logic is called Turnover logic (similar to COI logic), according to which the products are placed on the basis of their frequency/COI [93]. Fast-moving products are stored near the shipment area and slow-moving products are placed at the other end of the warehouse. A Class-based logic is a combination of the aforementioned strategies, and depends on the type and layout of the warehouse [36].

There are some other known storage strategies in warehouse scheduling e.g., Shortest Processing Time (SPT) logic, where warehouse jobs with the shortest processing time are scheduled first. Similarly, Earliest Due Date (EDD) logic sequences the product-picking process according to their due dates, and Critical Ratio (CR) logic is the combination of both aforementioned logics, which computes the ratio of SPT of the job to the EDD and schedules the job with the smallest CR value first [108]. Another strategy is known as First Come First Serve (FCFS) or First In First Out (FIFO), where products are stored in the sequence of their arrival to the shopfloor [86]. The early-arriving products will secure a place first in the storage area, and will be shipped sooner than the products delivered later to the warehouse, for a respective request. Another storage-assignment strategy is called Slotting by Affinity logic, where the products that are frequently ordered together are stored together. Such products are called Affine Product in literature but are not very common in a normal distribution warehouse [61]. Another product placement or picking strategy is called Pick from the Fewest/Put to the Fewest logic [150], where the priority is to fill a complete rack for product placement or to clear a complete rack for product picking in order to increase space availability. All these strategies are applied on the basis of an individual firm's requirements and characteristics of the warehouse to cater for all the business needs.

2.2.2 Robo-Warehousing

Technological advancements in the WMS domain are not new and have been evolving since the 1970s [159]. For accomplishing the Japanese success formula of Just In Time (JIT) [72], the supply chain and warehouses need to be efficient and smart to bear the pressure of simultaneous operations. Efficient warehouse management contributes to the timely delivery of the product, which ultimately increases business efficiency [31]. Modern warehouses are equipped with storage and retrieval (S/R) machines to pick up products from an input/output (I/O) location and store them at their specific locations, and then to retrieve outgoing products from other storage locations and deliver them to the I/O location.

There are certain algorithms that are utilised by researchers to demonstrate the working of standard warehousing strategies e.g., Pick Frequency/Part Affinity function algorithm [99] and Minimum Delay Algorithm (MDA) [196], which utilise the Affinity model to store the products based on their combined occurrence and frequency. This model is suitable for a particular scenario where there is a possibility for a combined occurrence of products. For scenarios where this is not possible, a parallel different storage strategy may help cater for the other cases such as defining zones, stocking products based on their expiry dates or prioritising fast moving products to store closer to the shipment area.

Warehouse scheduling is a typical NP-hard problem, and is one of the most challenging types of combinatorial optimisation problems [54]. One of the sound existing algorithms for warehouse management is Centobelli's Flow-shop scheduling [28], which provides support for controlling low-end operations in warehouses and focuses on order picking efficiency. Flow-shop supports the basic strategies of FCFS and CR logics. It recommends a constraint of the two-step method of product retrieval:

1. lowering the product from top-level racks and then separately; and
2. picking the products as a second job.

These two operations are carried out by two different forklift truck operators. The flow-shop algorithm recommends the execution of the work-flow be started with order acquisition, which leads towards loading and dispatching processes by generating an orders list on the basis of SPT and LPT and lowering products from higher shelves to lower shelves. This mechanism supports product retrieval but, for product placement, some parallel storage-assignment strategy other than FSFC and SPT might help, in order to optimise sorting and space availability within the warehouse in case of already overloaded storage locations. Furthermore, the inclusion of two buffer locations, dynamic buffer and static buffer, for leaving products for other forklift trucks, may cause other issues such as floor overloading, vanishing demarcation lines for truck tracks and lost/wandering items.

Another robo-oriented approach for warehousing is Ensemble Multi-objective Biogeography Based Optimisation (EMBBO), which is an optimisation algorithm for automated storage and retrieval in a warehouse. It is based on a series of biogeography based optimisations e.g., Vector Evaluated, Non-Dominated Sorting and Niched Pareto biogeography based optimisation [114]. EMBBO addresses warehouse scheduling with certain assumptions, such as that a warehouse should be designed to include multi-aisle racks on both sides and a single storage rack along each warehouse wall. It recommends to have only one storing and retrieving (S/R) machine. The S/R machine is able to move across the warehouse aisle by using the curved rails at the end of the picking aisles. S/R machines travel at a constant velocity both in the horizontal and in the vertical directions. Acceleration and deceleration affect the scheduling results in this algorithm, which is based on randomised storage assignment logic. There also exist some commercial solutions, for providing robo-oriented solutions for warehousing such as GrayOrange [186] and Unleashed [187], which utilise the same strategy as recommended by EMBBO. Such solutions take warehouse systems in the direction of complete automation, which may suit large enterprises but is not always the best choice for SMEs, not least because of the question of affordability, such as infrastructure and equipment cost [109].

2.3 Summary

In this chapter, we have discussed some of the background concepts, which help build a better understanding of the existing problems and the need for the proposed solution. So far we have only discussed the information which is necessary to understand the theoretical grounds of the novel approach presented by this thesis. We have discussed how existing Industry 4.0 solutions (e.g. IACS, EttF and AIA) support large setups but do not comply with SMEs' requirements of SC flexibility and customised CPS. This chapter also included details about current issues and requirements of SMEs, especially in terms of its compatibility with Industry 4.0 e.g. as in warehousing. In the next chapter, we discuss what types of tools and methodologies exist to create a prototype to validate the proposed system and what features the selected environment must provide, which makes it suitable for this problem and the solution addressed in this thesis.

Chapter 3

Overview of Agents and Related Technologies

Now as the background concepts have been made clear, the next chapters will present details about the proposed system. Upcoming chapters (Chapters 4, 5, and 6) are journal/conference articles, which include the results from the prototype developed for the proposed system. As such, there is a need to explain which tools are available for developing such systems and which tool/environment was selected for the prototype development for this thesis. Recalling from Chapter 1, the domain of agent-oriented technologies is one of the focus areas of this research, as it provides better fault tolerance, design flexibility, embedded intelligence and decentralised decision making, which is needed to build a comprehensive and scalable solution. MAS technologies support cloud architecture as well, which will further be utilised for building a conceptualised CPS architecture in this thesis. This chapter explicitly describes existing agent-oriented tools and methodologies and specifies how that information was used to select the right tool for this thesis.

In general, an agent is defined as an entity that can perceive its environment through its percepts or sensors and act accordingly through its effectors. Software Agents are computer programs which can perform their tasks intelligently and rationally [160]. Rationality (selecting the appropriate logic, comprehending the possible reasons for every trigger) is the main concern for software agents. Based on certain features and characteristics, agents may have multiple types, such as:

- Simple Reflex Agents, with simple if-else logic;
- Learning Agents, with the features of updating their knowledge-base; and
- Utility/Goal-based Agent, doing certain tasks to achieve a defined goal.

Further details of agents and their types is explained in Chapter 4, which also includes details about inter-agent communication mechanisms. Foundation for Intelligent Physical Agents (FIPA) [49], which is a standards organisation that promotes agent and MAS technologies and their interoperability, provides the standard of Agent Communication Language (ACL) for specification of agents' interaction and communication. Compliance of MAS technologies with FIPA-ACL standard increases interoperability [91]. While considering the design of an agent, there are certain factors which are considered important in developing agent-oriented systems [97], such as:

- agent's role: such as manager, coordinator or subordinate agent within the network;
- goal, which may be the definition of a desired state;
- interaction, how the agent can interact, and with whom;
- task, what tasks are provided for an agent to initiate;
- resource, what resources can the agent utilise;
- information, what kind of information can the agent access and what can it do with it; and
- knowledge, which is pre-stored information and can be updated when experiencing the environment.

In developing software systems and prototypes, programming tools and environments hold a vital role [20]. In order to experiment with the agent-oriented planning features more precisely, existing agent-oriented tools provide robust environments for the purpose of testing and validation e.g., JACK [70], JADE [81] and JaCaMo [78]. Although all these systems provide compliance with FIPA-ACL standard [48], including agent design, message passing and intermediary negotiation, they vary in their ontological architectures and implementation techniques. Thus, selection of the right prototyping tool becomes important to properly visualise processes in any implemented system. The variation of these systems and methodologies are discussed later in this chapter. Some results from the validation of the proposed system in JADE (which has been taken as the prototyping tool in this thesis) are included in Chapters 4 and 6. In order to build an understanding of the prototype for the validation of this system, this chapter presents a brief description of available tools and their features. It also clarifies an important difference between agent-based system development and agent-oriented system development. Some of the common agent-system-development methodologies are also included as part of this chapter. Later in this chapter is a brief description of the process of designing AOSF agents and the interpretation of their output through JADE.

MAS is a suitable technology for developing adaptive, autonomous, robust and complex industrial systems under the umbrella of Industry 4.0 [103, 106, 111]. A possible implementation of such an architecture for large setups is discussed in [184] and [52], which conclude that, in order to provide supply chain flexibility, a sound communication mechanism that supports a random plan repair strategy to increase performance efficiency is important, especially for SMEs [44]. The literature claims that the initiative of Industry 4.0 and CPS is an opportunity for utilising the planning features of agent-oriented systems [95]. The foundations to avoid problems in developing and validating agent systems are highlighted in [113]. Design and architecture of MAS is one of the elements that enhances the applicability of agent technologies in industry [95] and, for that, selection of appropriate agent development environments, tools and methodologies is pivotal.

3.1 Multi-Agent Development Environments

Analysis and design are very important when implementing an agent-oriented system, particularly compliance with *de-facto* standards such as ACL [48] and Knowledge Query and Manipulation Language (KQML, which is a basic language protocol for software agents' communication) [47]. Extending software development from object orientation to agent orientation, the Belief Desire Intention (BDI) architecture provides a foundation inspired by Bratman's early research work in the 1980s [19], which is a classical paradigm to develop intelligent software agents. The BDI architecture provides a basic direction to design agents which can balance their execution time in deliberating about their plans to achieve pre-defined goals. Winikoff [193] provided a nuanced understanding about testing BDI agents, claiming agent technology is better than procedural configuration while using BDI concepts. AOSF agents are also based on the BDI model and incorporate the standards of KQML and FIPA-ACL, as detailed in Chapter 4.

In the broader perspective, agent-oriented systems (also commonly referred to as multi-agent systems (MAS)) and agent-based modelling (ABM) are two separate research and development tracks [170]. Agent-based modelling is more inclined towards simulating real-world examples (based on reflex actions). On the other hand, agent-oriented systems provide the facility to design applied multi-agent systems. The following sections present brief details on agent-based development systems and agent-oriented development systems with their contributions towards industrial systems' support.

3.1.1 Agent-Based Development Systems

Agent-based modelling supports simulating real-world scenarios such as bushfire evacuation, taxi driver routing, and effects of vaccination using simulation platforms such as Repast [153], Netlogo [137] or MATSim [120], which may include a large number of agents in a single environment. A survey of such systems to simulate manufacturing processes has been presented in [168], in which the authors discussed 21 projects in the area of enterprise integration from which only one project [96] is related to warehouse management. From the perspective of enterprise integration and collaboration, multi-agent systems have contributed several worthwhile works, but an in-depth implementation of the real-time working of autonomous agents is not completely addressed [162].

Agent-based modelling has also been utilised for simulating decision support systems such as a two-part framework of Intelligent Cluster Optimisation (iCOP) [88] for an oil refinery, which simulates the management of supply chain data processing throughout the firm using multiple agents. Another similar work for the automotive industry (tractor manufacturing) is presented in [52], which is based on a multi-agent simulation-based integrated framework to reduce crashes in the centralised supply chain network (including the combination of already existing process-scheduling projects of DISPOWEB [190], KRASH [116], IntaPS [39], FABMAS [130], and ATT/SCC [200] (for details refer to [52]). Although such systems exist to simulate the scenarios using a huge number of reflex agents, for SMEs, a flexible and reconfigurable system using utility/goal-based agents is more suitable, which can reduce baseline issues in critical subsystems such as in warehouse management [41].

3.1.2 Agent-Oriented Development Systems

One focus area of this research is the development of a prototype of an agent-oriented system, for which selection of the right prototyping tool is important. There are several tools available, including, but not limited to, Jason [80] with AgentSpeak, JACK [70], JADE [81] and Jadex [20], all providing compliance with the FIPA-ACL standard [91]. Literature also recommends certain modelling techniques for mapping scenarios into MAS, such as Problem and Domain Definition Language (PDDL) [174] and Hierarchical Task Networking (HTN) [135]. PDDL provides standardisation for AI planning problems and HTN contains techniques that provide a mechanism for modelling the planning formalism into a structured hierarchy. Chapter 5 discusses these concepts in more detail.

In neo-classical AI, environments within the agent systems (also referred to as virtual agent environments) are considered as a first-class abstraction that encapsulates the functionalities to support agent activities [191]. For programming artifact based MAS environments, certain environment-oriented tools are usually utilised e.g., CArtAgO (Common "Artifact for Agent" Open framework) [27]. Such systems also utilise models for MAS organization e.g., Model of Organization in Multi-agent System (MOISE) [128], which follows Artifact and Agent (A&A) architecture model [155].

Another platform for the development of multi-agent systems is Jason, which provides user-customisable features such as handling plan failures and speech-act based inter-agent communication. Jason is an interpreter for an extended version of AgentSpeak [80]. Many recent systems that participated in a multi-agent programming contest [118], used Jason as a core programming tool. Jason has also been used in the development of the multi-agent development framework JaCaMo [78], which provides the combination of Moise [128] and CArtAgO [27]. Another agent development architecture is JACK [79], which is a lightweight agent development architecture with the provision of strong data types using existing programming constructs such as Java and embedded SQL. Although JACK is not a pure AI system and is not based on the BDI model, it strongly supports agent systems and software engineering concepts and provides the features of team-based development, modular development and software reuse functionalities [70].

JADE, which is taken as the prototyping tool for the purpose of this thesis, is an open source project [81] that is widely used for development of agent-oriented systems. It provides two different kinds of features, complying with FIPA standards such as message passing and agent life cycle management as well as providing object-oriented abstraction to extract other FIPA-standard features using Java runtime environment. For message passing among agents, JADE provides the flexibility of inter-platform message passing (which is also supported by FIPA) as well as intra-platform message passing through Java's Remote Method Invocation (RMI). Every agent in JADE runs in its own Java thread, where all the behaviours (functions) are implemented. One of the many promising features of JADE is its thread-per-agent model with intra-agent cooperative scheduling, which includes the implementation of all behaviours (methods) in a stack-like manner and each behaviour completes its execution in a defined time-frame until pre-empted by another behaviour (method). For the sake of simplicity, all the plans of JADE agents are written as Java classes, which are usually called a Plan library . There are several projects which have been implemented in JADE, e.g., CoMMA [145] and LEAP [15]. JADE also provides an add-on, named Jadex [20], for the facility of defining mental states in agents, which helps in emotion-related scenarios (which is not the focus area of this research).

In order to meet the requirement for prototyping AOSF framework with its associated AOSR strategy, JADE provides the required features with simplicity. All the AOSF agents are designed in compliance with the FIPA standard and their communication utilises the constructs of ACL for the purpose of message passing. The feature of a sniffer agent in JADE has also been utilised to monitor the communication between agents. All these details are highlighted later in this chapter in Section 3.3.

3.2 Agent Development Methodologies

For the development of agent-oriented systems, suitable process frameworks and methodologies play an important role. Since the evolution of agent orientation, there have been a huge number of process frameworks and methodologies introduced in the literature including, but not limited to, Gaia [194], ZEUS [138], Tropos [21], Prometheus [141] and NUMAP [66]. Selecting a suitable option that meets the requirements of a scenario is a vital measure towards reaching the overall objective.

Wooldridge and Jennings, the pioneers of agent technologies, proposed an agent development methodology, named Gaia [194], in their early research, to provide a process framework for agent-oriented analysis and design. It proposes a two-level hierarchical structure for designing agents: macro level (societal view) and micro level (individual agent's view). Similarly, ZEUS [138] is another agent development toolkit that facilitates the development of multi-agent systems through collaborative agent component libraries and integrated environments.

In order to support the complete life-cycle of agent-oriented system development, there are certain methodologies that provide the underlying principles and procedures to design a software agent with related notions of plans, goals and states, such as Tropos [21] and Prometheus [141]. Such methodologies provide the complete life cycle of agent-based application development including several stages e.g., requirement analysis, design phase and implementation phase. Design models in Tropos and Prometheus provide agent features e.g. role, goal, plan and dependability, which are designed in Agent Unified Modelling Langauge (AUML) [71]. In the initial design phases, Prometheus is similar to Gaia for providing general phases and flexibility to agent development process but it is different from Tropos as it provides an early requirements phase for detailed analysis of requirements.

A short comparison of these methodologies is presented in [66] with a complete life-cycle for the agent development process, called NUMAP (Newcastle University Multi-Agent Process) [66]. NUMAP provides a modular approach for designing agents rather than defining abstractions such as in Gaia methodology. The concepts used in NUMAP are closely related to BDI-implementation. NUMAP proposes requirements analysis, organisation design, agent design and implementation as key sub-phases in the whole life-cycle for development of agent-oriented systems. AOSF framework is designed by utilising features of NUMAP (as highlighted in Chapter 8) as a process methodology.

For the purpose of resolving complex multi-agent scenarios, different pure agent planner algorithms also exist in the literature e.g., Simple Hierarchical Order Planner (SHOP) [136], which provides hierarchical task network planning to create plans for a specific agent by decomposing tasks into smaller subtasks. AOSR WMS-planner includes the constructs of SHOP, because SHOP simply outperforms the other existing pure agent planners [136] such as TLPlan [11] and TALPlanner [101] and it also includes many features of PDDL [122], such as quantifiers and conditional artifacts. SHOP is a single agent planner that can be used for planning in a single agent environment and single-agent planning in a multi-agent environment.

The AOSF framework utilises its own WMS-planner for its associated AOSR WMS-strategy (as detailed in Chapters 6 and 8) to present a novel hybrid-logic based mechanism for efficient warehouse management. It utilises the concepts of NUMAP as a process methodology for a complete development life-cycle using JADE for its Sniffer Agent facility and other basic agent development features. A brief explanation on interpreting the output in JADE and its built-in features are presented in the following section.

3.3 Interpreting Output in JADE

JADE provides a simple but comprehensive environment for designing agents and visualising agents' interactions. There are certain features that make JADE distinctive in its nature, such as the facility of agent management and configuration via Agent Management System (*ams*), Directory Facilitator (*df*), Remote Monitoring Agents (*rma*) and, most importantly, the Sniffer Agent to assist in managing agent organisation within the working environment. It also provides facilitation for complying with the Contract Net Protocol (CNP), which helps in configuring a task-sharing environment within MAS [198].

The *ams* in JADE provides supervisory control to access the overall agent platform and to register other existing agents within the system. It also provides agents with unique agent identifiers (AIDs), which are saved in the *df*. The *df* holds a repository to store information related to registered, unregistered or modified agents. It also provides search options to locate agent details in complex systems. With the activation of JADE interfaces, the first element that initiates is an *rma*, which helps in the coordination of inter-platform and intra-platform agent communication utilising its remote method invocation (RMI) based features. It creates a GUI event in response to a message transmitted from one agent to the other. Pictorial representation of these facilities is detailed in Chapter 4 with an example of agents' interaction.

JADE provides a separate interface to monitor communication between constituent agents of a system. Using its thread per agent model it provides a separate instance of all the participating built-in and user-defined agents. Chapter 4 includes the details of Sniffer-Agent's GUI representation for the agents' interaction of the proposed system. The case discussed in Chapter 4 includes three user-defined agents (*ecu*, *pa* and two instances of a *crm* agent), representing their communication with each other to achieve

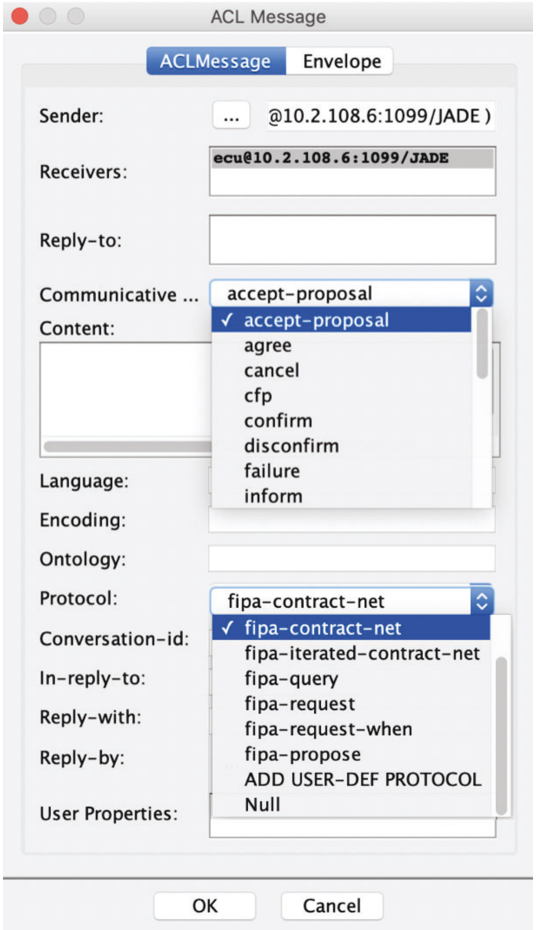


Figure 3.1: ACL Messaging Interface

goals such as to propose space availability for the upcoming products or to accept a proposal/request. JADE also provides an interface for customising the details of corresponding ACL messages between agents, as shown in Figure 3.1, which includes the details related to a certain message transmitted from a sender agent to a receiver agent. ACL communicative protocols can be selected from the provided options such as *request*, *request-if*, *propose* and *accept-proposal*. It also provides options for the specification of the communication protocol such as CNP or query/request based model, with the option of selecting a time delay between receiving and replying to a message. For the purpose of AOSF agents, Figure 3.1 depicts the selected specification for sending a message to *ecu* agent from *pa* agent with an *accept-proposal* message, complying with CNP protocol with no time delays. It represents the features of an ACL message including sender, receiver, reply, communication protocols, and delays if applicable. As displayed in the drop down list, JADE provides certain communicative message types (e.g. request, propose, inform, call for proposal, failure and accept-proposal) and protocols, supported by FIPA.

3.4 Summary

Now as we have discussed the available tools and explored the features for the selected tool for prototype development for this thesis, the next chapters (4, 5, and 6) will move on to explaining the proposed system in detail, including some experiments from the prototype developed in JADE. After discussing the contribution in these three publication chapters in Chapter 7, we will address the proposed strategy in more detail in Chapter 8, where complete validation of the system is included. All the functionality is implemented in JADE using its aforementioned built-in and customisable features to design the recommended mechanisms.

Chapter 4

Agent-Oriented Smart Factory (AOSF) Framework

4.1 Introduction

SMEs constitute a substantial proportion of the industrial sector. To ensure SMEs do not fall prey to the high-tech standard of Industry 4.0, a generic and flexible system is needed to allow SMEs to secure the benefits of Industry 4.0 frameworks without the undue expenses of implementing complete automation [5]. The research question (RQ1) from Section 1.4:

- *Which framework under the umbrella of Industry 4.0 is compatible with SMEs to improve the issues of warehousing? [5, 8, 119, 133, 151] Do Industry 4.0 standards focus on SMEs explicitly?*

aims to find a dynamic solution that may help bring SMEs into the limelight of Industry 4.0 to provide its advanced features such as concurrent information exchange, enterprise integration, seamless operations and cloud-based cognitive abilities. This chapter explores the existing available solutions under the umbrella of Industry 4.0 and discusses their focus areas. Although several Industry 4.0 frameworks exist in literature, they mainly focus on large industrial setups [8], hence this chapter presents the generic but comprehensive end-to-end SC architecture of Agent-Oriented Smart Factory (AOSF) and explains how it attempts to reduce issues for SMEs, particularly from the warehouse side, using its associated hybrid logic-based Agent-Oriented Storage and Retrieval (AOSR) WMS strategy. This work is a copy of the following publication:

Ud Din F., Henskens F., Paul D., Wallis M. (2018) "Agent-Oriented Smart Factory (AOSF): An MAS Based Framework for SMEs Under Industry 4.0". In: Jezic G., Chen-

Burger YH., Howlett R., Jain L., Vlacic L., Šperka R. (eds) Agents and Multi-Agent Systems: Technologies and Applications 2018. KES-AMSTA-18 2018. *Smart Innovation, Systems and Technologies*, vol 96. Springer, Cham

This publication was presented at the International Conference on Agents and Multi-Agent Systems: Technologies and Applications, held in Gold Coast, Queensland, Australia in June 2018 [85]. The paper won the Best Research Paper Award for the conference and is published as a Book Chapter [41] in Springer Cham for the book-series of Smart Innovation, System and Technologies.

The rest of this chapter includes this paper as Section 4.2, which provides a complete overview of AOSF framework, which provides a CPS-based SC architecture, with constituent agents' classification and their communication and negotiation strategy. A few initial results of the implementation of this strategy, in comparison with an existing standard WMS strategy, are also presented in this paper, though the detailed explanation related to the AOSF's associated AOSR WMS-strategy are presented in Chapter 6 and validated in Chapter 8. Section 4.3 provides extra details on how the AOSF agents interact and how they are implemented in JADE. This explanation will help in building more understanding about how the initial experimentation is conducted. Section 4.4 presents details of how the AOSF framework provides three dimensional integration along the whole supply chain to overcome critical issues in SMEs such as lack of real-time stock information, unmanaged inventory and human-operated functional delays. The tier-based architecture of the AOSF framework provides a proper integration mechanism through an Intra-Enterprise Wide Network (IWN), which provides three different types of integration: Vertical, Horizontal and End-to-End. At the end of the chapter, Section 4.5 provides a summary of the concepts presented in this chapter including the comprehensive AOSF framework, its agent classification and three dimensional enterprise integration.

4.2 Publication

Agent-Oriented Smart Factory(AOSF): An MAS based framework for SMEs under Industry 4.0

Fareed Ud Din[1], Frans Henskens[1], David Paul[2], and Mark Wallis[1]

[1] University of Newcastle, NSW, Australia
fareed.uddin@uon.edu.au
[2] University of New England, NSW, Australia

Abstract. For the concept of Industry 4.0 to come true, a mature amalgamation of allied technologies is obligatory, i.e. Internet of Things (IoT), Big Data analytics, Mobile Computing, Multi-Agent Systems (MAS) and Cloud Computing. With the emergence of the fourth industrial revolution, proliferation in the field of Cyber-Physical Systems (CPS) and Smart Factory gave a boost to recent research in this dimension. Despite many autonomous frameworks contributed in this area, there are very few widely acceptable implementation frameworks, particularly for Small to Medium Size Enterprises (SMEs) under the umbrella of Industry 4.0. This paper presents an Agent-Oriented Smart Factory (AOSF) framework, integrating the whole supply chain (SC), from supplier-end to customer-end. The AOSF framework presents an elegant mediating mechanism between multiple agents to increase robustness in decision making at the base level. Classification of agents, negotiation mechanism and few results from a test case are presented.

Keywords: Smart Factory, Multi-Agent Systems (MAS), Cyber-Physical Systems (CPS), Small to Medium Size Enterprises (SMEs)

1 Introduction

A gradual evolution of the industrial revolution, which started in 18^{th} century, is still in progression. The first version started with the incorporation of water-steam mechanical systems [1]. A second revolution arose from incorporating mass production, the division of labour and auto-mechanical implantation in the mid 20^{th} century [2], which yielded a third version incorporating Programmable Logic Controllers (PLC) by 1970s [3]. This rapid continuation led to the fourth industrial revolution, termed as Industry 4.0 by The German Federal Ministry of Education and Research in 2011 [4]. The proliferation of research in the domain of Industry 4.0 has yielded many worthwhile answers to the questions on topics like Theories/Perspectives [5], CPS architecture [6], Interoperability/Integration [7], Enabling Technologies for implementing Industry 4.0 such as IoT [8], Big Data [9], Multi-Agent Systems (MAS) and their applications [10]. Although there is a huge compilation of published literature in this domain, there is still much research to be done from the perspective of integration with Industry 4.0 i.e. supply chain management (SCM) [11], service oriented architectures (SOA) [12], multi-agent systems (MAS) [10], and enterprise resource planning (ERP) [5].

MAS provides better fault tolerance by providing decision making at the local level components [10]. Many solutions are contributed to the manufacturing industry, including but not limited to enterprise integration, enterprise collaboration, process planning, scheduling and controlling shop floor [13]. Prior research has advanced efforts to provide complete autonomous systems, but none of the works focused in depth on the implementation of an agent-oriented smart factory for Small to Medium Size Enterprises (SMEs) [14]. The domain of Manufacturing Process Planning, which is a rich area, where agent technologies are implemented to provide a solution for scheduling resources, is itself an NP-hard problem [15] because of time and probability escalation. To solve the specified problems of resource allocation and scheduling, different techniques are employed in existing literature such as Petri Nets, Genetic Algorithms and Neural Networks [15, 16]. Managing a warehouse is one of the many problems in industry [17]. Extensive research has been carried out in warehouse optimisation in multiple dimensions such as the works presented in [18–20]. One of the many problem in managing a warehouse is the design and structure of warehouse [18]. In a warehouse, Receiving Area (RA) is a place where products, coming to be stocked, are placed first for identification and inspection purposes only and Expedition area is place for temporary placement of products. Keeping RA and EA overloaded, causes the concerns of mismanagement in warehouse [17]. Automated solutions for warehouse management also exist in literature, where the automation is implemented through AS/RS (Automated Storage and Retrieval System via Robo-machines) to pick and ship the products using predefined trajectory and conveyor belts [21]. But in case of Small to Medium Size Enterprises (SMEs) affording such a high-tech system may be difficult [22].

This paper presents a framework of an Agent-Oriented Smart Factory (AOSF), which provides an overall supply chain layout and an agent communication mechanism for SC entities to interact together; to bring robustness in operations. Design of formal semantics and axioms for AOSF-agents follows the concept of eight orthogonal ontological constructs proposed by Kishore in [23], i.e. agent, role, goal, interaction, task, resource, information, and knowledge. The AOSF framework not only provides a generic framework for overall supply chain but also its implementation in Agent-Oriented Storage and Retrieval (AOSR) brings effective results in managing a warehouse for SMEs.

Section 2 addresses the general framework of AOSF with the classification of different constituent agents and their communication strategy. Section 3 includes the results from AOSR warehouse management system (WMS). Some future development in this project is also mentioned in Section 3.

2 AOSF Framework

The architecture of the Agent-Oriented Smart Factory (AOSF: shown in Fig-1), is based on end-to-end Supply Chain (SC) model [24] and Cyber-Physical System (CPS) general framework [6], which is deeply rooted in the concept of Industry 4.0. AOSF framework is extended from an inbound supply chain, towards outbound supply chain, including an in-plant supply chain. This conceptualisation is based on our previous work [24], which explains the structural implementation of an enterprise-wide information management system where now agents are embedded from both ends: the purchase and sales sides. An Enterprise Resource Planning System (ERP) integrates all the other functional areas of a business into one central database. On the rear side it incorporates Logistical Information System (LIS) to maintain in/outbound supply chain and on the front side, it facilitates customer relationship management (CRM). This is how its structural framework provides an advancement in mechanism for a seamless flow of information. Implementing an ERP system only is not enough; pre-implementation and post-implementation factors are also necessary to seek the promised features of an ERP [24]. The next sections provide an insight into the work being presented.

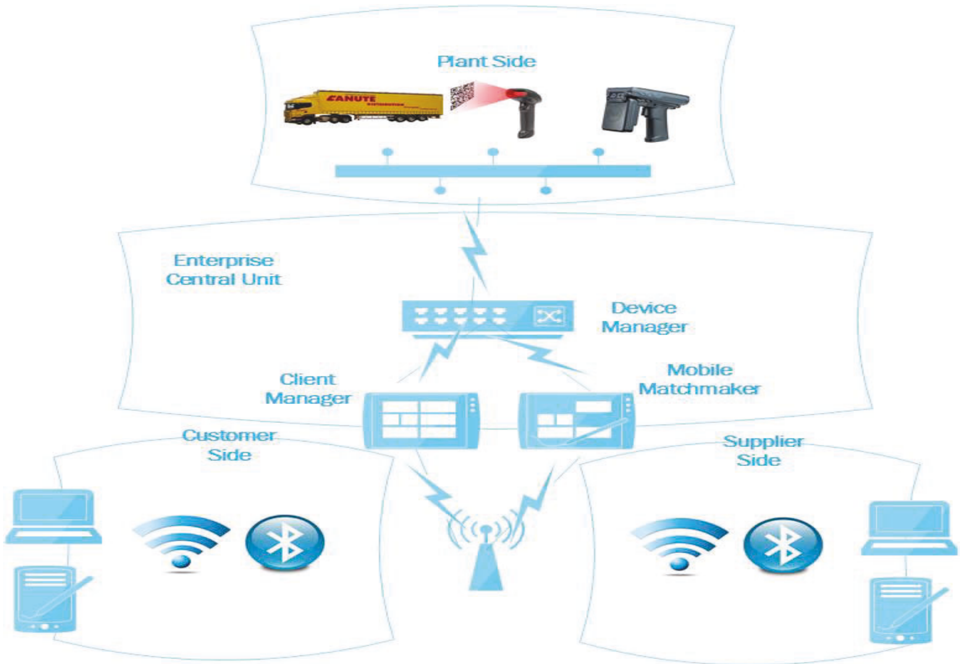


Fig. 1. Three Layers of AOSF Framework

The AOSF framework presents the foundation of a MAS on the basis of already contributed work in literature, to provide a unified structure for making a factory smart enough to perform the operations seamlessly. It does not only present a general framework for the whole supply chain but also provides a classification of agents with their coordination and negotiation mechanisms for resource optimisation. The objective of AOSF framework is two fold: first to provide an architecture for SMEs including end-to-end SC integration in compliance with Industry 4.0 standards; and second to provide intelligence and decision making at the base level which is achieved through AOSR-WMS planner. The AOSF framework is comprised of three main functional components: Enterprise Central Unit, Customer/Supplier Side and a Plant Side.
Enterprise Central Unit: This is a core sub-system that embeds a client manager, a device manager and a mobile matchmaker. These three elements, following the principles presented by Ruta [25], are responsible for providing domotic ambience within the system, as they provide seamless connectivity of resources to the requests coming from different clients. Enterprise Central System performs the main operations in the whole supply chain as it is linked with both ends: the Customer/Supplier Side and the Plant Side. Enterprise Central Unit of AOSF framework can be considered as a CPS, coordinating between the user interfaces and device level. Formalisation of Enterprise Central Unit as a pure CPS is intentionally left for the next phase of AOSF framework development because it comes with the concerns of security and privacy. Device Manager, which is responsible for managing the sensor devices for their current status, keeps track of overall connectivity. It maintains a fact table which includes the details of devices, including IP addresses and device properties. Client Manager keeps track of the clients of the systems, it receives the requests, ranks them and gets them fulfilled in coordination with Mediator Agent (MA). Mobile Matchmaker resolves the matching conflicts with requests coming from the user side and plant side. It is responsible for finding the appropriate correspondence for mobile clients towards a particular device function. These components are attached to the IP backbone of the enterprise including all the other devices in the network, e.g., RFID tag scanners, bar-code scanners, and GPS locators.
Customer/Supplier Side: Conventionally customers are at the front side of an enterprise and suppliers are on the rear side. For SMEs, to follow this convention, AOSF provides a general solution; not only for the integration of in-plant components but also for a Logistical Information System (LIS) at the rear side and Customer Management System (CRM) at the front side. The extension at both ends makes AOSF framework flexible enough to integrate E-Commerce based applications with CRM and procurement analytics-based applications connected with LIS on the rear side. The sub-component of Customer/Supplier Side is collectively termed as 'user level', which sends and receives multicast IP frames through Wi-Fi and Bluetooth technologies in compliance with the standardised rules. Devices connected to this sub-system may be mobile devices such as notebooks, smart-phones or Personal Digital Assistants (PDAs), able to send and receive semantic annotations. We can compare this strategy with the work pre-

sented by Loseto [26] in the domain of Home and Building Automation where the communication between clients and a static system is based on IEEE 802.11 and Bluetooth protocols.
Plant Side: The plant side is a complex domain to model, which may include different machines and devices such as production machinery, temperature/pressure sensors and image analysers. The components of this architecture belong to the same IP backbone to ensure all divisions are integrated. All the sub-systems including Plant Side, Enterprise Side and User Side (Supplier/Customer sides) are designed on the pattern of intelligent agents. Particularly, the running agent on Enterprise Central Unit is responsible for: (i) discovering and coordinating with suitable Plant Side device functionalities compatible with Customer or Supplier Side context requirements via semantic and domotic inferences; (ii) ranking and prioritisation of received requests against the best services to get activated in compliance with the requirements; (iii) finding and resolving inconsistencies in Plant Side′s current status, functionalities and resources; (iv) information storage against the output of matchmaking, negotiation and coordination processes. Enterprise Central Unit is also responsible for the configuration through its Device Manager, Mobile Matchmaker and Client Manager, for example maintaining standardisations, complying with protocols and establishing new bidirectional tunnelling channels to ensure the system is ready to accept semantic requests. Categorization and communication between different agents in this architecture are defined in the next section.

2.1 Architecture of Agents

Agents in the AOSF framework are rational rather than omniscient, in nature. This means an agent in this architecture senses percepts in reaction to some sequence of actions or observations, but it is possible only when the environment change is visible. If some changes occur that are not able to be sensed by the agent, the agent is not responsible for the failure. For example, if a plant side's temperature sensor is supposed to open a valve of a machine without having prior knowledge of more production instructions from the engineering side, then an omniscient agent will complete its task to open the valve as per atmospheric conditions without catering to the instructions from the engineering side. 'Seeking instruction from engineering side', was only possible if it had been previously set in the knowledge base of the agent. Even though the agent includes four basic elements: percepts, built-in knowledge, actions and goal as per [27], it is still incomplete. For maximising performance, the AOSF agents are designed on the basis of, not only the aforementioned four basic constructs but with four additional orthogonal constructs: agent's role, interaction, resource and information. Kishore [23] mentioned these eight orthogonal constructs as a baseline definition for designing agents. An agent is not complete in its nature with an architecture only; it also needs a program to run on a defined set of instructions. This means an agent is a combination of architecture and program.

Agent type is an important part to be addressed for the design of an agent program. There are three basic types of model agents in the AOSF architecture:

Smart Device Agents (SDAs), Mediator Agents (MAs) and User Agents (UAs). Smart Device Agents are generally categorised as simple reflex agents whereas Mediator Agents are knowledge-based and goal-based agents. User Agents are utility-based agents, which try to accomplish the goals of a user as per their needs through a defined coordination mechanism. SDAs are modelled on reflexes which are built into their architecture, where the possible percept-action combinations are previously set in a knowledge base for each agent. This knowledge base can be summarised on the basis of trends in the sequences because reflexes for an agent may have the same response for percepts with the same meanings, e.g. if there is a bar-code tag in the range of a scanner, then it will perform the same READ action. So in such a sequence, a Condition-Action-Rule may apply that can be written as:

if Barcode tag is in range then read the tag
if Barcode is read then compare the tag

An SDA in the AOSF framework is modelled below:

function Smart-Device-Agent(percept) {
initialise actions, condition, condition-action-rules-set
current-state = *INTERPRET-INPUT(percept)*
rule = *RULE-MATCH(current-state, rules)*
action = *ACTION-RULE(rule)*
return }

Similarly, the User Agent, which is a software side agent, can be categorised as a utility-based agent (as shown in Fig-2), as they generate a high-quality behaviour to maximise the utility for the user. Where utility is a function that best describes the satisfaction level of the user when the user may have two or more goals to achieve, e.g. speed, accuracy or safety.

In this network, the connected resources (i.e. sensors or devices) and agents may vary as per the requirements of the environment and this increment or decrement in the network is unpredictable. A new user or device can be connected or disconnected regardless of time with no need to redefine the protocol for communication and negotiation. The User Agent, which is a specifically designed software agent, runs on a mobile device like laptop or PDA, and can make requests to an MA against a resource or functionality as per need. SDAs are responsible for providing one or more services (i.e., functional profiles, scanning or searching). This multi-agent based architecture allows SDAs to generate requests to MAs in order to support an autonomous configuration and adaptation to changes in the environment. SDAs are usually embedded within advanced smart devices (i.e. appliances with some in-built computational services and local memory storage capabilities) with employed agent planning and coordination strategies. The AOSF framework is based on Hierarchical Task Network (HTN) Planning [28]. MAS planning strategies are planned to be implemented within

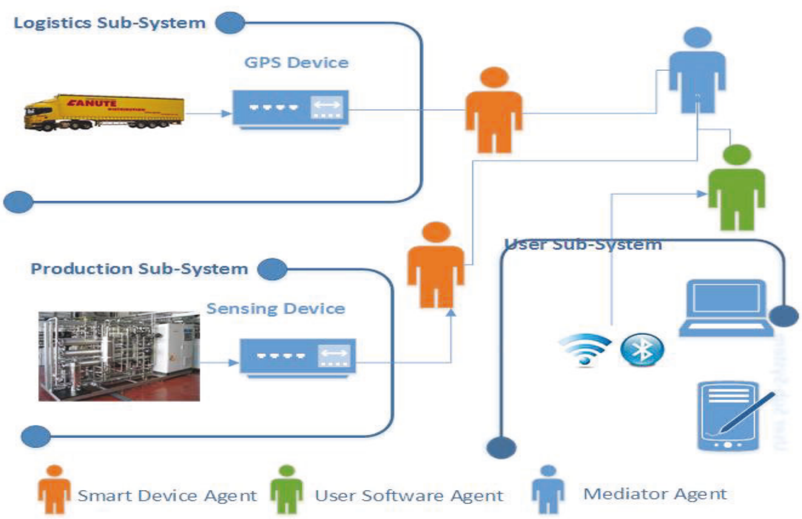


Fig. 2. AOSF Agents' Architecture

this framework in future e.g. PDDL [29] or IXTET-EXEC [30]. These strategies allow the provision of execution control, generation of plans, plan repair, and replanning strategies. SDAs stores device's current status and properties in a semantic way that is offered against the request triggered by other agents (i.e. other SDAs, MAs and UAs). SDAs semantically support elementary appliances connected to the system, in the case any agent generates a request, the request will be replied to after mediation. Conversely, if MA refers to standard SD properties, the request will be forwarded to the device seamlessly. The focus of this communication scheme is to configure a better possible situation in order to fulfil the request in the best semantic way.

3 Results and Future Work

A case of a company's distribution warehouse with constraints and limitations is applied to AOSR planner component of general AOSF framework. In contrast to a standard WMS, which provides a centralised management of tasks such as tracking location and level of products in the racks using a single logic, AOSR-Planner Agent (PA) uses a hybrid logic as per the products' characteristics to generate the placement plan. This plan is modified during runtime based on new parameters. AOSR does not rely on only one strategy; it provides a combination of different slotting and re-slotting strategies like zone logic [31], First In First Out (FIFO) [32], Put/Pick from the fewest [33], which make it hybrid in nature. After selecting the zone logic, the PA selects other suitable logic to store/sort products into the defined zone in accordance with the product specification and categorisation. AOSR planner passes through different states of the system, which are categorised as per the parameters sensed from the environment. For example preliminary states are normal initialisation states where the stocking is initiated assuming the available capacity for each product. The hybrid nature of logic selection in AOSR minimises the conflicts, hence in the initialisation states, no conflict arises, and the products are placed as per their defined racks, which are suggested by the PA. The proactive nature of AOSR, which makes it different than a standard WMS, helps to sense the upcoming conflict-states of the system. Conflict-states of the system are the states when the same parameters are sensed for a particular product, e.g. the advance shipment and delivery (AS/DN). In such a case, PA decides which products need to be re-slotted, as it can predict that more products are coming, so it re-slots the previous smaller quantity products having prior knowledge of shipment. This is how it reduces the issues of wandering items and overcrowded receiving/expedition areas (RA/EA).

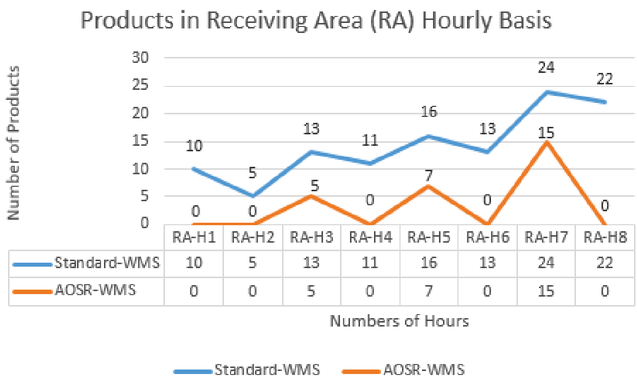
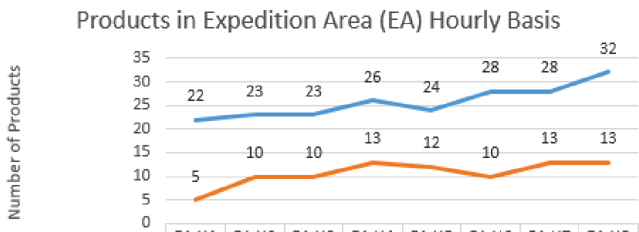


Fig. 3. Experimentational Results of Receiving Area using AOSR

Fig-3 represents the execution results between a standard WMS and AOSR-WMS. The graph shows the average number of products in RA on a particular day on an hourly basis. PA-algorithm generates a dynamic placement plan for the products to be placed into the exact racks based on the hybrid logic. keeping RA overloaded with products, increases the concerns of lost/wandering items and may leads towards stock imbalance [17]. The AOSR algorithm is designed to utilise its auto-inspection mechanism through features of RFID scanning and weight sensing to avoid such problems. Thus the products, in their certain packing units (i.e. case/pallets, box/cases), stay in RA just for identification and are then placed in the suitable racks as mentioned in the placement plan generated by the planner algorithm. Fig-3 shows a clear difference between the results of a standard WMS and AOSR-WMS. A closer look can explain that, in case of AOSR, the time span for first two hours, RA is entirely free and products are shifted to their exact places after consecutive intervals but in case of a standard WMS the RA becomes more and more congested with upcoming products as the time passes. During the time interval of H-8, the gap is quite apparent, representing a clear performance difference.

Through properly defined zone logic there are multiple EA's defined in a warehouse to accurately place the product and to identify the exact location even when they are in EA. Fig-4 demonstrates the results of AOSR to be better than a standard WMS, by reducing the total quantity of products in EA by nearly half, on an hourly basis. The AOSR algorithm is programmed to move the products in EA only when it cannot find a suitable space in the rack for a product in both cases: minimum possible and maximum possible available space. Only those products whose shipment date is near, are placed into EA, and so, very soon they are moved from EA to the shipping area, leaving the EA free for future possibilities. Thus the objective of maintaining a minimum number of products in EA is also achieved by AOSR-WMS. This is how the AOSR algorithm keeps EA less loaded so that the demarcation lines remain evident for the unobstructed movement of forklift trucks and floor staff within the shop floor.



	EA-H1	EA-H2	EA-H3	EA-H4	EA-H5	EA-H6	EA-H7	EA-H8
Standard-WMS	22	23	23	26	24	28	28	32
AOSR-WMS	5	10	10	13	12	10	13	13

Fig. 4. Experimentational Results of Expedition Area using AOSR

In future the AOSF framework and its associated AOSR-WMS algorithm are planned to be implemented with JaCaMo [34] with its associating environment and organization programming. It is expected that the AOSF framework will use an existing planner such as DOMAP [35] and IXTET-EXEC [30], to enhance the functionality and robustness of operations in SMEs. Handling tasks with the same priority can also be elegant future work in order to provide more flexibility in decision making for the user side. The implementation of Plant Side and multiple dimensions of User Side is also intentionally left for upcoming development in this particular project. The AOSF framework is planned to incorporate cloud architecture in the next phase of development in order to completely fulfil the idea of CPS.

References

1. P. M. Deane, *The first industrial revolution.* Cambridge University Press, 1979.
2. J. Mokyr, "The second industrial revolution, 1870-1914," *Storia dell'economia Mondiale*, pp. 219–45, 1998.
3. C. Freeman and F. Louçã, *As time goes by: from the industrial revolutions to the information revolution.* Oxford University Press, 2001.
4. M. Brettel, N. Friederichsen, M. Keller, and M. Rosenberg, "How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective," *International Journal of Mechanical, Industrial Science and Engineering*, vol. 8, no. 1, pp. 37–44, 2014.
5. S. Wang, J. Wan, D. Li, and C. Zhang, "Implementing smart factory of industrie 4.0: an outlook," *International Journal of Distributed Sensor Networks*, vol. 12, no. 1, p. 3159805, 2016.
6. J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, no. Supplement C, pp. 18 – 23, 2015.
7. W. He and L. Da Xu, "Integration of distributed enterprise applications: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 35–42, 2014.
8. A. A. Majeed and T. D. Rupasinghe, "Internet of things (iot) embedded future supply chains for industry 4.0: An assessment from an erp-based fashion apparel and footwear industry," *International Journal of Supply Chain Management*, vol. 6, no. 1, pp. 25–40, 2017.
9. G. Manogaran, C. Thota, D. Lopez, and R. Sundarasekar, "Big data security intelligence for healthcare industry 4.0," in *Cybersecurity for Industry 4.0*, pp. 103–126, Springer, 2017.
10. M. K. Adeyeri, K. Mpofu, and T. A. Olukorede, "Integration of agent technology into manufacturing enterprise: A review and platform for industry 4.0," in *Industrial Engineering and Operations Management (IEOM), 2015 International Conference on*, pp. 1–10, IEEE, 2015.
11. D. Ivanov, A. Dolgui, B. Sokolov, F. Werner, and M. Ivanova, "A dynamic model and an algorithm for short-term supply chain scheduling in the smart factory industry 4.0," *International Journal of Production Research*, vol. 54, no. 2, pp. 386–402, 2016.
12. S. Voss, H.-J. Sebastian, and J. Pahl, "Introduction to intelligent decision support and big data for logistics and supply chain management minitrack," 2017.
13. W. Shen, Q. Hao, H. J. Yoon, and D. H. Norrie, "Applications of agent-based systems in intelligent manufacturing: An updated review," *Advanced engineering INFORMATICS*, vol. 20, no. 4, pp. 415–431, 2006.
14. N. M. Sadeh, D. W. Hildum, and D. Kjenstad, "Agent-based e-supply chain decision support," *Journal of Organizational Computing and Electronic Commerce*, vol. 13, no. 3-4, pp. 225–241, 2003.
15. W. Shen, "Distributed manufacturing scheduling using intelligent agents," *IEEE intelligent systems*, vol. 17, no. 1, pp. 88–94, 2002.
16. W. Shen, "Genetic algorithms in agent-based manufacturing scheduling systems," *Integrated Computer-Aided Engineering*, vol. 9, no. 3, pp. 207–217, 2002.
17. G. Richards, *Warehouse management: a complete guide to improving efficiency and minimizing costs in the modern warehouse.* Kogan Page Publishers, 2017.
18. R. B. De Koster, A. L. Johnson, and D. Roy, "Warehouse design and management," 2017.

19. P. Centobelli, G. Converso, T. Murino, and L. Santillo, "Flow shop scheduling algorithm to optimize warehouse activities," *International Journal of Industrial Engineering Computations*, vol. 7, no. 1, pp. 49–66, 2016.
20. H. Ma, S. Su, D. Simon, and M. Fei, "Ensemble multi-objective biogeography-based optimization with application to automated warehouse scheduling," *Engineering Applications of Artificial Intelligence*, vol. 44, pp. 79–90, 2015.
21. R. Manzini, R. Accorsi, G. Baruffaldi, T. Cennerazzo, and M. Gamberi, "Travel time models for deep-lane unit-load autonomous vehicle storage and retrieval system (avs/rs)," *International Journal of Production Research*, vol. 54, no. 14, pp. 4286–4304, 2016.
22. M. Llonch, M. Bernardo, and P. Presas, "A case study of a simultaneous integration in an sme: implementation process and cost analysis," *International Journal of Quality & Reliability Management*, no. just-accepted, pp. 00–00, 2017.
23. R. Kishore, H. Zhang, and R. Ramesh, "Enterprise integration using the agent paradigm: foundations of multi-agent-based integrative business information systems," *Decision Support Systems*, vol. 42, no. 1, pp. 48 – 78, 2006.
24. F. Ud Din and S. Anwer, "Erp success and logistical performance indicators a critical view:," *International Journal of Computer Science Issues http://www.ijcsi.org/papers/IJCSI-10-6-1-223-229.pdf*, pp. 223–229, 2013.
25. M. Ruta, F. Scioscia, T. Di Noia, and E. Di Sciascio, "Reasoning in pervasive environments: An implementation of concept abduction with mobile oodbms," in *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on*, vol. 1, pp. 145–148, IEEE, 2009.
26. G. Loseto, F. Scioscia, M. Ruta, and E. Di Sciascio, "Semantic-based smart homes: a multi-agent approach.," in *WOA*, 2012.
27. S. Russell, P. Norvig, and A. Intelligence, "A modern approach," *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, vol. 25, p. 27, 1995.
28. L. Minglei, W. Hongwei, and Q. Chao, "A novel htn planning approach for handling disruption during plan execution," *Applied Intelligence*, vol. 46, no. 4, pp. 800–809, 2017.
29. V. Strobel and A. Kirsch, "Planning in the wild: modeling tools for pddl," in *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, pp. 273–284, Springer, 2014.
30. S. Lemai-Chenevier, "Ixtet-exec: planning, plan repair and execution control with time and resource management," *PhD thesis*, 2004.
31. D. Piasecki, "Warehouse management systems (wms)," *Inventory Operations Consulting LLC, http://www. inventoryops. com/warehouse_management_systems. htm*, 2005.
32. M. M. Jones, M. O. Juneja, K. Gnanamurthy, K. Kandikuppa, J. Y. W. Sheu, E. R. V. William, G. R. Hadagali, S. S. Rawat, V. Berry, D. Agrawal, *et al.*, "Consigned inventory management system," Mar. 31 2016. US Patent App. 14/499,372.
33. D. Preuveneers and Y. Berbers, "Modeling human actors in an intelligent automated warehouse," in *International Conference on Digital Human Modeling*, pp. 285–294, Springer, 2009.
34. C. JaCaMo Framework for Jason and Moise, "Jacamo framework for jason, cartago and moise." http://jacamo.sourceforge.net, 1 2017.
35. R. C. Cardoso and R. H. Bordini, "A distributed online multi-agent planning system," *Distributed and Multi-Agent Planning (DMAP-16)*, p. 15, 2016.

4.2.1 Erratum

The paper included in Section 4.2 states that JaCaMo [78] may be used as another tool to implement agent-oriented heuristics of AOSF/AOSR, which provides additional features of implementing environment programming and emotion handling. However, environment programming and emotion handling are not the focus areas of this research, hence, JADE is used as the main tool to perform all the experiments in this thesis. The design mechanisms provided by other available tools such as Jack [70] and Jadex [20] are also explored but the features provided by JADE (as detailed in Chapter 3) are much simpler and suitable for the AOSR strategy as compared to the aforementioned tools. Hence, all the algorithmic heuristics of AOSR have been implemented in JADE [81], which provides simplicity with the flexibility to design multiple agents and facility of sniffer agent interfaces to monitor the overall agents' activity.

4.3 Further Details on Interaction within AOSF-Agents

AOSF Agents are modelled on the basis of a typology of dependencies, which includes task dependencies, task−resource dependencies, and resource−resource dependencies. Coordination provides work-flow that deals with issues of task−task and task−resource dependencies. Furthermore, the need for multi-dimensional decision making requires the processes that underlie the coordination mechanism i.e. the notions of senders, receivers, messages, and language protocols. The pragmatics of the AOSF agents are based on Knowledge Query Manipulation Language (KQML) in compliance with FIPA-ACL (Agent Communication Language) [48]. To represent the interaction between the AOSF agents, a UML base interaction diagram is depicted in Figure 4.1. MAs are modelled to fulfil the responsibility of providing the semantic conditions for intelligent interaction among agents, against resources, in order to acquire available services. Top-down search [125] is employed for coordinating the Hierarchical Plans coming from users or devices. MAs are responsible for finding the best deal among the agents in the negotiation phase. For mediation roles, MAs are modelled to follow the sound algorithmic base of Pareto Optimal [90] and Contract Net Protocol (CNP) [198].

Clercq [35] employed the same strategy to find the best option among choices on specific criteria after receiving a service request. Agents in this systems are able to negotiate for available resources and services, to resolve conflicting situations between the agents and provide users with decisive information on the basis of in-built algorithmic intelligence to the agent architecture with respect to utility. This system's interaction is explicitly mentioned in the sequence/interaction model given in Figure 4.1 with an example of resource sharing and plan coordination. A User/Software Agent sets up its own plan based on a desired goal and then sends a request to a mediator agent (MA) with initial status and goals. MA holds the updated status and properties of all the SDAs included in the system by a simple pooling sequence of ping requests. On meeting the criteria of plan A sent by UA, MA sends an executable copy to SDA for execution. Otherwise, if criteria doesn't match successfully with plan A and SDAs current status, MA sends another executable plan B for approval and after that sends an approved, agreed and executable plan to SDA.

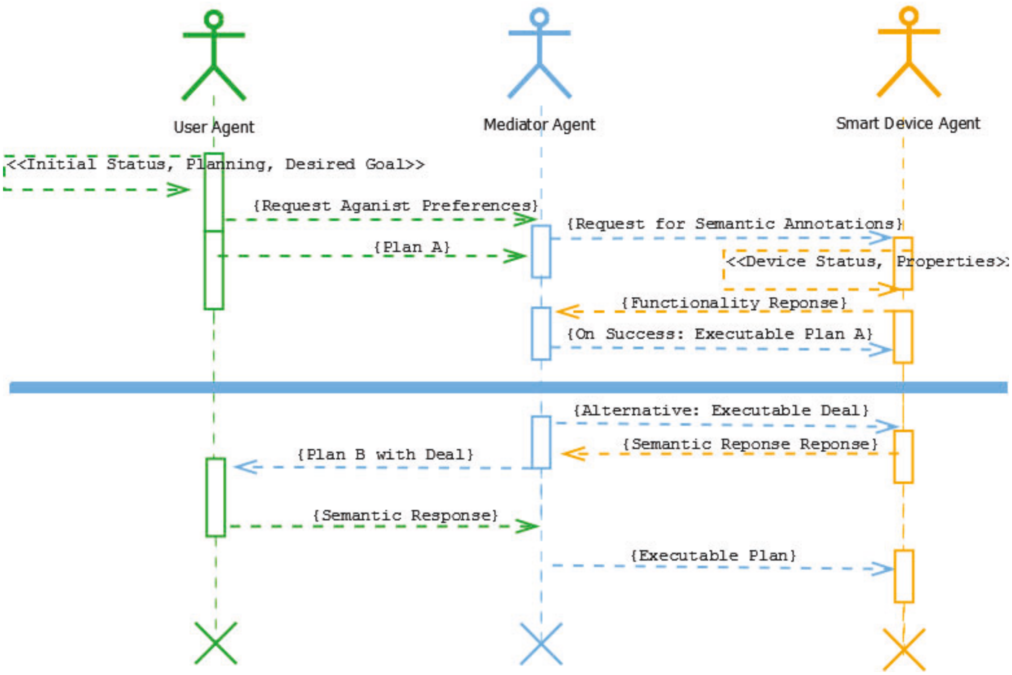


Figure 4.1: AOSF Agents' Interaction Model

For a logical implementation of the system, Unified Modelling Language (UML) is used to portray the systematic flow. Use Case Models are the logical schemes to represent different actors and procedures of the working system using multiple relationships like

<<*uses*>>, <<*extends* >> and generalisation. Figure 4.2 represents the Use Case Model for AOSR algorithm which shows that a floor supervisor, as a warehouse agent, can initiate the process of any operation. CRM and AOSR algorithm also behave as agents in the same platform but have their own container in implementation.

ASNs/ADNs represent the input and output of this system which is generated by CRM, which itself behaves as an agent in this environment. AOSR Agent is a Planner Agent that performs as the core agent for the system to initiate the procedures of shipping or receiving. Procedures for shipping and receiving the products (*Procedure-Ship* and *Procedure-Receive* in Figure 4.2) use the *Search-Placement* use-case in order to find the product in the warehouse whose placement has been generated.

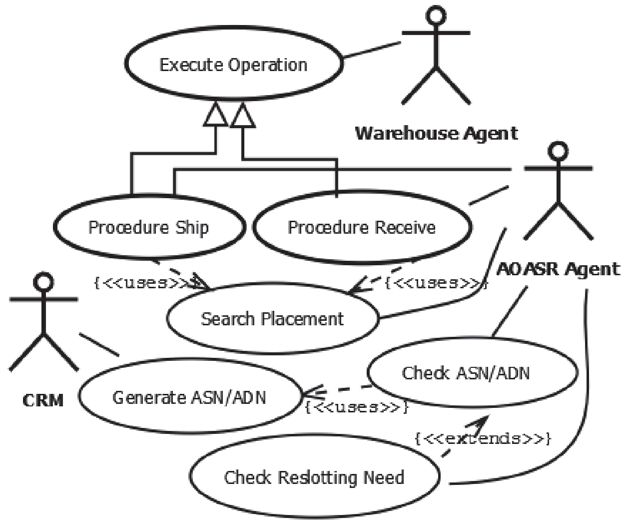


Figure 4.2: AOSF Agents' Interaction Model

In contrast to a standard WMS, which provides centralised management of tasks such as tracking location and level of products in the racks using a single logic, AOSR-Planner Agent (PA) uses a hybrid logic as per the products' characteristics to generate the placement plan. This plan is modified during runtime based on new parameters e.g. updated quantities or change in characteristics. AOSR does not rely on only one strategy; it provides a combination of different slotting and re-slotting strategies such as zone logic [147], First In First Out (FIFO) [86] and Put to/Pick from the fewest [150], which make it hybrid in nature. After selecting the zone logic, the PA selects another suitable logic to store/sort products into the defined zone in accordance with the product specification and categorisation. AOSR planner passes through different states of the system, which are categorised as per the parameters sensed from the environment. For example, preliminary states are normal initialisation states where it is initially assumed that the available capacity for each product can be stocked. The hybrid nature of logic selection in AOSR minimises the conflicts of quantity-space mismatch, hence in the initialisation states, no conflict arises, and the products are placed as per their defined racks, which are suggested by the PA.

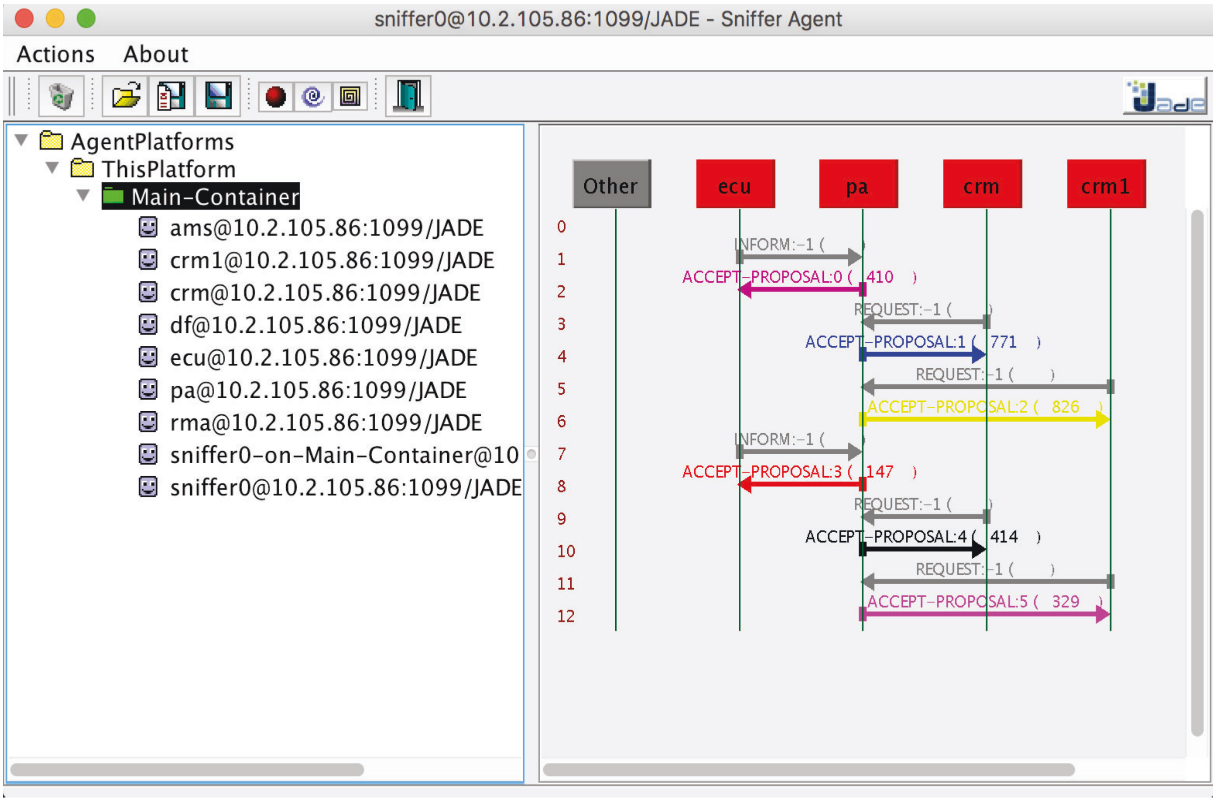


Figure 4.3: AOSF Agent Interaction in JADE Environment

The proactive nature of AOSR, which makes it different than a standard WMS, helps to sense the upcoming conflict-states of the system. Conflict-states of the system are the states where the same parameters are sensed for a particular product, e.g. for advanced shipment and delivery (AS/DN). In such a case, PA decides which products need to be re-slotted, as it can predict that more products are coming, so it re-slots the previous smaller quantity products having prior knowledge of shipment. This proactive nature of AOSR helps in reducing the issues of wandering items and overcrowded receiving/expedition areas (RA/EA).

Figure 4.3 represents the reflection of a Sniffer Agent which monitors the interaction between AOSF agents in Java Agent Development Environment (JADE). In this environment Agent Monitoring System (*ams*), Directory Facilitator (*df*) and Remote Monitoring Agents (*rma*) are the built-in agents for JADE. The main container includes three different categories of AOSF Agents:

1. Enterprise Central Unit (*ecu*);
2. Planner Agent (*pa*); and
3. Customer Relationship Management Agent (*crm*).

In the case presented above, CRM-agents have two instances with names of *crm* and *crm1*, in order to represent more than one customer location. For every scenario of product delivery to the warehouse *ecu* sends an advance shipment notice to *pa* (as shown in interaction 1 and interaction 7) and for very order request the *crm* agent sends an advance shipment notice to *pa* (as shown in interactions 3,5,9 and 11).

4.4 Enterprise Integration and AOSF

Based on Enterprise Integration (EI) concepts, the AOSF framework includes an internal supply chain to handle in-plant activities and an extended supply chain to coordinate between external entities. The concept of a virtual supply chain is also included in AOSF framework where structural elements (production units and transportation units) and control elements (demand, supply, process flows and inventory status) have been incorporated. Enterprises nowadays are more or less distributed, which calls for proper integration measures to be taken, because the loss in data-flow may lead towards high impact risks [65]. In order to keep the system running, enterprises sometimes run their contributing subsystems such as Manufacturing Execution System (MES), Customer Relationship Management (CRM) and Supply Chain Management (SCM) at different distributed servers. In such a scenario the strategies of Enterprise Application Integration (EAI) become necessary [178].

Integration in enterprises is discussed in two different dimensions: inter-enterprise and intra-enterprise integration [65]. Inter-enterprise integration, which includes the integration of two or more separate enterprises together, yields novel opportunities for ventures by offering cross-enterprise communication services [183]. Interoperability is the next concern while addressing the issue of inter-enterprise integration [111]. On the other hand, integration within the organisation, which is called intra-enterprise integration, is also important to keep the enterprise as one single updated unit. Intra-enterprise integration is further divided into two concentration tracks:

- Vertical Integration; and
- Horizontal Integration.

In order to bring an improvement in decision making, there must be a communication mechanism that supports a random plan repair strategy, i.e. updating the product placement plan on runtime, to increase performance efficiency, especially in SMEs [44].

Vertical Integration is the process to enhance the transparency of data flow and runtime status availability to a top-level application layer [183]. It connects different layers of information processing within a single unit of an enterprise, such as the flow of data from the physical layer to the connection/control layer, passing through the MES layer to update the interfaces of the ERP system. For example, if a customer wants to place an order for a particular product from his hand-held device, the application layer should provide real-time stock availability of the products through its control mechanism. When the order is placed, the MES and ERP sides should also be updated to reflect the integrated change.

Horizontal Integration is essential to provide the coordination mechanism between several entities of the same unit. Distributed organisations often-times need a strategy for coordination or maybe negotiation between several entities in order to properly utilise available resources [178]. Horizontal Integration may be further divided into the integration of physical hardware, a communication layer, a data processing layer and a business logic layer [77].

Another dimension of intra-enterprise integration is inter-departmental integration [92], also known as End-to-End/Unit-to-Unit Integration, which explores the coordination of different subsystems within an enterprise. In order to maintain the consistency of data-flow, all subsystems of an enterprise should receive the update regarding upcoming or undergone changes such as the reflection of actual stock value [40].

The comprehensive architecture AOSF caters for three of the aforementioned types of intra-enterprise integration as shown in Figure 4.4. In the AOSF framework, agents are distributed at different enterprise levels, whether it is the application layer or the physical device layer. As per the AOSF Agent categorisation [41], Smart Device Agents (SDAs) operate at the physical hardware layer, and User-side Agents (UAs) with Mediator

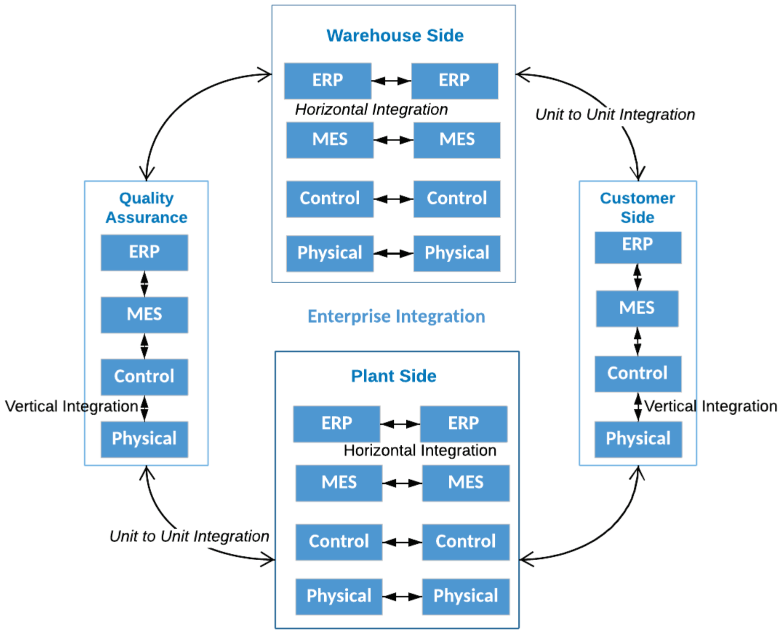


Figure 4.4: Enterprise Integration and AOSF

Agents (MAs) operate at Enterprise Application Layer. MAs provide a bridge for agents at different layer to cross-communicate for efficient resource utilisation. Input from UAs sets the preferences for the MES layer which passes through to the hardware level via Control Layer in order to execute a functionality. For example, to pass an instruction related to fewer or more production requirements from the application software layer to the physical hardware layer, the AOSF agents perform the following steps:

1. Keeping in view the stock value from warehouse planner agent and the upcoming orders, the CRM component suggests the increased or reduced production requirements to the admin user.
2. Based on user input from the application side, the instructions are transferred to the MES level and then pass through the control mechanism to the sensors, which need to be activated.
3. In order to accomplish the goal, the embedded smart device agents coordinate to check what tasks should be completed and stop when the goal is achieved.

Step *1* follows the concept of End-to-End/Unit-to-Unit Integration as the agents communicate in between different units, e.g. WMS and CRM. Step *2* follows the idea of Vertical Integration as it follows the sequence of top-down instructional flow from Application Layer to MES, then Control layer and ultimately to the Physical Hardware Layer.

Horizontal Integration is performed in Step *3* where SDAs and UAs interact with other same level agents to complete the goal together. The AOSF framework supports intra-enterprise integration with vertical and horizontal integration, but for inter-enterprise integration, the concerns of interoperability, trust, security and privacy are still open issues.

4.5 Summary

This chapter has provided an introduction to the AOSF framework and demonstrated how it works with the help of initial experimentation. Agent classification and communication strategies are also discussed in the context of overall supply chain architecture. This chapter also explained the three dimensional integration that the AOSF framework takes into account, utilising the recommended agent-oriented architecture.

The next step is to tie the connection of the general AOSF framework with the Agent Oriented Storage and Retrieval (AOSR) WMS strategy. For that a formal problem and domain definition is important. The next chapter details the problem and domain definition for AOSF.

Chapter 5

Problem and Domain Definition for AOSF

5.1 Introduction

Recalling from Chapter 1, one of the objectives of this research is to reduce the problems in SME-oriented warehouses. To meet this objective, this research presents the Agent Oriented Storage and Retrieval WMS system (AOSR-WMS), as highlighted in Chapter 4. Now, as we have discussed the AOSF framework in detail, including initial experimentation with test scenarios, we need to build an understanding of how the whole system of AOSR WMS can be modelled. One of the many ways to address the issue of modelling a flexible system is problem and domain definition [135]. This chapter focuses on the intermediate step of coming up with proper notations/grammar to work with, before actually building/developing the actual system.

Defining the problem and domain with a sound strategy helps in modelling complex systems [25]. For the purpose of providing a proper problem and domain definition for AOSR WMS, we have utilised the techniques of PDDL [135] and HTN [25] (highlighted in Chapter 3 and detailed in this chapter), which will leave us with a hierarchical structure to model the recommended hybrid logic of AOSR algorithm. This chapter provides adequate constructs of PDDL and HTN for implementing the AOSR algorithm efficiently.

The comprehensive architecture of agent-oriented end-to-end SC, presented by the AOSF framework in Chapter 4, provides a flexible layout for the implementation of a CPS-based enterprise integration network for SMEs, which meets the concern raised in RQ1 for incompatibility of SME-oriented SC with Industry 4.0. In order to provide a robust solution under this framework, the domain of Multi-Agent Planning (MAP), under MAS, is taken as an enabling technology, for the implementation of AOSR WMS. MAP provides the features of efficient communication and sensing from the environment, offering elegant negotiation mechanisms between the constituent elements. In order to find a solution for the second research question (RQ2) defined in Section 1.4:

- *Can MAS technologies provide Industry 4.0 benefits to SMEs? [95, 111, 183] What MAS solutions exist under the umbrella of Industry 4.0 to facilitate the standards?*

this thesis aims to explore the existing MAS systems under Industry 4.0 standard and to find the appropriate options where MAS technologies can support in developing a system to cater for the issues of warehousing in SMEs. There exist several agent-based systems in the literature, which explain the compatibility of agent technologies with high-tech manufacturing systems (as detailed in Chapter-3 [39, 116, 130, 190, 200]. The framework discussed in this thesis also incorporates agent-oriented technology to provide a robust, flexible and scalable system. In order to resolve the problem of developing a solution that may incorporate multiple software agents, their communication, negotiation and integration strategies, defining the problem and its domain is one of the preliminary steps. MAS technologies provide support not only for standard elements in the design of a scalable system such as in foundation architecture, communication, integration, supervision, but it also provides agent planning and plan-repairing strategies, which helps in laying out the foundation of a self-configurable system. Moreover, it also provides a mechanism for building a formal problem and domain definition. Thus, MAS technology is selected as a focus of this research as it provides all the required support and also the compatibility with the high tech standards such as Industry 4.0. For setting up the problem and domain definition of AOSF framework, this chapter includes Multi-Agent Hierarchical Task Networking (MA-HTN) planning formalism for efficient prototyping of the system. The remainder of this chapter is a copy of the following publication:

Ud Din F., Henskens F., Paul D., Wallis M., (2018) "Formalisation of Problem and Domain Definition for Agent-Oriented Smart Factory (AOSF)", in *IEEE Region 10 Symposium (TenSymp)*, IEEE, 2019, pp. 265-270

This publication includes details of how the possible actions to be taken by constituent agents can be further decomposed into smaller subtasks (primitive and non-primitive) to accomplish predefined goals such as maximising space availability and reducing the number of products in Expeditions Areas (EAs) and Receiving Areas (RAs). The conflicts and the dependencies of AOSF agents are also expressed in this chapter by using Backus-Naur-Form (BNF) [121]. BNF helps in identification of primitive and non-primitive tasks and their pre- and post-conditions, which are the necessary segments of domain and problem definition grammar. For example, the instance taken in the article below, related to domain definition, includes a receiving process, slotting process and check-reslotting-need process as non-primitive tasks because they can be further decomposed into subtasks, whereas reserving a slot, moving a product from one location to another and updating the placement plan are tasks which cannot be broken down in the system, and so are categorised as primitive tasks. For the problem definition, the facts (such as receiving ASN/ADN, in the taken instance in the article below), and initial/goal states are the important elements to be defined. These problem and domain definitions for AOSF agents help in developing the AOSR system efficiently with proper heuristics to cater for any uncertainties, such as pre-occupation of an available storage location or an undefined Stock Keeping Unit (SKU), for a particular product.

This article also includes a detailed example of a product receiving process and provides a complete HTN-tree as an explanation of decomposition of tasks in an attempt to achieve primitive tasks efficiently. The receive process in the included example is a non-primitive task, which is subdivided further to slot/store a product either into racks or into EA. This process also accounts for checking the condition for the product to be in RA, when a proper ASN is received. There are certain conditions for slotting/storing the product too, e.g. if it finds a suitable and available rack then it moves the products to the racks and updates the space availability, which itself is a non-primitive task and is further sub-divided into two tasks (explained pictorially in the article below). If there is no rack available for the product then it checks whether reslotting is necessary; in case reslotting is needed, it moves the existing products from racks to EA and places the product in hand to the racks, otherwise it places the product into EA and updates the plan.

5.2 Publication

Formalisation of Problem and Domain Definition for Agent Oriented Smart Factory (AOSF)

Fareed Ud Din
School of Electrical Engineering and Computing
University of Newcastle
NSW, Australia
fareed.uddin@uon.edu.au

Frans Henskens
School of Electrical Engineering and Computing
University of Newcastle
NSW, Australia
frans.henskens@newcastle.edu.au

David Paul
School of Science and Technology
University of New England
NSW, Australia
dpaul4@une.edu.au

Mark Wallis
School of Electrical Engineering and Computing
University of Newcastle
NSW, Australia
mark.wallis@newcastle.edu.au

***Abstract*—Industry 4.0 is revolutionising recent industrial setups. Literature has examined the idea of Smart Factory under the umbrella of Industry 4.0 extensively, but further research into the applicability of such frameworks for Small to Medium Size Enterprises (SMEs) is still required. To help address this issue, the Agent-Oriented Smart Factory (AOSF) framework focuses on a multi-agent architecture for end-to-end Supply Chain (SC) in SMEs. This paper presents a Cyber Physical System (CPS) based extension to the general AOSF framework and design heuristics for problem and domain definition of Agent Oriented Storage and Retrieval (AOSR) warehouse system using Multi-Agent Hierarchical Task Networking (MA-HTN) planning formalism.**

***Index Terms*—Smart Factory, Small to Medium Size Enterprises (SMEs), Agent-Oriented Storage and Retrieval system (AOSR), Agent-Oriented Smart Factory (AOSF), Warehouse Management System (WMS)**

I. Introduction

The industrial revolution, which began at the end of 18th century, is still in progression. On the basis of three previous technological revolutions in industry, the fourth one is an informational revolution, which is termed as Industry 4.0. It is defined in the literature as an integrated, adaptive, optimised, and service-oriented interoperable framework for automating manufacturing processes [1]. It was initially defined in 2011, and by now, many contributions have been made in this area of research [2]. The proliferation of research in the domain of Industry 4.0 has yielded many worthwhile answers [3] to questions on topics including Theories/Perspectives [4], Cyber Physical Systems (CPS) architecture [5], Enabling technologies for implementing Industry 4.0, such as Internet of Things (IoT) [6], Big Data [7], Multi-Agent Systems (MAS) [8] and their applications.

Efforts towards implementing Smart Factory from the Artificial Intelligence side is mostly delivered through MAS based solutions to provide distributed and autonomous systems to the manufacturing industry, including but not limited to enterprise integration, enterprise collaboration, process planning, scheduling and controlling of the shop floor [9]. Finding a universally acceptable solution is very difficult and even modelling agent architectures to resolve the complexity is not easy, particularly in the domain of enterprise information systems, where integration requires agility in order to meet current competitive demands [10]. Domain of manufacturing process planning is an NP-hard problem [11] because of time and probability escalation. To solve the specified problems of resource allocation and scheduling, different techniques are employed in existing literature such as Petri Nets, Genetic Algorithms and Neural Networks [11], [12]. Industry-specific implementations also exist in literature e.g. for the automotive industry [13] and for the petrochemical industry [14]. Although extensive research has been conducted to provide a complete autonomous system, none of the works have focused in depth on the implementation of agent-oriented technologies for Small to Medium Size Enterprises (SMEs) [15]. Even though the idea of industry 4.0 is transforming the manufacturing industry but recent research claims that industry 4.0 cannot be purely mapped to SMEs [16]–[20].

For an enterprise, supply chain (SC) is an end-to-end architecture, from the back-end suppliers towards the front side customers. It does not only cover the supply to a firm but also the in-plant operations within. For SC operations, warehouses serve as the real backbone for maintaining the whole value chain. [21]. A general distribution-warehouse includes many operations such as *i)* receiving products *ii)* storing them with identification of empty racks *iii)* searching the products in warehouse, and *iv)* picking the products from assigned placements considering more than one possible location for a particular product. Handling the warehouse operations is practically impossible without a Warehouse Management System (WMS). A WMS is a software application that supports day-to-day operations in a warehouse [22]. WMS programs enable

Distributed Computing Research Group, University of Newcastle, Australia

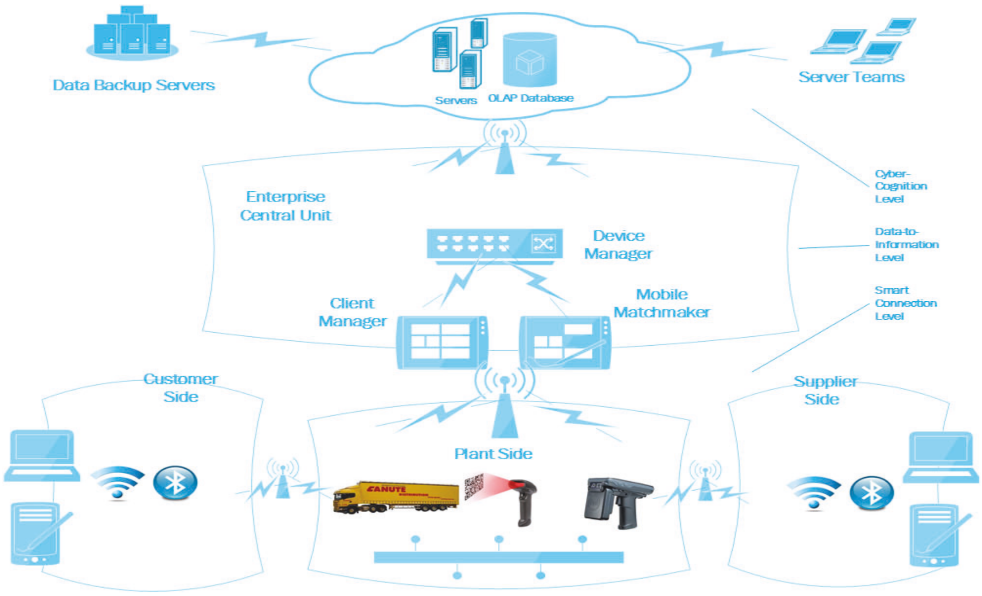


Fig. 1. AOSF Framework

centralised management of tasks such as tracking inventory levels and stock locations. Warehouse scheduling is a typical NP-hard problem, and is one of the most challenging types of combinatorial optimisation problems [22]. Although modern automated solutions for warehouse management also exist in literature such as the works mentioned in [23]–[26], where the automation is implemented through AS/RS (Automated Storage and Retrieval System through Robo-machines) to pick and ship the products using predefined trajectory and conveyor belts. But in case of SMEs, affording such a high-tech system is a question [27].

SMEs still face many issues, which are not solved by existing techniques, such as wandering items/picking lists [28], inaccurate current stock value at runtime [29], unmanaged receiving and expedition areas [30], unmanaged storage capacity [31] and inappropriate retrieval scheduling [32]. A generic solution is still needed to best fit the supply chain flexibility required in SMEs [33]. In supply chain research there are three different directions: *i)* agent-based supply chain frameworks, *ii)* agent-based supply chain simulations and *iii)* real-time application of agents in markets incorporating enterprise software applications in industry [34]. The Agent Oriented Smart Factory (AOSF) framework (introduced in our previous work [35]) not only provides a generic framework for overall supply chain but can also be used to realise efficient results in managing a warehouse for SMEs. This paper presents design description and guidelines for implementing a multi-agent system based solution for addressing warehouse issues in SMEs. For defining problem and domain design constructs of Agent Oriented Storage and Retrieval (AOSR) warehouse system, Multi-Agent Hierarchical Task Networking (MA-HTN) planning formalism is used. Section-II presents a CPS-based extension to a general SC layout of AOSF framework. Agent communication and negotiation strategies are well defined within the underlying architecture. Section-III presents problem and domain definitions for optimising multi-agent planning in warehouse implementation of the AOSF framework through AOSR system using MA-HTN formalism. Future research and development in this project is discussed in section-IV

II. AOSF Framework

The architecture of the Agent-Oriented Smart Factory (AOSF) [35], is based on end-to-end Supply Chain (SC) model (presented in our previous work [36]) and Cyber-Physical System (CPS) general framework [5]. AOSF framework is extended from an inbound supply chain, towards outbound supply chain, including an in-plant supply chain as well. An enterprise information system integrates all the other functional areas of a business into one central database. On the rear side it incorporates the supplier management system (SCM) to maintain in/outbound supply chain and on the front side, it facilitates customer relationship management (CRM).

The extended AOSF architecture, as depicted in Figure-1, presents the conceptualisation of AOSF framework into a CPS, which encapsulates the architecture into three generic layers. The plant side and front-end user side (customer/supplier side) make one layer, named the *Smart Connection Layer*. At this layer, all smart devices are implanted. Enterprise Central Unit, which takes care of overall integration and holds most of the data and transforms it to decisive information, is generally called *Data to Information Layer*. The *Cyber Cognition Layer* is the top layer where cloud architecture is maintained, and provides the overall cognitive abilities of the system.

The AOSF framework provides a mechanism of agent communication for SC entities to interact together; to bring robustness in operations. Design of formal semantics and axioms for AOSF-agents follows the concept of eight orthogonal ontological constructs proposed by [37], i.e. agent, role, goal, interaction, task, resource, information, and knowledge. Elegant mediation and negotiation mechanism between agents provide a sound technological basis which proves to be more efficient, less error-prone, and most importantly, modelled in a way to make it more flexible and adaptive to change. The AOSF framework makes a contribution in the domain of pervasive computing including aspects of the IoT paradigm. Further agent negotiation strategies for the AOSF framework are planned for the near future. To bring robustness in warehouse operations for SMEs, modelling of domain definition and problem formalism leads towards better multi-agent planning resolution.

III. Problem and Domain Definition

In the perspective of AOSF, the warehouse is considered as a homogeneous multi-agent planning domain. An accurate description of the domain and relevant problem to be solved, helps in finding a solution to the planning problem [38]. In the domain and problem definition of the Agent-Oriented Storage and Retrieval system (AOSR), we have used the constructs of MA-HTN, which is a multi-agent extension of hierarchical task networking (HTN) [39]. The HTN planning domain representation includes division and sub-division of tasks into sub-tasks to solve a specific problem easily. It provides a set of operators and methods. Operators are applied to primitive tasks on the basis of some preconditions, guaranteeing that some postcondition will be true afterwards, which eventually leads to a state transition of the system. Methods decompose a task into subtasks until they reach a primitive task where an operator can be readily applied. The HTN planning problem representation includes a list of atoms that are true during the initial state of the system and goals of the system. MA-HTN supports the *online* multi-agent planning, as the information during the runtime is also considered, and to represent the problem and domain in an agent planning environment. Problem and domain representations help to formulate a planning scenario.

Problem representation is defined by the information from the environment and domain representation is based on visible and accessible actions and goals of the system agents. Actions can cause *conflicts* and may have *dependencies*. Both the *conflicts* and *dependencies* are applicable for operators in MA-HTN. The notation used in MA-HTN is Backus-Naur-Form (bnf) [40]. A bnf specification is a set of derivation rules:

$$< symbol >::= \_\_expression\_\_$$

where, $< sysmbol >$ is a non terminal symbol and expression is a sequence of one or more terminal or non-terminal symbols. To support a choice between two expressions the symbol | is used, for example:

$$\_\_expression\_\_ \mid \_\_expression\_\_.$$

Some of the grammar rules to properly define the constructs provided by MA-HTN are mentioned below:

- each single quote pair that encloses a symbol is considered a string
- symbols enclosed by brackets are optional
- symbols preceded by $ are variables, that represent terminal symbols.
- Symbols with the * symbol, represent that zero or more instances are possible.
- Symbols with the + symbol, represent that one or more instances are possible.

TABLE I
MA-HTN bnf grammar for Domain representation

1	def-domain	::=	def-domain $AOSR
2	agent	::=	agent $PA
		::=	agent $SCM
		::=	agent $CRM
3	non-primitive-task	::=	! $receive ?p
		::=	! $slot-in-rack ?p ?r
		::=	! $update-availability ?r
		::=	! $check-reslotting-need ?r ?p2
4	primitive-task	::=	$reserve ?l
		::=	$move ?p ?l1 ?l2
		::=	$update-plan ?P
5	precondition	::=	$destination ?p ?l
		::=	not $at ?l | $at ?l
		::=	not $available ?l | $available ?l
		::=	not $reslotting-needed ?p ?l |
		::=	$reslotting-needed ?p ?l
6	conflict	::=	$ship $PA
7	dependency	::=	$receive ?p $SCM
		::=	$ship ?p $CRM

Table-I represents the domain definition in AOSR system. As per the constructs defined in MA-HTN the name of the domain is added dynamically by an agent is execution. The name of the agent is added in the domain description to identify, which domain the agent belongs to.

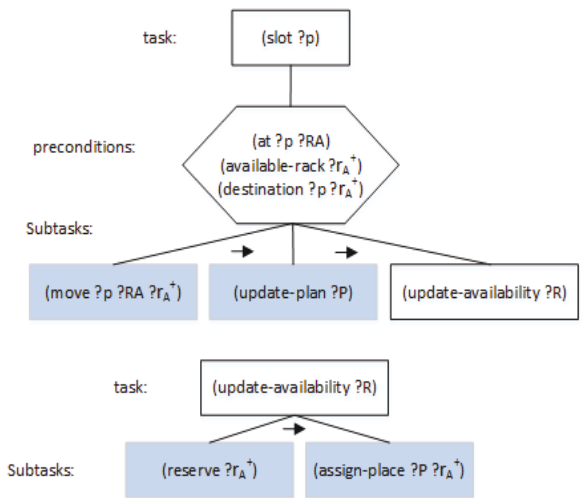


Fig. 2. HTN Slot Method

In MA-HTN each agent has its own domain and problem formulation, which supports the consideration of privacy factors. For the AOSR system, the planner agent (PA) coordinates with the supply chain management agent (SCM) and customer relationship management agent (CRM). Coordination between agents during execution keeps the whole system updated and integrated. Methods represent non-primitive tasks and operators represent primitive tasks.

TABLE II
MA-HTN BNF GRAMMAR FOR PROBLEM REPRESENTATION

1	def-problem	::=	def-problem $WMS
2	agent	::=	agent $PA
		::=	agent $SCM
		::=	agent $CRM
3	fact	::=	$ASN $pid,cid,quantity,sku,date
		::=	$ADN $pid,sid,quantity,sku,date
4	initial-state	::=	$\$s_0$ $r,ra,ea
5	goal-state	::=	$free-RA $p,ra,r,ea
		::=	$free-EA $p,ea,r

Conflict lists, which represent negative interactions and Dependency lists, which represent actions that need other agents to succeed, are also added. Despite applying constraints on AOSR system to avoid conflicts during runtime, unavailability of a certain products in stock may arise, which can be reduced by applying threshold alerts within the AOSR system.

The problem definition for the AOSR system is represented in table-II, where the availability of advance shipment or delivery notice (AS/DN) is considered as a fact for agents with the included parameters of product identification (pid), customer/supplier identification, to whom the products are shipping (cid/sid), quantity required, mutually agreed and predefined stock keeping unit (SKU) and the date for shipment or delivery. As AOSR computes and alters its placement plan during runtime, the initial states and goal are also change dynamically. To represent an initial state of the world for the planner agent there are different artifacts available, e.g. set of racks (r), receiving area (ra) and expedition area (ea). In order to manage the storage capacity efficiently, the AOSR planner algorithm is designed to prioritise the availability of receiving and expedition areas.

Figure-2 represents a simple slotting method of AOSR in MA-HTN formalism. The *slot method*, which is a non-primitive task, places a product *?p* into a suitable rack, having a precondition for the availability of that particular rack and is further dived into three subtasks. From the three subtasks, *move* and *update-plan* are primitive tasks and *update-availability* is a non-primitive task, which is further divided in two primitive tasks. Arrows represent the flow of the methods. Shaded boxes represent the primitive tasks.

Figure-3 depicts a complete plan for receiving a product and placing it at a proper place, whether it is a matching rack or an EA. The *method receive* makes sure that when the product is received it already has its advance delivery notice (ADN).

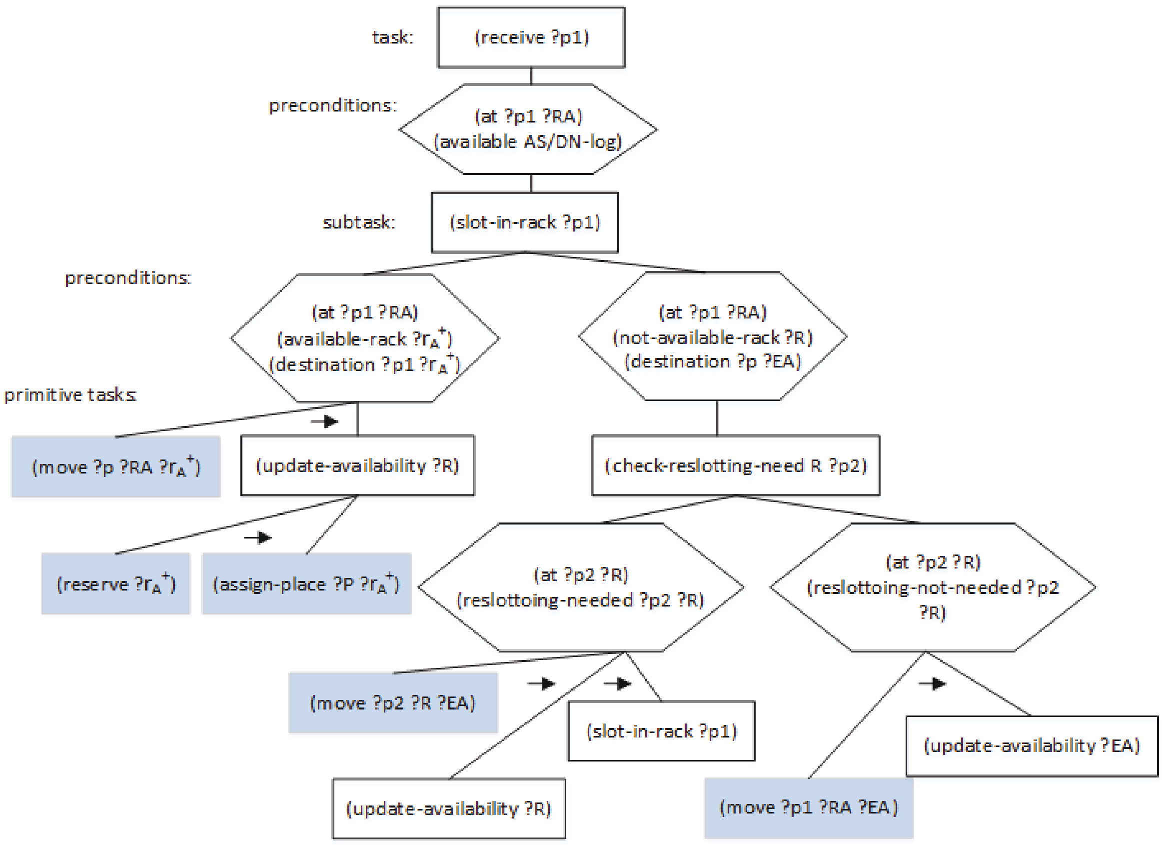


Fig. 3. HTN Receive Method

The priority of the AOSR planner algorithm is to place the products in the racks rather than to be in the EA. So, in the case of unavailability of a rack for product *?p1*, it checks if a similar product *?p2* could be re-slotted. If the precondition of re-slotting need is met, it re-slots the products and places the upcoming product into the rack fulfilling its FIFO logic. The parallel storing and sorting process reduces the problem of a piled EA which obstructs movement on the shop-floor.

Tools like CArtAgO [41] provide artifacts to parse the information of defined initial states into the dynamically generated agent's own fact list and initial state list. The name of the domain and problem can be collected through simple Java-based agent classes. The goal list of agents can be embedded to each agent if a static approach is used. For a dynamic approach, tools such as Moise [42] provide allocation of organisational goals to agents' personal goals. The methods may be parsed from an agent's plan library, and preconditions may be parsed from the context of the plan and tasks may be parsed from agent's body.

IV. Conclusion and Future work

AOSF framework includes an end-to-end integration of the whole enterprise, covering both the upstream and downstream operations of both supplier and customer sides. It presents the design constructs for the implementation of an AOSR system, which specifically focuses the warehouse side of SMEs. It tends to provide a flexible placement plan with moderate level storage and retrieval system excluding automated conveyor belts and robo-machines, which makes it affordable for SMEs. The outline of problem and domain definitions helps in formalising design parameters for a complex systems such as warehouse scheduling. MA-HTN planning formalism provides a comprehensive layout for meta-heuristics for modelling the homogeneous multi-agent planner algorithm for warehouses in SMEs.

In future the AOSF framework and its associated AOSR system are planned to be implemented with JaCaMo [43], using Jason [44] for agent development, CArtAgO [41] for Environment Programming and Moise [42] for developing Agent Organization. From the planning and plan repairing side AOSF framework is planned to go through the implementation of existing planners such as DOMAP [45], IXTET-EXEC [46] or Prottle strategies [47] to enhance the functionality and robustness of operations in SMEs. Handling multiple tasks with the same priority can also be elegant future work in order to provide more flexibility in decision making on the user side. Also, the implementation of Plant Side and multiple dimensions of User Side is also intentionally left for upcoming development in this particular project.

References

[1] Y. Lu, "Industry 4.0: A survey on technologies, applications and open research issues," *Journal of Industrial Information Integration*, 2017.

[2] Y. Liao, F. Deschamps, E. d. F. R. Loures, and L. F. P. Ramos, "Past, present and future of industry 4.0-a systematic literature review and research agenda proposal," *International Journal of Production Research*, vol. 55, no. 12, pp. 3609–3629, 2017.

[3] B. O. G. Ekren, A. Erkollar, "Research trends in industry 4.0: A content analysis," *International Symposium on Production Research*, vol. 55, no. 12, p. 13 15, 2017.

[4] S. Wang, J. Wan, D. Li, and C. Zhang, "Implementing smart factory of industrie 4.0: an outlook," *International Journal of Distributed Sensor Networks*, vol. 12, no. 1, p. 3159805, 2016.

[5] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, no. Supplement C, pp. 18 – 23, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S221384631400025X

[6] A. A. Majeed and T. D. Rupasinghe, "Internet of things (iot) embedded future supply chains for industry 4.0: An assessment from an erp-based fashion apparel and footwear industry," *International Journal of Supply Chain Management*, vol. 6, no. 1, pp. 25–40, 2017.

[7] G. Manogaran, C. Thota, D. Lopez, and R. Sundarasekar, "Big data security intelligence for healthcare industry 4.0," in *Cybersecurity for Industry 4.0*. Springer, 2017, pp. 103–126.

[8] M. K. Adeyeri, K. Mpofu, and T. A. Olukorede, "Integration of agent technology into manufacturing enterprise: A review and platform for industry 4.0," in *Industrial Engineering and Operations Management (IEOM), 2015 International Conference on*. IEEE, 2015, pp. 1–10.

[9] W. Shen, Q. Hao, H. J. Yoon, and D. H. Norrie, "Applications of agent-based systems in intelligent manufacturing: An updated review," *Advanced engineering INFORMATICS*, vol. 20, no. 4, pp. 415–431, 2006.

[10] M. Lützenberger, T. Küster, N. Masuch, and J. Fähndrich, "Multi-agent system in practice: when research meets reality," in *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2016, pp. 796–805.

[11] W. Shen, "Distributed manufacturing scheduling using intelligent agents," *IEEE intelligent systems*, vol. 17, no. 1, pp. 88–94, 2002.

[12] S. Weiming, "Genetic algorithms in agent-based manufacturing scheduling systems," *Integrated Computer-Aided Engineering*, vol. 9, no. 3, pp. 207–217, 2002.

[13] D. Frey, P.-O. Woelk, T. Stockheim, and R. Zimmermann, "Integrated multi-agent-based supply chain management," in *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on*. IEEE, 2003, pp. 24–29.

[14] N. Julka, R. Srinivasan, and I. Karimi, "Agent-based supply chain management—1: framework," *Computers & Chemical Engineering*, vol. 26, no. 12, pp. 1755–1769, 2002.

[15] N. M. Sadeh, D. W. Hildum, and D. Kjenstad, "Agent-based e-supply chain decision support," *Journal of Organizational Computing and Electronic Commerce*, vol. 13, no. 3-4, pp. 225–241, 2003.

[16] J. M. Mller, O. Buliga, and K.-I. Voigt, "Fortune favors the prepared: How smes approach business model innovations in industry 4.0," *Technological Forecasting and Social Change*, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0040162517312039

[17] M. Andulkar, D. T. Le, and U. Berger, "A multi-case study on industry 4.0 for smes in brandenburg, germany," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[18] C. Arnold, D. Kiel, and K.-I. Voigt, "How the industrial internet of things changes business models in different manufacturing industries," *International Journal of Innovation Management*, vol. 20, no. 08, p. 1640015, 2016.

[19] R. Masdefiol, M. del Mar, and F. Stävmo, "Industry 4.0–only designed to fit the german automotive industry: A multiple case study on the feasibility of industry 4.0 to swedish smes," 2016.

[20] A. Radziwon, A. Bilberg, M. Bogers, and E. S. Madsen, "The smart factory: exploring adaptive and flexible manufacturing solutions," *Procedia Engineering*, vol. 69, pp. 1184–1190, 2014.

[21] H. M. Abu Al-Rejal, J. D. Abu Doleh, L. M. Salhieh, Z. M. Udin, and S. Mohtar, "Barriers of supply chain management practices in manufacturing companies in republic of yemen: Pre-war perspective," *International Journal of Supply Chain Management*, vol. 6, no. 3, pp. 246–251, 2017.

[22] J.-P. Gagliardi, J. Renaud, and A. Ruiz, "Models for automated storage and retrieval systems: a literature review," *International Journal of Production Research*, vol. 50, no. 24, pp. 7110–7125, 2012.

[23] N. R. Jennings, "On agent-based software engineering," *Artificial intelligence*, vol. 117, no. 2, pp. 277–296, 2000.

[24] R. Manzini, R. Accorsi, G. Baruffaldi, T. Cennerazzo, and M. Gamberi, "Travel time models for deep-lane unit-load autonomous vehicle storage and retrieval system (avs/rs)," *International Journal of Production Research*, vol. 54, no. 14, pp. 4286–4304, 2016. [Online]. Available: https://doi.org/10.1080/00207543.2016.1144241

[25] H. Ma, S. Su, D. Simon, and M. Fei, "Ensemble multi-objective biogeography-based optimization with application to automated warehouse scheduling," *Engineering Applications of Artificial Intelligence*, vol. 44, pp. 79–90, 2015.

[26] P. Centobelli, G. Converso, T. Murino, and L. Santillo, "Flow shop scheduling algorithm to optimize warehouse activities," *International Journal of Industrial Engineering Computations*, vol. 7, no. 1, pp. 49–66, 2016.

[27] M. Llonch, M. Bernardo, and P. Presas, "A case study of a simultaneous integration in an sme: implementation process and cost analysis," *International Journal of Quality & Reliability Management*, no. just-accepted, pp. 00–00, 2017.

[28] J. Gu, M. Goetschalckx, and L. F. McGinnis, "Research on warehouse operation: A comprehensive review," *European Journal of Operational Research*, vol. 177, no. 1, pp. 1 – 21, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221706001056

[29] T. Poon, K. Choy, H. K. Chow, H. C. Lau, F. T. Chan, and K. Ho, "A rfid case-based logistics resource management system for managing order-picking operations in warehouses," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8277 – 8301, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S095741740800715X

[30] G. Richards, *Warehouse management: a complete guide to improving efficiency and minimizing costs in the modern warehouse*. Kogan Page Publishers, 2017.

[31] W. Lu, V. Giannikas, D. McFarlane, and J. Hyde, "The role of distributed intelligence in warehouse management systems," in *Service Orientation in Holonic and Multi-Agent Manufacturing and Robotics*. Springer, 2014, pp. 63–77.

[32] L. Li, *Supply chain management: Concepts, techniques and practices: Enhancing value through collaboration*. World Scientific Publishing Co Inc, 2007.

[33] A. Brintrup, Y. Wang, and A. Tiwari, "Supply networks as complex systems: a network-science-based characterization," *IEEE Systems Journal*, 2015.

[34] A. Brintrup, "Behaviour adaptation in the multi-agent, multi-objective and multi-role supply chain," *Computers in Industry*, vol. 61, no. 7, pp. 636–645, 2010.

[35] F. Ud Din, F. Henskens, D. Paul, and M. Wallis, "Agent-oriented smart factory (aosf): An mas based framework for smes under industry 4.0," in *KES Smart Innovation Systems and Technologies*, J. Jezic, J. Chen-Burger, R. Howlett, L. Jain, L. Vlacic, and R. Sperka, Eds. Australia: Springer, 2018, pp. Accepted 2018–03–06.

[36] F. Ud Din and S. Anwer, "Erp success and logistical performance indicators a critical view:," *International Journal of Computer Science Issues http://www.ijcsi.org/papers/IJCSI-10-6-1-223-229.pdf*, pp. 223–229, 2013.

[37] R. Kishore, H. Zhang, and R. Ramesh, "Enterprise integration using the agent paradigm: foundations of multi-agent-based integrative business information systems," *Decision Support Systems*, vol. 42, no. 1, pp. 48 – 78, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923604002155

[38] R. C. Cardoso and R. H. Bordini, "A multi-agent extension of hierarchical task network," in *10th Workshop-School on Agents, Environments, and Applications (WESAAC)*, 2016.

[39] D. S. Nau, T.-C. Au, O. Ilghami, U. Kuter, J. W. Murdock, D. Wu, and F. Yaman, "Shop2: An htn planning system," *Journal of artificial intelligence research*, vol. 20, pp. 379–404, 2003.

[40] D. D. McCracken and E. D. Reilly, "Backus-naur form (bnf)," 2003.

[41] CArtAgO, "Common artifact based architecture for agent open," http://cartago.sourceforge.net, 12 2017.

[42] Moise, "Moise organization programming for mas," http://moise.sourceforge.net, 12 2017.

[43] JaCaMo-Framework, "Jacamo framework for jason, cartago and moise," http://jacamo.sourceforge.net, 1 2017. [Online]. Available: http://jacamosourceforge.net

[44] Jason, "Java based interpreter for agentspeak," http://jason.sourceforge.net/wp/, 12 2017.

[45] R. C. Cardoso and R. H. Bordini, "A distributed online multi-agent planning system," *Distributed and Multi-Agent Planning (DMAP-16)*, p. 15, 2016.

[46] S. Lemai-Chenevier, "Ixtet-exec: planning, plan repair and execution control with time and resource management," *PhD thesis*, 2004.

[47] I. Little, D. Aberdeen, and T. Sylvie, "Prottle: A probabilistic temporal planner," 2005.

5.2.1 Erratum

The paper included in Section 5.2 states that JaCaMo may be used as another tool to implement agent-oriented heuristics of AOSF/AOSR, however, JADE is used as the main tool to perform all the experiments in this thesis for the reasons/details highlighted in Section 4.2.1.

Chapter 6

Agent-Oriented Storage and Retrieval-WMS Strategy

6.1 Introduction

Transforming intricate requirements, such as internal management of warehouses, entails prior specification of problem and domain definitions [25]. We have discussed this crucial aspect of problem and domain definition for the AOSF framework and its associated AOSR WMS system in the previous chapter, which provides the baseline details to stipulate the required aspects, such as conflicts, dependencies and conditions for prototyping the AOSR WMS strategy. Now, it becomes easier to build an understanding about design and flow of the whole AOSR system. In order to build a system that works well with the parent AOSF system, it is important to formulate the intercommunication mechanism and set out the flow of the system with proper conditions. This chapter provides details about how the subsystem of AOSR interacts with the central system of Enterprise Central Unit (ECU), which keeps all the components in the overall AOSF framework connected and integrated. It also includes the overview of algorithmic heuristics of the AOSR. Further details of how different algorithms interact with each other are detailed later in Chapter 8.

The details highlighted in Chapter 4, related to the AOSF framework under Industry 4.0, particularly for SMEs, and in Chapter 5, related to the details required for building a strategy for an associated WMS mechanism, satisfy the concerns raised in RQ2 and help provide an answer to RQ3:

- *How can issues of warehousing be resolved for SMEs using Industry 4.0? [60, 110, 149, 156] If SMEs cannot afford the high-tech robo-oriented warehousing system then how can the problems of warehouse management be resolved? Do the exist-*

ing standard WMS strategies meet the management requirement of SMEs-oriented warehouses?

This chapter explicitly addresses the concerns of RQ3 by including details of the AOSR-WMS strategy for providing a moderate level semi-autonomous system for SMEs under the umbrella of Industry 4.0. For the purpose of AOSR's integration with its parent framework of AOSF, initial experimentation is included in Chapter 4 and the prerequisites of its prototyping are detailed in Chapter 5. This chapter includes details of the *6-Feature* strategy recommended by AOSR, in comparison with standard warehousing strategies detailed in Section 2.2.1. It also includes details related to AOSR-WMS planner, how it interacts with the AOSF's general SC framework and the overview of its algorithmic heuristics. This chapter also presents some results from the implementation of this strategy to demonstrate its efficiency in comparison with the standard WMS approach. Details are presented in the following publication, which comprises the rest of this chapter:

Ud Din F., Henskens F., Paul D., Wallis M. and Hashmi M., "AOSR-WMS planner associated with AOSF framework for SMEs, under Industry 4.0," In review with *Cybernetics and Systems*, 2018.

This publication includes the hierarchical flow of the algorithmic heuristics, to generate a comprehensive product placement and retrieval plan while taking care of re-slotting needs within the warehouse. AOSR strategy recommends a generic but volatile warehouse design layout, which can be fixed initially and then modified as per business needs. One of the features that make AOSR distinctive, other than its hybrid logic selection, is its re-slotting approach, where the WMS-planner algorithm senses percepts from the environment to predict upcoming shipment/delivery request and performs the pick-place pair tasks simultaneously to make more space available within the warehouse.

6.2 Publication

AOSR: An Agent Oriented Storage and Retrieval WMS planner for SMEs, associated with AOSF framework, under Industry 4.0

Fareed Ud Din[a], Frans Henskens[a], David Paul[b], Mark Wallis[a] and Muhammad Adnan Hashmi[c]

[a]The University of Newcastle, NSW, Australia; [b]The University of New England, NSW, Australia; [c]The University of Lahore, Punjab, Pakistan

ABSTRACT

The concept of a smart factory, under Industry 4.0, relies heavily on Cyber-Physical-Systems (CPS) and Intra-enterprise-Wide-Networks (IWN). Cloud-based monitoring and supervision are incumbent to accomplish the promises of enterprise integration, automation, seamless information exchange, intelligent self-organisation and decentralised decision making. Extensive research has been conducted on the idea of Smart Factory under the umbrella of Industry 4.0, but there is still much research to be done from the perspective of such frameworks in Small to Medium Size Enterprises (SMEs). In order to address the issues of warehousing in SMEs, the Agent-Oriented Smart Factory (AOSF) framework provides a generic End-to-End Supply Chain (SC) model. This paper presents an Agent-Oriented Storage and Retrieval (AOSR) planner, under the umbrella of AOSF and Industry 4.0, that provides a smart plan to manage product placement and retrieval efficiently in a warehouse. This planner uses AI planning techniques, more specifically Hierarchical Task Network (HTN) planning to ensure different warehouse operations in a timely manner. This paper presents the 6-feature strategy of AOSR planner as well as the algorithmic heuristics. Initial experimental results are also presented that demonstrate the efficacy of the proposed approach in comparison to a standard warehouse management strategy.

KEYWORDS

Smart Factory, Small to Medium Size Enterprises (SMEs), Agent-Oriented Storage and Retrieval system (AOSR), Agent-Oriented Smart Factory (AOSF), Warehouse Management System (WMS), Enterprise Integration (EI), Supply Chain (SC).

Distributed Computing Research Group, The University of Newcastle, NSW, Australia

1. Introduction

The industrial revolution, which began at the end of the 18th century, is still in progression. The current era belongs to the information revolution, which is termed as Industry 4.0. Literature defines Industry 4.0 as an integrated, adaptive, optimised, and service-oriented interoperable framework for automating the manufacturing process Lu (2017). Industry 4.0 has been welcomed globally since its inception. Germany has included Industry 4.0 in its *High-Tech Strategy 2020 Plan* GATAI (2011). US Industry is following its *Advance Manufacturing Plan 2012* with revision by establishing *Industrial Internet Consortium (IIC)* Wide (2014). China and Taiwan have also announced their industrial automation strategy with the names of *China Manufacturing 2025* and *Taiwan Productivity 4.0*, respectively Lin, Shyu, and Ding (2017). South Korea has also launched its *Manufacturing Innovation 3.0* formula Kang et al. (2016). High investments are expected from Japan and Singapore as well Liao et al. (2017). As per the survey in PRWeb (2017) about the applicability and success of Industry 4.0 in large set-ups, 82% of the organisations said that they have implemented smart manufacturing and have experienced improved efficiency, 49% said they have experienced fewer defects rate, and 45% said that they have increased their customer satisfaction.

Recent studies reveal that Industry 4.0 mainly focuses on large enterprises Arnold, Kiel, and Voigt (2016) and only marginally on SMEs Schmidt et al. (2015). In fact, the literature claims that such an approach can possibly endanger the business model of SMEs Andulkar, Le, and Berger (2018). There are 3.6 million companies under the umbrella of SMEs in Germany alone, which employ 62.8% of the workforce Commission (2017). SMEs contribute nearly 57% of the annual GDP of Australia as per the report in NABaustralia (2018). Hence, for widespread implementation of Industry 4.0, its applicability in SMEs is a crucial aspect Masdefiol, del Mar, and Stävmo (2016). If the compatibility with semi-autonomous systems is not identified in a timely fashion, there is a chance that SMEs become victims rather than beneficiaries of Industry 4.0 Müller, Buliga, and Voigt (2018).

For an enterprise, supply chain (SC) is an end-to-end architecture, from the back-end suppliers towards the front-side customers. It does not only cover the supply to a firm but also the in-plant operations. For SC operations, warehouses serve as the real backbone for maintaining the whole value chain Abu Al-Rejal et al. (2017). Handling warehouse operations efficiently is not possible without a Warehouse Management System (WMS). WMS is a software application that supports day-to-day operations in a warehouse. Extensive research is conducted to provide complete autonomous systems, e.g. the solutions with robo-machines and auto-conveyor belts such as Flowshop Algorithm Centobelli et al. (2016) and EMBBO Ma et al. (2015). Even though the idea of Industry 4.0 is transforming the manufacturing industry, recent research claims that it cannot be purely mapped to SMEs Müller, Buliga, and Voigt (2018); Andulkar, Le, and Berger (2018). SMEs are still facing warehousing issues, such as wandering items/picking lists Gu, Goetschalckx, and McGinnis (2007); Business2Community (2018), inaccurate stock value at runtime Poon et al. (2009), unmanaged receiving and expedition areas Richards (2017), unmanaged storage capacity Lu et al. (2014) and inappropriate retrieval scheduling Li (2007).

Agent-Oriented Smart Factory (AOSF) Din et al. (2018a) provides an end-to-end supply chain (SC) model including inbound, in-plant and outbound supply chain principles, which are highlighted in our previous work Din and Anwer (2013). To help address the issues of warehousing in SMEs and to expose them towards the benefits of Industry 4.0, this paper presents an Agent-Oriented Storage and Retrieval (AOSR) strategy associated with its parent AOSF framework. The focus of this paper is to provide a strategy that can yield a smart plan for the placement and retrieval of products into/from the respective racks in a SMEs oriented warehouse using a hybrid logic rather than a static approach. The *6-Feature* WMS Strategy of AOSR provides the details about how a hybrid logic can help in reducing the overhead in Receiving and Expedition Areas (RA/EA) within a warehouse. Algorithmic heuristics of AOSR strategy are based on our previous work on the formalisation of problem and domain definitions using Multi-Agent Hierarchical Task Networking (MA-HTN) Din et al. (2018b). As a validation of this novel approach, a constrained test scenario is implemented, which yields substantial results in comparison to the standard warehouse management techniques Lu et al. (2014), Chen et al. (2013).

The rest of this paper is structured as follows. Section 2 highlights the general SC architecture of AOSF framework. Section 3 includes the *6-Feature* WMS Strategy of AOSR. Algorithmic heuristics are presented in section 4. Initial results, in comparison to the standard warehouse management techniques, are presented in section 5. Finally, section 6 provides conclusion and discusses some possible future directions.

2. AOSF Framework and AOSR Strategy

The AOSF framework Din et al. (2018a) provides a three-layered CPS-based end-to-end SC architecture as depicted in Figure-1. The bottom layer is the smart connection layer, where the plant side and front-end user side (customer/supplier side) with all the smart devices are implanted. The middle layer serves the purpose of data to information exchange, which includes an enterprise central unit to take care of overall integration and holds most of the data to transform it into decisive information. The top layer, where cloud architecture is maintained and provides overall cognitive abilities to the system, is called the cyber cognition layer. The AOSF framework provides a mechanism of agent communication for SC entities to interact together; to bring robustness to operations.

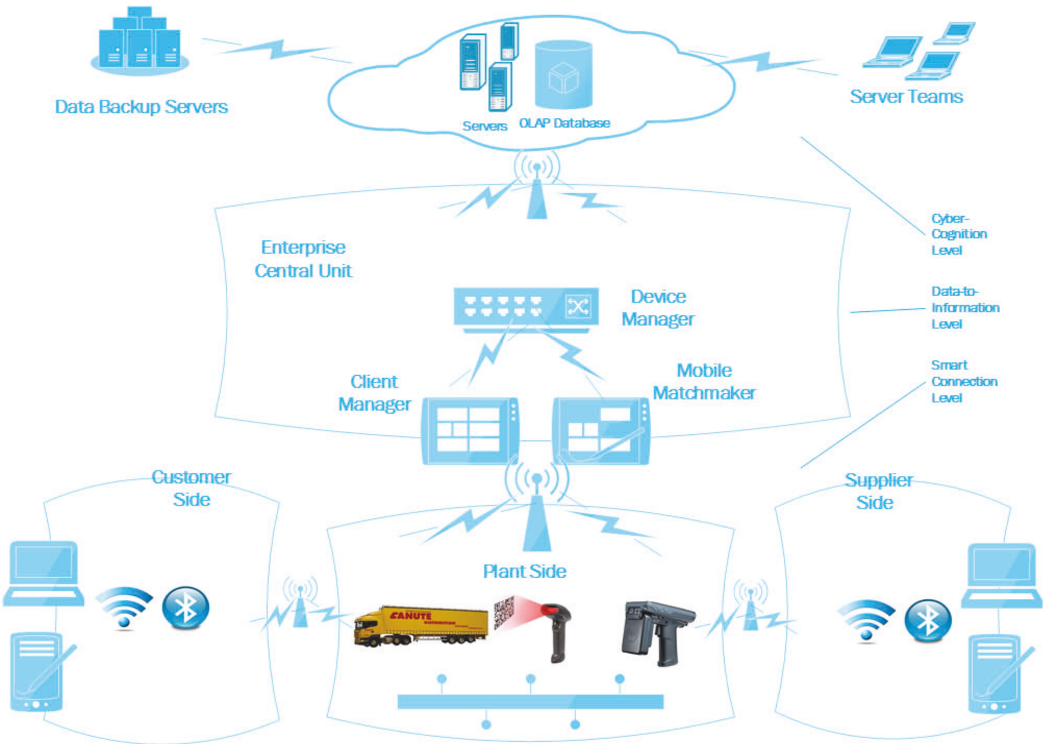


Figure 1.: AOSF Framework

The AOSF framework presents an associated Agent-Oriented Storage and Retrieval (AOSR) system, which provides moderate level SME-based warehouse management. It provides features of recording product characteristics such as exact dimensions, weight, unit of measure for stocking (each/box/case/pallets), max-stack-size, maximum quantity per location, hazard classification, whether it is a fast or slow mover and whether it is a finished good or unfinished. In order to pick/store the products into the racks, it provides a combination of different logics, including Zone Logic, First Come First Serve (FCFS) Jones et al. (2016) and Pick-From-Fewest/Put-To-Fewest Logic Preuveneers and Berbers (2009).

Managing warehouse operation is not an easy task. Each of the operations in a warehouse is itself complicated enough on its own Grosse, Glock, and Neumann (2017). Placing products and picking them with a proper strategy is a significant concern which may lead towards warehouse issues, such as wandering items/picking lists Gu, Goetschalckx, and McGinnis (2007), inaccurate current stock value at runtime Poon et al. (2009), unmanaged receiving and expedition areas Richards (2017), unmanaged storage capacity Lu et al. (2014) and inappropriate retrieval scheduling Li (2007).

Warehouse products may have different combinations of the characteristics which makes it difficult to store them in a particular order. The design of a warehouse and rack characteristics are important to efficiently store or retrieve products from their defined places as per their stock keeping units (SKUs) Koster, Johnson, and Roy (2017). In order to limit the issues occurring for the placement of products in the racks of a warehouse, there must be a sound product placement strategy in a Warehouse Management System (WMS). There are many propositions in literature in multiple dimensions such as Intelligent Products Giannikas et al. (2013), Ensemble Biography Based Algorithm Ma et al. (2015) and lean strategies Chen et al. (2013). Automated solutions for warehouse management also exist in literature, where the automation is implemented through AS/RS (Automated Storage and Retrieval System through Robo-machines) to pick and ship the products using predefined trajectories and conveyor belts Manzini et al. (2016). However, in the case of SMEs, the cost of such systems is often too expensive Llonch, Bernardo, and Presas (2017).

A standard WMS provides centralised management of tasks such as tracking the location and level of products in the racks Myerson (2012). Although WMS systems including features such as RFID and Voice Recognition Techtarget (2017) are also available, facilitating enterprises to manage stock, the issues of warehouse management still persist, such as receiving area overloading, demarcation lines vanishing, manual re-slotting and wandering/lost items Business2Community (2018). In order to limit such issues AOSR provides the *6-Feature* WMS planning strategy for receiving, storing and retrieving products in warehouse.

3. ***6-Feature*** **WMS Strategy of AOSR**

The AOSR *6-feature* planning strategy follows the WMS functionalities recommended by VDI (Association of German Engineers) VDI (2017) such as keeping a record of SKUs, inventory location, capacity and storage with respect to delivery/shipment notices. The features of AOSR-WMS planning strategy are mentioned as below:

- ***F1:*** The AOSR system provides an overall structure of the warehouse to alleviate the common issues occurring in a day-to-day warehouse environment. Despite great importance, zoning strategies have been given less attention than other issues Koster, Johnson, and Roy (2017). The AOSR recommends the subdivision of a warehouse into different volatile zones as per the categories of company's products, i.e., finished or unfinished goods, hazardous or non-hazardous materials, fast or slow moving goods. The AOSR recommends volatile zoning, in order to fulfil the supply and demand relationship. A reference warehouse architecture recommended by the AOSR strategy is presented in Figure-2. In the current case, an equal share is given to all the specified categories of products.

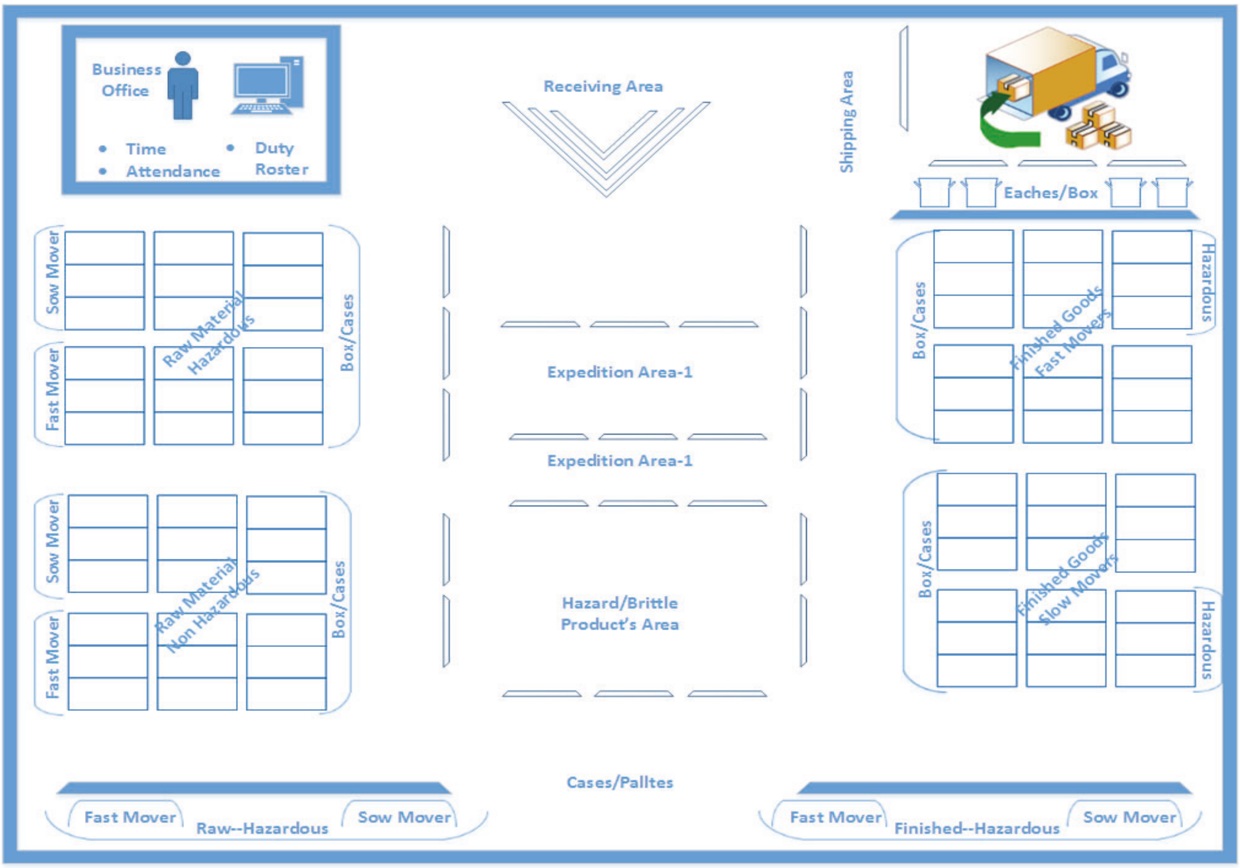


Figure 2.: An AOSR reference warehouse structure

- ***F2:*** Each zone has racks in it, that can be further divided into different levels. The number of racks and levels are flexible and can be configured initially before launching the setup. As a constraint for experimental purpose, all the racks are divided into 3 levels with each level containing space for 5 SKUs, yielding a total of 15 SKUs in a rack. This implies that, for a minimal setup, it provides the storage capacity for around 2 thousand products with the flexibility to support 20 thousand products to be scheduled in a single routine day in a ten times larger scenario.
- ***F3:*** SKUs as well as RFID implantation may be mutually settled between the parties (vendors and the case company) to maintain consistency. In order to keep it simple and standard, SKUs are constrained to **Each/Box, Box/Cases and Case/Pallets**.
- ***F4:*** Product placement strategy is another important aspect in warehouse management

Giannikas et al. (2013). The AOSR mechanism does not rely only on one strategy, it provides a combination of different slotting and re-slotting strategies including zone logic Piasecki (2005), First Come First Serve (FCFS) Jones et al. (2016) and Put/Pick from fewest Preuveneers and Berbers (2009), which makes it hybrid in nature. After selecting the zone, the Planner Agent (PA) selects another logic to store/sort products into the defined zone in accordance with the product specification and categorisation.

- ***F5:*** Another critical feature that makes AOSR suitable for Industry 4.0 is its integration with CPS based general SC architecture. The Planner Agent (PA) in AOSR coordinates with the Enterprise Central Unit (ECU) to communicate with software-side user agents (which may be the CRM Agent or SCM Agent) to sense any upcoming order. Figure-3 reflects the interaction of AOSR Planner Agent (PA) with ECU and CRM/SCM Agents. PA receives Advance Shipment Notices (ASNs) from ECU side and and Advance Delivery Notices (ADNs) from CRM/SCM side as an input. For every upcoming operation it initiates different processes and updates the stock with corresponding features and coordination mechanism.

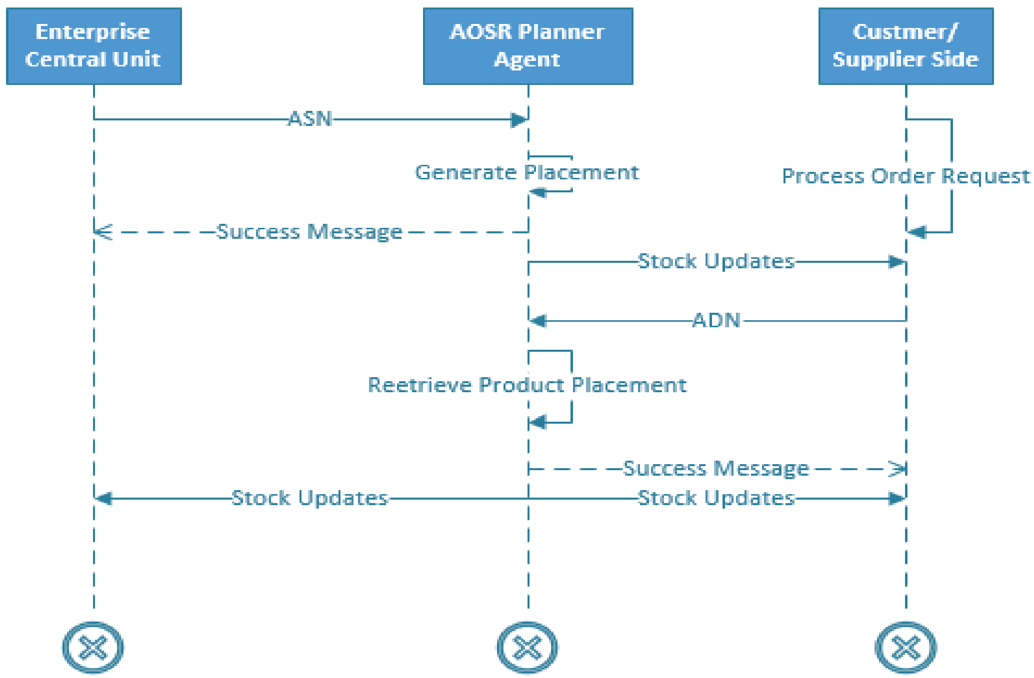


Figure 3.: Interaction of AOSR Planner Agent (PA)

- ***F6:*** In order to make AOSR proactive, Business Process Re-Engineering (BPR) strategy is utilised where it requires delivery and shipment notices prior to the execution day. If it is not mutually set between the parties (vendor and the company), the overhead may be increased as the system would be performing a reactive approach. The threshold for this prior notice is flexible and can initially be defined by the company and later modified. The set of ASNs and ADNs, which varies with time, provides AOSR with heuristics to check for any re-slotting needs. A comprehensive product placement plan, that keeps on updated with the time, is the output of this system.

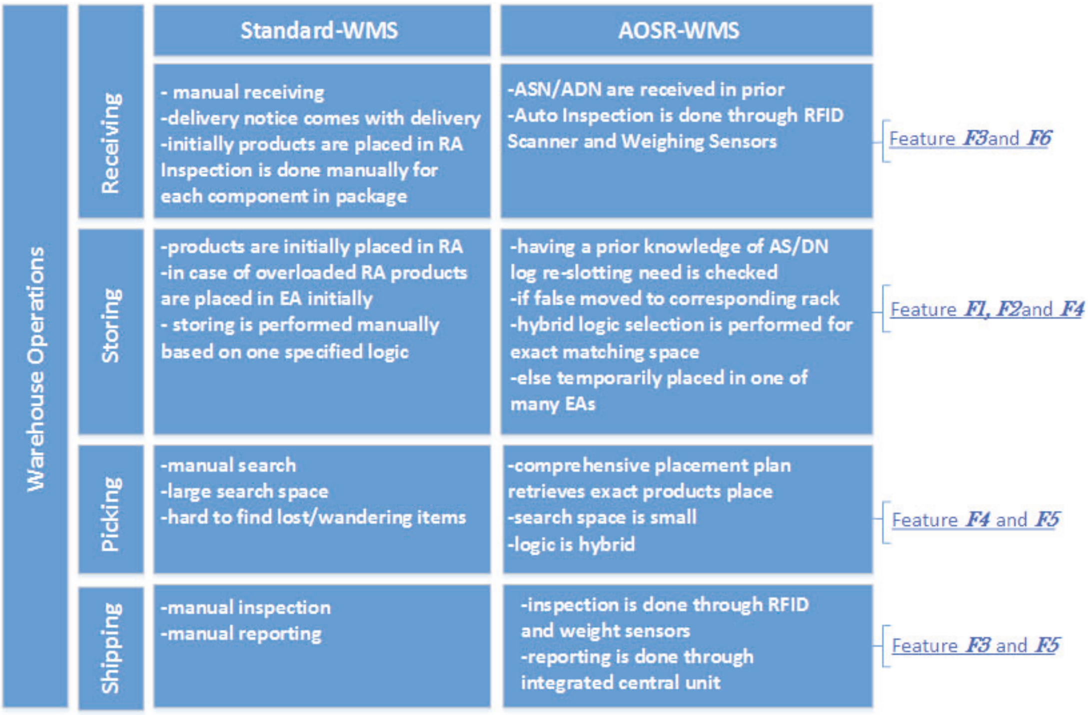


Figure 4.: Comparing a standard WMS based on Lu et al. (2014), Chen et al. (2013) and AOSR

The feature ***F1*** defines the major product placement area and then ***F2*** provides more information for exact placement location. One of the reasons, for not having a proper place for certain packing of a product is not having the predefined SKUs, which is constrained by ***F3***. ***F4*** makes AOSR a hybrid WMS strategy, ***F5*** shapes AOSR as a CPS-based system, which makes it suitable for Industry 4.0 and ***F6*** makes it proactive to resolve the problem of lost and wandering picking list and overcrowded Receiving Area (RA). This is how the highlighted issues of warehouses in SMEs may be reduced by employing AOSR strategy along with the general AOSF framework. It does not provide a fully automated solution with robo-machines and autonomous conveyor belts but it focuses on the problems of SMEs, where affording a high-tech solution is itself a concern. Figure-4 reflects the comparison of the AOSR with a standard WMS strategy.

4. AOSR Algorithmic Heuristics

The AOSR algorithm provides a solution to the problems of scheduling products and their slotting and re-slotting by giving a comprehensive placement plan based on descriptive initial states of the system having a particular set of actions. It takes ASN/ADN as an input (necessarily to be received prior to the receiving of products). A set of products and a set of racks with their characteristics are also provided as input. Based on these inputs the AOSR planner algorithm provides a comprehensive plan as an output. Figure-5 represents its abstract level architecture.

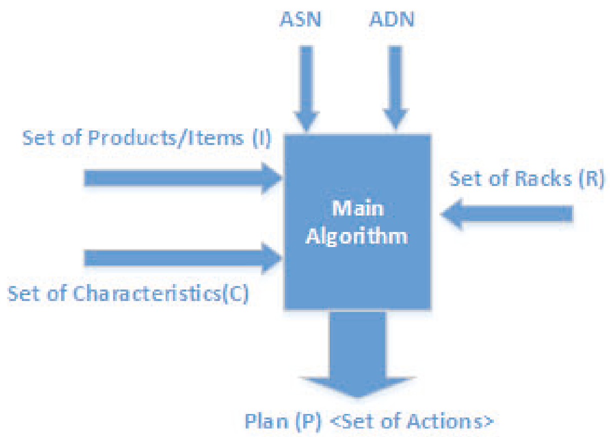


Figure 5.: AOSR Algorithm

The comprehensive plan **P** is defined as a set of time-stamped actions (TAs) which can be either to pick or to place a product in/from a rack or to search a suitable rack or product with associated ASN/ADN as represented in equation-1. **TA** is set of actions with respect to time and date in order to be identified uniquely as represented in equation-2 and 3.

$$\mathbf{P} = \{TA_1, TA_2, TA_3, ..., TA_{\mathrm{n}}\} \tag{1}$$

$$\mathbf{TA} =< A, DateTime > \tag{2}$$

$$\mathbf{A} =< move(p, src, dst) > \tag{3}$$

In equation-1, n is the possible number of time stamped actions. In equation-3 p is any product to be considered for shipping or receiving with quantity as specified in ASN/ADN, *src* is the source from where the items are to be picked and *dst* is the destination. Figure-6 represents the detailed algorithmic flow of the AOSR algorithm. Initially different zones are defined as per the characteristics of racks in the warehouse, and the initial state of the warehouse. In a warehouse, there is a series of Advance Shipment Notices (ASNs) and Advance Delivery Notices (ADNs), according to which the operations store, retrieve, place and pick/put-away are invoked.

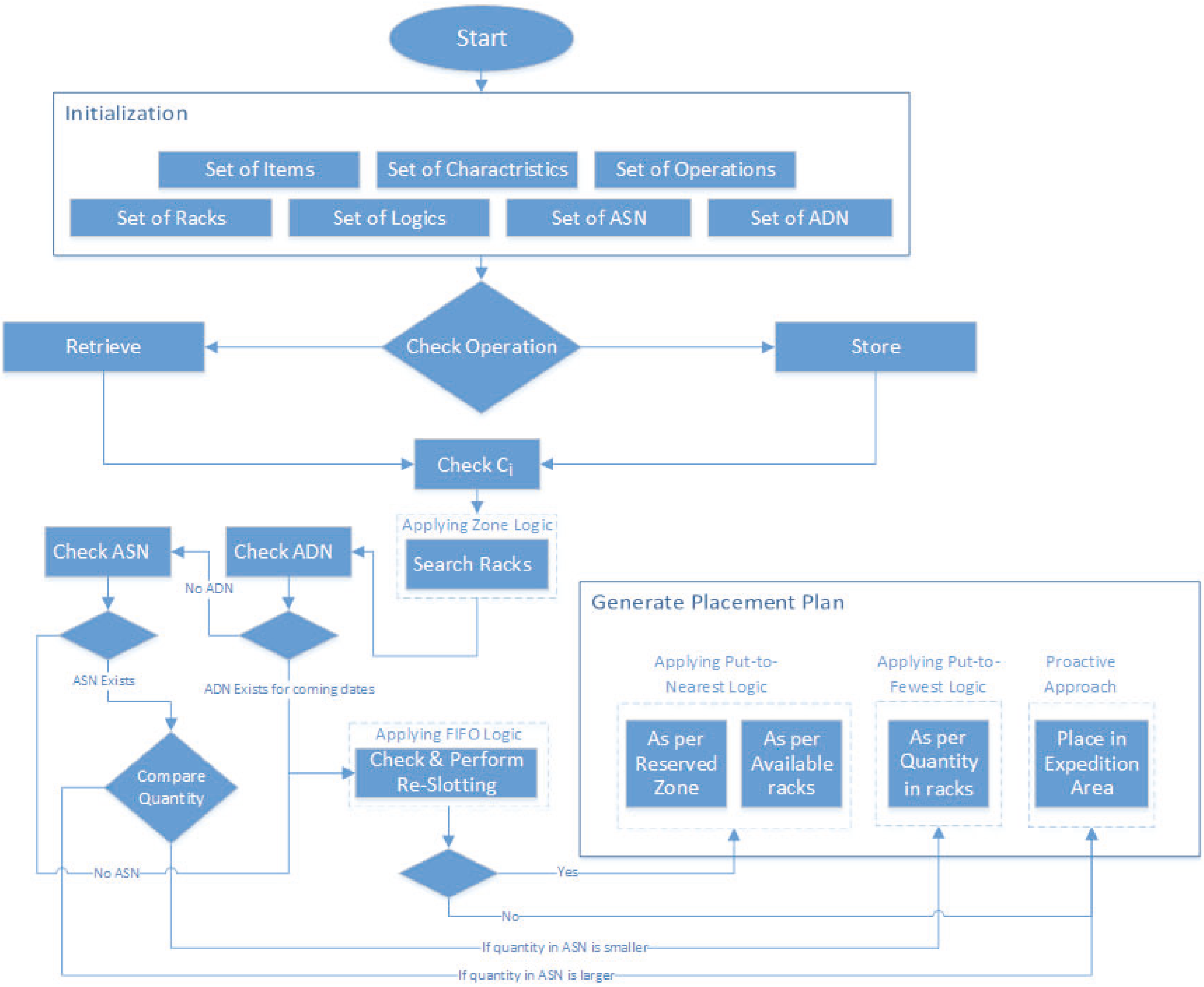


Figure 6.: Algorithmic Flow of AOSR-WMS

Incorporation of multiple agents in this framework enhances the suitability for integration with industry 4.0 in a CPS-based general SC architecture. Mediating and negotiating strategies are inbuilt to the AOSR planner agent to interact with the different classification of the constituent agents. Business process re-engineering (BPR) makes AOSR proactive, as it requires delivery and shipment notices prior to the execution day in order to generate a placement plan. AOSR employs its heuristics to check for any re-slotting need to cater to the problem of lost and wandering picking list and overcrowded Receiving Area (RA).

Algorithm-1 represents the overview of AOSR planner algorithm for the process of receiving and finding an appropriate location for upcoming products. In a standard warehouse, the products are placed into RA or EA, which creates overloading and causes problems afterwards for placing the products into their respective racks Sande (2017). The AOSR algorithm automatically searches for an appropriate rack-slot with matching characteristics and then places the product into that specific space, if there is no available slot then it checks if an item needs to be shipped from the racks in near future. If so, it re-slots that product to the expedition area and places the newly arriving product into the rack. This is how the AOSR algorithm reduces the complexity of handling products in EA for a long time and efficiently utilises the spaces in the racks for a longer time frame, consequently reducing the hassles on the shop floor.

Algorithm 1: Overview of *procedure_store* in AOSR Planner

```
procedure_store (P, ASN, ADN, Racks)
for each ASN loop
  if true (availableRack ← search_rack (Racks, ADN))
      P ← P ∪ move(product_in_adn, RecArea, availableRack);
      break ;
  else if  true ( search_expeditionArea (ExpArea, ADN))
          //storing and sorting are performed in parallel//
          if (true (check_reslottingNeed (ASN, ADN, Racks))
            P ← P ∪ move(product_in_rack, rack, ExpArea);
            P ← P ∪ move(product_in_adn, RecArea, rack);
        else
            P ← P ∪ move(product_in_adn, RecvArea, ExpArea);
    end if
end loop
s ← updated state
```

Finding a proper place for a particular product is a trivial question that can be resolved as soon as the indication for a product to be received is sensed by the system. Searching a rack is itself a process as it checks all the available racks for the product quantity and finds the minimum possible space for products to fit in, in order to utilise the maximum space-availability to incorporate upcoming products. When all the characteristics of products are matched with that of a rack, then an available rack is checked for possible space, and if space is matched with the quantity of the product, then this available rack is assigned to that particular product. The function *search_rack()* helps to optimise the space by finding minimum slot adjustment within the racks based on put-to-the-fewest logic Preuveneers and Berbers (2009). The logic of put-to-the-nearest Piasecki (2005) is also embed with in the search available rack as it selects the nearest rack from list of available racks in a particular case.

If a suitable rack with matching space is found, the product is placed there in the given quantity otherwise this algorithm attempts to find alternative space by checking the re-slotting need with the support of function *check_re-slotting_need()* or, in the worst case, placing the products in EA after checking for availability through the function *search_expeditionArea()*. For placing and picking a specific product, the procedure *move()* is invoked, which generates a new placement for the product to be stored, after updating the quantity level in the previous location.

The products may be moved from RA to Rack, Rack to Shipment Area, RA to Expedition Area, EA to Rack or Rack to EA. In all the move cases, the PA checks for space availability. Similar to *procedure_store()*, the AOSR system includes *procedure_retrieve()* as well. Having an efficient placement plan generated by *procedure_store()* it becomes easier to retrieve a product from the smart plan yielded by the system. Whenever the function *procedure_retrieve()* is invoked it analyses the system's current state in order to find the exact place of a product either in the racks or in the expedition area. This algorithm provides a solution for checking both ends and fulfils the demand. It explicitly invokes the *move()* function and embeds it to the general Plan of the warehouse activities.

5. Results and Discussion

A test scenario of a distribution warehouse with constraints and limitations has been applied to the AOSR planner component. In contrast to a standard WMS, which provides centralised management of tasks such as tracking location and level of products in the racks using a single logic Myerson (2012), the AOSR-Planner Agent (PA), based on its hybrid nature, firstly applies the zone logic on receiving the product and then selects another suitable logic to store/sort products into the defined zone in accordance with the product's specifications and categorisation. The AOSR planner passes through different states of the system, which are categorised as per the parameters sensed from the environment. The test cases applied to the AOSR with products' characteristics and their placement scheme are represented in Figure-7, where different states of the system are denoted by ***S1-S4***. There are different cases in every state of the system e.g. state ***S1*** includes ten different cases represented by *C1.1-1.10* and state ***S2*** includes four cases represented by *C2.1-2.4*. For every case the details about products' characteristics are mentioned. Based on these details the PA generates the placement locations, represented in the last column, which is a unique identification for the location of every product. State ***S1*** and state ***S2*** are normal initialisation states. The hybrid nature of logic selection in AOSR minimizes conflicts, hence the products are placed as per their defined racks, which are suggested by the Planner Agent (PA).

Represents System's first state: S1

Represents System's second state: S2

Represents System's third state: S3

Represents System's fourth state: S4

Represents System's fifth state: S5

T. ID	Item ID	S/R ID	Dimension	Weight	Unit	Hazard	Fast/ Slow	Finished /Raw	Q	Date	Placement
C1.1	P-9001	S-001	24x36"	10kg	Eaches /box	0	0	0	20	01/10 /17	E/B00011 E/B00012 E/B00013 E/B00014
C1.2	C-3921	S-002	40x64"	20kg	Box/ case	0	1	0	10	01/10 /17	B/C01011 B/C01021
C1.3	k-9803	S-003	20x32"	15kg	Box/ Case	1	0	0	10	01/10 /17	B/C10011 B/C10021
C1.4	R-3392	S-004	120x200"	30kg	Cases/ Pallets	0	0	1	10	01/10 /17	C/P00111 C/P00121
	An ADN received for P-9002 on 02/10/17 with Quantity 25										
C1.5	P-9002	S-005	24x36"	20kg	Box/ cases	1	1	1	25	02/10 /17	B/C11111 B/C11121 B/C11131 B/C11112 B/C11122
C1.6	k-9804	S-006	20x32"	15kg	Box/ Cases	0	1	1	20	02/10 /17	B/C01111 B/C01121 B/C01131 B/C01112
C1.7	R-1292	S-007	120x20"	40kg	Cases/ Pallets	1	1	1	15	03/10 /17	C/P11111 C/P11121 C/P11131
C1.8	K-3269	S-008	98x160"	45kg	Case/ Pallets	1	1	0	20	03/10 /17	C/P11011 C/P11021 C/P11031
C1.9	F-9210	S-009	20x32"	30kg	Box/ Cases	1	0	0	30	03/10 /17	B/C00011 B/C00021 B/C00031 B/C00012
C1.10	L-3092	S-010	24x36"	35kg	Box/ Cases	1	0	1	30	03/10 /17	B/C10111 B/C10121 B/C10131 B/C10112 B/C10122 B/C10132
	An ASN is received for R-1292 on 04/10/17 of Quantity 3										
	An ASN is received for K-3269 on 04/10/17 of Quantity 10										
	An ASN is received for F-9210 on 04/10/17 of Quantity 13										
C2.1	P-9002	R-001	24x36"	20kg	Box/ Cases	1	1	1	25	03/10 /17	B/C11111 B/C11121 B/C11131 B/C11112 B/C11122
C2.2	R-1292	R-002	120x20"	30kg	Cases/ Pallets	0	0	1	3	04/10 /17	C/P11121
C2.3	K-3269	R-003	98x160"	45kg	Case/ Pallets	1	1	0	10	04/10 /17	C/P11012 C/P11031
C2.4	F-9210	R-003	20x32"	30kg	Box/ Cases	1	0	0	13	04/10 /17	B/C00021 B/C00012 B/C00022
	An ASN is received for L-3092 at 06/10/15 of Quantity 10										
	AS ADN is received for L-3092 on 05/10/15 of Quantity 25										
C3.1	L-3092	Pick	24x36"	35kg	Box/ Cases	1	0	1	10	04/10 /17	B/C10131 B/C10121 B/C10111
C3.2	L-3092	Put - Away	24x36"	35kg	Box/ Cases	1	0	1	10	04/10 /17	Expedition Area
C4.1	L-3092	S-011	24x36"	35kg	Box/ Cases	1	0	1	25	05/10 /17	B/C10112 B/C10122 B/C10132 B/C10113 B/C10123
C5.1	L-3092	R-004	24x36"	35kg	Box/ Cases	1	0	1	10	06/10 /17	Expedition Area

** NOTE: Placement is generated in the following scheme

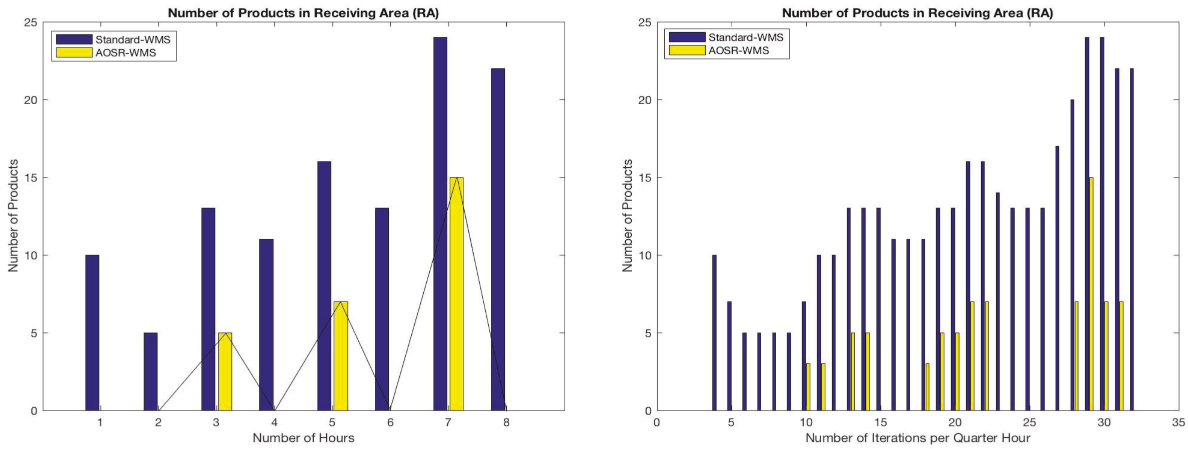
_ / _					
Unit	Hazard Bit	Fast /Slow Bit	Finished / Raw Bit	Rack Level	Rack Number

Figure 7.: Test Cases of the Company's Warehousing Products

The proactive nature of AOSR, which makes it different than a standard WMS, helps to sense the upcoming conflict-states of the system. Conflict-states are the states where the same parameters are sensed for a particular product, e.g. the advance shipment notices (ASNs) and advance delivery notices (ADNs), such as the State ***S3***, where PA decides which products need to be re-slotted.

In the case of ***C3.1-2***, the advance shipment and delivery notices (ASN/ADN) of the same product are sensed by PA. Thus the system can predict that more products are coming, so it re-slots the previous smaller quantity products as it knows that they are needed to be shipped soon (using its prior knowledge of ASN/ADN). This is how it reduces the issues of wandering items and overcrowding of RA or EA.

The AOSR-WMS was initially implemented in ASP .NET framework with Model View Control (MVC) software development architecture and the 3-Tier application development framework in order to increase the security of system design. Models, Views and Controllers have been designed using C sharp, Java scripting, JSON, JQuery and Cascading Style Sheets.

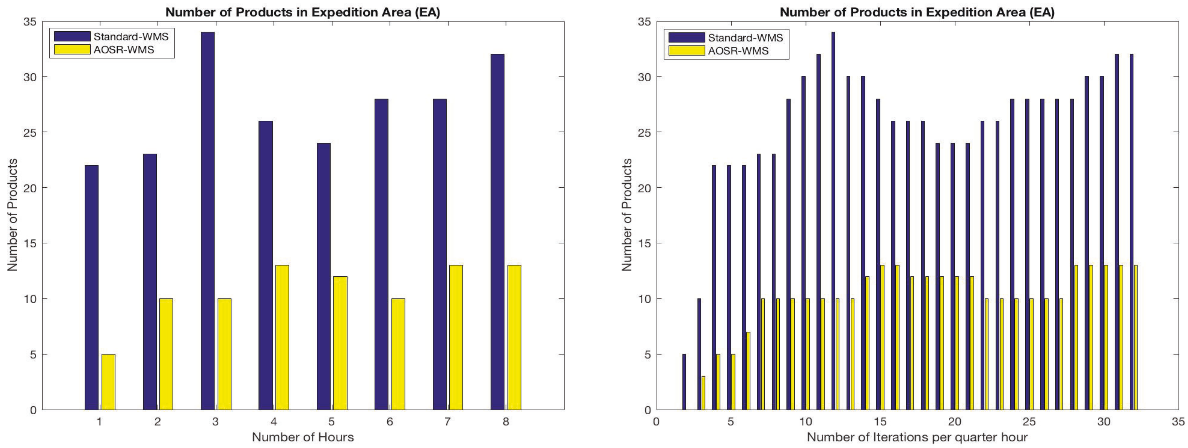


(a) No. of Products per Hour (b) No. of Products per Quarter Hour

Figure 8.: Products in Receiving Area

Figure-8 represents a comparison of the execution results between a standard WMS Chen et al. (2013), Giannikas et al. (2013) and AOSR-WMS. PA-algorithm generates a dynamic placement plan for the products to be placed into the exact racks based on its hybrid logic. The graphs show the average number of products in RA on a particular day on an hourly and quarter-hourly basis. In a warehouse, keeping RA overloaded with products increases the concerns of lost/wandering items that may lead towards stock imbalance Poon et al. (2009); Lu et al. (2014); Richards (2017). The AOSR algorithm is designed to utilise its auto-inspection mechanism through features such as RFID scanning and weight sensing to avoid such problems.

So products, in their certain packing units (i.e. case/pallets, box/cases), stay in RA just for identification and are then placed in the suitable racks as mentioned in the placement plan generated by the planner algorithm. Figure-8 shows a clear difference between the results of a standard WMS and AOSR-WMS. A closer look reveals that, in case of AOSR, in the time span for first two hours, RA is entirely empty and products are shifted to their exact places after consecutive intervals but in case of a standard WMS the RA becomes more and more congested with upcoming products as time passes. During the time interval of Hour 8, the gap is quite apparent, representing a clear performance difference.



(a) No. of Products per Hour (b) No. of Products per Quarter Hour

Figure 9.: Products in Expedition Area

Through adequately defined zone logic there are multiple EAs defined in a warehouse to accurately place the products and to identify the exact location even when they are in EA. Graphs in Figure-9 demonstrate the results of AOSR to be better than a standard WMS, by reducing the total quantity of products in EA by nearly half, on an hourly basis as well as a quarter hourly basis. The AOSR algorithm is programmed to move the products to EA only when it cannot find suitable space in the rack for a product, in the both cases i.e. minimum possible and maximum possible available space. Only those products, whose shipment date is near, are placed into EA, and so, very soon they are moved from EA to the shipping area, leaving the EA free for future possibilities. Thus the objective of maintaining a minimum number of products in EA is also achieved by the AOSR-WMS. This is how the AOSR algorithm keeps EA less loaded so that the demarcation lines remain evident for the unobstructed movement of forklift trucks and floor staff within the shop floor.

6. Conclusion and Future Work

The AOSR-WMS planner strategy, with its underlined architecture, provides a solution for a smart factory under Industry 4.0, particularly for resolving issues of warehousing in SMEs. The AOSF framework includes an end-to-end integration of the whole enterprise, covering both the upstream and downstream operations of the supplier and customer sides. The AOSR algorithm focuses explicitly on the warehouse management side of SMEs. It provides a flexible placement plan with a moderate level semi-automated storage and retrieval strategy that excludes automated conveyor belts and robo-machines, making it affordable for SMEs.

The experimental implementation of the AOSF framework has shown positive results. Currently, all the tasks in ASN/ADN are considered to have the same priority. Handling the tasks of different priorities can be elegant future work in order to provide more flexibility in decision making for the user side. Also, the implementation of Plant Side and multiple dimensions of User Side are also left for upcoming development in this particular project.

References

Abu Al-Rejal, Hussein Mohammed, Jamal D Abu Doleh, Loay Mohammed Salhieh, Zulkifli Mohamed Udin, and Shahimi Mohtar. 2017. "Barriers of Supply Chain Management Practices in Manufacturing Companies in Republic of Yemen: Pre-War Perspective." *International Journal of Supply Chain Management* 6 (3): 246–251.

Andulkar, Mayur, Duc Tho Le, and Ulrich Berger. 2018. "A multi-case study on Industry 4.0 for SMEs in Brandenburg, Germany." In *Proceedings of the 51st Hawaii International Conference on System Sciences*, .

Arnold, Christian, Daniel Kiel, and Kai-Ingo Voigt. 2016. "How the industrial internet of things changes business models in different manufacturing industries." *International Journal of Innovation Management* 20 (08): 1640015.

Business2Community. 2018. "Issues in Wareouse Management Systems." `https://www.business2community.com/product-management/top-5-warehouse-management-problems-solve-02027463`. [Online; accessed 17-July-2018].

Centobelli, P, G Converso, T Murino, and L Santillo. 2016. "Flow shop scheduling algorithm to optimize warehouse activities." *International Journal of Industrial Engineering Computations* 7 (1): 49–66.

Chen, James C, Chen-Huan Cheng, PoTsang B Huang, Kung-Jen Wang, Chien-Jung Huang, and Ti-Chen Ting. 2013. "Warehouse management with lean and RFID application: a case study." *The International Journal of Advanced Manufacturing Technology* 69 (1-4): 531–542.

Commission, European. 2017. “2016 SBA Fact Sheet Germany,.” http://ec.europa.eu/growth/smes/business-friendlyenvironment/performance-review-2016-en. [Online; available 10-June-2017].

Din, Fareed Ud, and Saroop Anwer. 2013. “ERP Success and Logistical Performance Indicators A Critical View:.” *International Journal of Computer Science Issues http://www.ijcsi.org/papers/IJCSI-10-6-1-223-229.pdf* 223–229.

Din, Fareed Ud, Frans Henskens, David Paul, and Mark Wallis. 2018a. “Agent-Oriented Smart Factory (AOSF): An MAS Based Framework for SMEs Under Industry 4.0.” In *KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications*, 44–54. Springer.

Din, Fareed Ud, Frans Henskens, David Paul, and Mark Wallis. 2018b. “Formalisation of Problem and Domain Definition for Agent Oriented Smart Factory (AOSF).” In *IEEE Region 10 Symposium (IEEETenSymp)*, Accepted, awaiting publication. IEEE Xplore.

GATAI. 2011. “German Industry 4.0 Plan.” https://www.gtai.de/GTAI/Navigation/EN/Invest/industrie-4-0.html. [Online; accessed 14-November-2017].

Giannikas, Vaggelis, Wenrong Lu, Duncan McFarlane, and James Hyde. 2013. “Product intelligence in warehouse management: A case study.” In *International Conference on Industrial Applications of Holonic and Multi-Agent Systems*, 224–235. Springer.

Grosse, Eric H, Christoph H Glock, and W Patrick Neumann. 2017. “Human factors in order picking: a content analysis of the literature.” *International Journal of Production Research* 55 (5): 1260–1276.

Gu, Jinxiang, Marc Goetschalckx, and Leon F. McGinnis. 2007. “Research on warehouse operation: A comprehensive review.” *European Journal of Operational Research* 177 (1): 1 – 21. http://www.sciencedirect.com/science/article/pii/S0377221706001056.

Jones, Matthew Michael, Manju Omprakash Juneja, Karthik Gnanamurthy, Kalasagar Kandikuppa, Jason Yun Wei Sheu, Edwin Rudge V William, Govinda Rao Hadagali, et al. 2016. “Consigned inventory management system.” Mar. 31. US Patent App. 14/499,372.

Kang, Hyoung Seok, Ju Yeon Lee, SangSu Choi, Hyun Kim, Jun Hee Park, Ji Yeon Son, Bo Hyun Kim, and Sang Do Noh. 2016. “Smart manufacturing: Past research, present findings, and future directions.” *International Journal of Precision Engineering and Manufacturing-Green Technology* 3 (1): 111–128.

Koster, Ren B. M. De, Andrew L. Johnson, and Debjit Roy. 2017. “Warehouse design and management.” *International Journal of Production Research* 55 (21): 6327–6330. https://doi.org/10.1080/00207543.2017.1371856.

Li, Ling. 2007. *Supply chain management: Concepts, techniques and practices: Enhancing value through collaboration*. World Scientific Publishing Co Inc.

Liao, Yongxin, Fernando Deschamps, Eduardo de Freitas Rocha Loures, and Luiz Felipe Pierin Ramos. 2017. “Past, present and future of Industry 4.0-a systematic literature review and research agenda proposal.” *International Journal of Production Research* 55 (12): 3609–3629.

Lin, Kuan Chung, Joseph Z Shyu, and Kun Ding. 2017. "A Cross-Strait Comparison of Innovation Policy under Industry 4.0 and Sustainability Development Transition." *Sustainability* 9 (5): 786.

Llonch, Marc, Merce Bernardo, and Pilar Presas. 2017. "A case study of a simultaneous integration in an SME: implementation process and cost analysis." *International Journal of Quality & Reliability Management* (just-accepted): 00–00.

Lu, Wenrong, Vaggelis Giannikas, Duncan McFarlane, and James Hyde. 2014. "The role of distributed intelligence in warehouse management systems." In *Service Orientation in Holonic and Multi-Agent Manufacturing and Robotics*, 63–77. Springer.

Lu, Yang. 2017. "Industry 4.0: A Survey on Technologies, Applications and Open Research Issues." *Journal of Industrial Information Integration* .

Ma, Haiping, Shufei Su, Dan Simon, and Minrui Fei. 2015. "Ensemble multi-objective biogeography-based optimization with application to automated warehouse scheduling." *Engineering Applications of Artificial Intelligence* 44: 79–90.

Manzini, Riccardo, Riccardo Accorsi, Giulia Baruffaldi, Teresa Cennerazzo, and Mauro Gamberi. 2016. "Travel time models for deep-lane unit-load autonomous vehicle storage and retrieval system (AVS/RS)." *International Journal of Production Research* 54 (14): 4286–4304. `https://doi.org/10.1080/00207543.2016.1144241`.

Masdefiol, Rodriguez, Maria del Mar, and Fanny Stävmo. 2016. "Industry 4.0–Only designed to fit the German automotive industry: A multiple case study on the feasibility of Industry 4.0 to Swedish SMEs." .

Müller, Julian Marius, Oana Buliga, and Kai-Ingo Voigt. 2018. "Fortune favors the prepared: How SMEs approach business model innovations in Industry 4.0." *Technological Forecasting and Social Change* `http://www.sciencedirect.com/science/article/pii/S0040162517312039`.

Myerson, Paul. 2012. *Lean supply chain and logistics management*. McGraw-Hill New York.

NABaustralia. 2018. "SMEs contribution to Australia's GDP,." https://www.canstar.com.au/business-loans/small-medium-businesses-contributing-half-australias-gdp/. [Online; available 18-June-2018].

Piasecki, Dave. 2005. "Warehouse Management Systems (WMS)." *Inventory Operations Consulting LLC, http://www. inventoryops. com/warehouse_management_systems. htm* .

Poon, T.C., K.L. Choy, Harry K.H. Chow, Henry C.W. Lau, Felix T.S. Chan, and K.C. Ho. 2009. "A RFID case-based logistics resource management system for managing order-picking operations in warehouses." *Expert Systems with Applications* 36 (4): 8277 – 8301. `http://www.sciencedirect.com/science/article/pii/S095741740800715X`.

Preuveneers, Davy, and Yolande Berbers. 2009. "Modeling human actors in an intelligent automated warehouse." In *International Conference on Digital Human Modeling*, 285–294. Springer.

PRWeb. 2017. "Manufacturing industry Growth Survey." `http://www.prweb.com/releases/2013/12/prweb11430148.htm/`. [Online; accessed 18-August-2017].

Richards, Gwynne. 2017. *Warehouse management: a complete guide to improving efficiency and minimizing costs in the modern warehouse.* Kogan Page Publishers.

Sande, Analisa. 2017. "Problems of Receiving Process in Warehouses." https://www.business2community.com/product-management/improve-warehouse-receiving-process-01974283. [Online; accessed 17-July-2018].

Schmidt, Rainer, Michael Möhring, Ralf-Christian Härting, Christopher Reichstein, Pascal Neumaier, and Philip Jozinović. 2015. "Industry 4.0-potentials for creating smart products: empirical research results." In *International Conference on Business Information Systems*, 16–27. Springer.

Techtarget. 2017. "Object Oriented Warehouse Mangement System." http://techtarget.com/definition/warehouse-management-system-WMS. [Online; accessed 31-October-2017].

VDI. 2017. "Standard for Warehouse Management System." https://infostore.saiglobal.com/en-au/Standards/VDI-4490-2007-743848/. [Online; accessed 11-December-2017].

Wide, Process World. 2014. "US Industry 4.0 Plan." https://www.process-worldwide.com/usa-industry-40-the-american-way-a-536602/. [Online; accessed 14-November-2017].

Chapter 7

How the AOSF-AOSR Strategy addresses the need of Industry

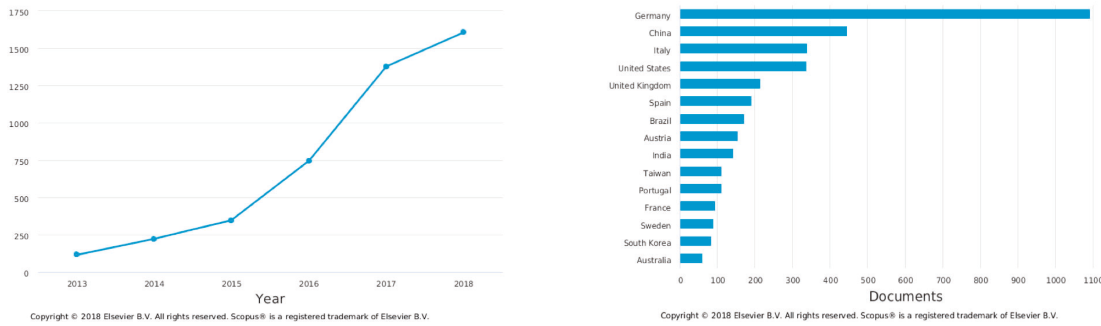
Now we have discussed the details of the proposed system: the concept of AOSF framework in Chapter 4; the problem and domain definition in Chapter 5; and an understanding and design of the AOSR strategy in Chapter 6. This chapter will fabricate the interwoven concepts together by summarising all contributions made so far. Before jumping onto complete validation of the AOSR strategy, this chapter justifies the need of the AOSF-ASOR strategy in relation to the recent research literature. In order to address the highlighted issues, the AOSF framework with its associated AOSR-WMS strategy focuses on the need for a flexible, hybrid and customisable solution that can support SMEs. This two-fold solution does not only provide implementation guidelines for a CPS-based SC framework under Industry 4.0 but also brings the robustness and seamlessness in WMS operations for SMEs. Where the AOSF framework provides flexibility for customising the architectural layers of its recommended framework (detailed in Chapter 4), the AOSR strategy provides a volatility in configuring the warehouse structure (detailed in Chapter 6). Hence, the proposed system presents its support and contribution in building a compatible system for SMEs under the umbrella of Industry 4.0. This Chapter is segmented into two sections, addressing two inter-connected parts of the overall proposed system: the AOSF framework in Section 7.1 and the AOSR-WMS strategy in Section 7.2.

7.1 The Need of AOSF framework

Recent research provides outstanding efforts in optimising Supply Chain (SC) and Logistical Management for the manufacturing industry [173, 197]. Researchers claim that state of the art technological concepts such as the Internet of Things (IoT), Big Data, Cloud Computing and Multi-Agent Systems (MAS) are the main tracks that can provide continuous support to bring more ubiquity in the domain of manufacturing supply chains [181]. More specifically, recent Information Systems (IS) and Enterprise Resource Planning (ERP) Systems provide automated business processes to overcome issues in coordination and communication within enterprise levels, but both the literature and industry have often raised concerns of lack of customisation and incompatibility as overwhelming issues with existing systems, demanding more flexible solutions [184].

Recalling from Chapter 1, the literature claims that among the developments to overcome issues in the manufacturing industry, the concept of Industry 4.0 has been welcomed globally since its inception in 2011, but even after extensive research in this domain, SMEs are mostly overlooked [133]. There is still a need for a more dynamic but moderate-level solution to provide SMEs with the benefits of Industry 4.0. The AOSF framework presented in Chapter 4 is an attempt to fill this gap by providing a comprehensive CPS-based Industry 4.0 SC framework with a focus on SMEs. It provides flexibility for adding or reducing the base layers (connection levels) as per the need of individual SMEs. The concept of Enterprise Central Unit (ECU) in this framework provides connectivity to all the bottom layers and links with the top level cyber layers, where back-end server systems provide cognitive abilities and backup facilities to the whole system.

Figure 7.1 provides insight into recent research trends over 6 years, from 2013 to 2018. As highlighted in Figure 7.1a, the potential and frequency of research in Industry 4.0 are increasing every year by a substantial amount, with an expectation of more contributions to be seen in coming years. Though such research trends provide more efficient ways and novel ideas to manage industrial set-ups, offering more flexibility and automation, recent claims made by researchers reveal that Industry 4.0 mainly focuses on large enterprise [8], and only marginally on SMEs [164]. Researchers have highlighted potential contributions needed in the domain on Cyber-Physical Systems (CPS) and Agent Technologies to provide a solution to future industrial systems [144], emphasising that the compatibility with SMEs is still an open question [133]. As explained in the previous chapters and validated in the next chapter, AOSF framework provides a solution to the problems of flexibility and compatibility of SC frameworks under Industry 4.0 with respect to SMEs.



(a) Contributions by year

(b) Contributions by region

Figure 7.1: Publication Statistics 2013-18 "Industry 4.0" [166]

In the manufacturing industry, SMEs contribute a significant proportion in the economy, e.g. in Australia SMEs contribute nearly 57% of the annual GDP as per a report by NAB [134], and in Germany, there are around 3.6 million firms in the SMEs sector, which employs around 62.8% of the workforce [5]. In order to keep up with industrial advancement, Germany has proven to be a leader in the domain of Industry 4.0 research and development (as reflected by statistical trends in Figure 7.1b), which complies with its proportion of SMEs in the industrial sector. One factor for such a trend could be the push by government policies, such as in Italy, where the government has recently announced a 250% tax depreciation on upgrades to new IT systems for the technological transformation of production processes [180]. However, the industrial sectors in Australia and South Korea are lagging behind in their required pace in order to maintain the right balance to justify more than 50% of annual GDP-share contributed by SMEs (the AOSF framework with its associated AOSR strategy may contribute to alleviate the issue within Australia and around the globe). The right balance between research trends may help in unveiling the real hindrance in providing SMEs with the true advantages of Industry 4.0. In fact, this paradox can possibly endanger the growth of SMEs [5]. Hence, for widespread implementation of Industry 4.0, its applicability in SMEs is a crucial aspect [119]. In order to bring SMEs the expected benefits of Industry 4.0, its compatibility with semi-autonomous systems is highly needed [43] [133].

The lack of clarity in the implementation process with proper SC elements for SMEs is one of the limitations of applying Industry 4.0 [127]. To overcome this issue the AOSF framework provides a CPS-based end-to-end SC framework, which incorporates both the front-end and back-end of an enterprise. As warehouse management systems

(WMS) serve as a backbone of SMEs, AOSF framework provides an associated semi-autonomous Agent-Oriented Storage and Retrieval (AOSR-WMS) strategy (explained in Chapter 6) to cater to the problems of warehousing in SMEs. The three-tier-based architecture of AOSF provides not only enterprise integration strategies but also agent communication and negotiation strategies that allow seamless execution of every trigger within the supply chain.

7.2 The Need of AOSR Strategy

SC is an integration of key business processes and elements to build a value chain for all participating stakeholders. SMEs, though they play a significant role in the economy, face a number of issues originating from inefficiencies of SC networks [142]. SC operations rely heavily on warehouses [1], as every informational flow ends or corresponds to the disposition of products/materials from or to the warehouse, respectively (explained in detail with a test scenario in Chapter 8). WMS systems are usually employed in the manufacturing industry for handling day-to-day operations in a warehouse efficiently. Literature often relates SC issues with WMS, as it is the backbone of a SC network [1]. There are many studies conducted to highlight problems in WMS, such as the survey conducted in [142] related to process failure and root-cause analysis in SMEs. Out of 13 SC related issues, 8 were from warehouse inefficiency, particularly from WMS inaccuracies. Similarly, the study conducted in [1] highlighted warehouse inefficiencies as a main barrier in managing SC operations. Hence, for uprooting the causes of failures in supply chains, warehouse management is an integral and essential part to be rectified.

Industrial automation is becoming more and more incumbent as procedures are becoming very complex regardless of whether the systems work autonomously. Different ideas are contributed to solve the problem of resource scheduling, utilising advanced AI techniques, e.g. Petri Nets [18] and Genetic Algorithms [189]. Similarly, many contributions are made into the track of production systems automation such as MASINA [3], SCDIA [117] and PABADIS [112] architectures, which discuss plant automation systems. Such models extend support for providing operational flexibility within SC networks. Although extensive research has been conducted to provide complete autonomous systems, proper SC implementation guidelines in compliance with Industry 4.0 standards and its implementation in warehouses, specifically, is still an open research area [41].

Literature often includes solutions in the domain of production planning and control, e.g integrated agent based SC systems [52] to reduce the crashes in a centralised supply chain network for automotive industry (tractor manufacturing). This system highlights the flexibility offered by MAS techniques by combining the existing supply chain scheduling project-series, e.g. DISPOWEB (Dispositive Supply-Web Coordination) [190], KRASH (Karlsruhe Robust Agent Shell) [116], IntaPS (Integrated Agent-based Process Planning) [39], FABMAS (Agent-Based System for Production Control of Semiconductor Manufacturing Processes) [130] and ATT/SCC (Agent-based Tracking and Tracing of Business Process) [200]. Although these projects provide support for production planning and control, issues related to warehouse management persist [43].

From the perspective of warehouse management, several solutions exist, such as Flowshop Algorithm [28] and EMBBO [114]. Even though the idea of Industry 4.0 is transforming the manufacturing industry, recent research claims that it cannot be purely mapped to SMEs to resolve the problems in warehouse management [5, 133]. High cost, infrastructural change and incompatibilities are the main reasons that SMEs are still facing warehousing issues [43]. In order to bridge this gap, the AOSR-WMS strategy (presented in Chapter 6), presents a hybrid mechanism for warehouse management by providing an efficient slotting and re-slotting based strategy. It reduces the concerns of overloading Receiving and Expedition Areas (RAs/EAs), maximises the storage of products within the racks and ultimately improves the management of the shopfloor (with performance validated by test scenarios in Chapter 8).

For SMEs, as the business processes are usually quite flexible, the AOSF-AOSR strategy presented in this thesis provides a comprehensive solution that can be implemented thoroughly. SMEs can implement this solution gradually in parts i.e. by establishing a conceptualised CPS (as discussed in Chapter 8) and then by customising AOSR framework as per their business need, which is one of the benefits of using AOSR-WMS planner that it provides volatile settings of racks, receiving areas and expedition areas (to be easily modified/redefined as detailed in Chapter 6). Hence it provides a flexible layout that can be adopted by exiting industries in a step-by-step fashion.

In this Chapter, we have discussed a synopsis of the contribution made by this system, to help provide a clear understanding of its benefits; the next chapter will focus on a thorough validation of this system. Chapter 8 will provide details about algorithmic heuristics of AOSR strategy, in connection with Chapter 6, which includes the design and initial experimentation of AOSR system. Extensive results, generated from the prototype developed in JADE [81], are also presented in the next chapter, in order to provide a comparison with the existing standard-WMS methodology [29, 110].

Chapter 8

Validation

Testing and validation are important tools to assess performance of a system. This chapter includes validation test cases and scenarios applied to the AOSF framework and its associated AOSR-WMS strategy to affirm the validity of the overall system. It is sub-divided into two independent sections as outlined below:

Section 8.1 includes a test scenario within the supply chain of a firm and relates it with two different possible cases of information exchange from the front-end customer side and back-end supplier side. It also highlights the importance of the Business Process Re-engineering (BPR) strategy recommended by the AOSF framework, which is further endorsed by prototype results in Section 8.2.

Section 8.2 provides a complete overview of AOSR's updated algorithmic heuristics, its validation with test datasets and their implementation results taken from the prototype developed in JADE. Based on these results, the second part of the chapter discusses the use of multiple warehousing and product placement/retrieval mechanisms, e.g. Zoning Logic, FIFO Logic and Pick from/Put to the Fewest Logic. It also provides a comparison of the recommended AOSR hybrid product placement and retrieval strategy with standard WMS strategies. This section includes detailed results from prototyping the hybrid-logic-based AOSR algorithm, which combines not only all the aforementioned logic schemes but also the 'Pick from/Put to the Nearest logic', in order to reduce the overall activity-time within the shop-floor. It also includes validation of how the *6-Feature* strategy recommended by the AOSR system helps in bringing improvement and pro-activeness within a warehouse.

8.1 Test Scenario and AOSF

Supply Chain (SC) is a philosophical boundary-less network within a business set-up that prevails from the supplier side towards the customer side. Several different events could occur from any of the constituent parts of a SC network, e.g. supply chain management (SCM), enterprise central unit (ECU), logistical information system (LIS) or customer relationship management (CRM). This section addresses two possible approaches to create a comprehensive informational flow passing through different existing SC components:

1. *Create the bulk approach*, where the events are triggered from the supplier side and require the involvement of different units of the SC network. This case can be further segregated into two sub-scenarios, *Scenario 1A: Creating the bulk from inside* and *Scenario 1B: Creating the bulk from outside.*
2. *Break the bulk approach*, where the action is invoked from the customer side, creating a wave of initiation of different sub-components of the supply chain up to the warehouse.

Both of the cases, with their sub-scenarios, are reflected by the linear representation of the SC network in Figure 8.1, detailed below.

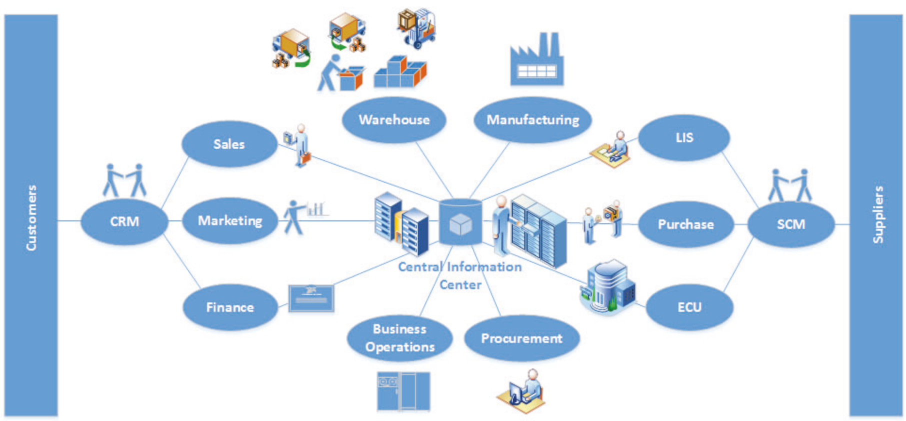


Figure 8.1: A Linear View of Supply Chain Network

Scenario 1A: Creating the bulk from within the enterprise is a scenario where the manufacturing unit informs the central information centre about the completion of a particular batch, which is further updated to ECU. ECU collects details about the execution of production planning, process and disposition [69] related to particular finished or semi-finished (raw) products to be stored in the warehouse. After processing the data, ECU transforms it to decisive information and initiates a trigger, invoking a call for products to be stored with the details about dispositioning from the manufacturing side and delivery towards the warehouse side.

Scenario 1B: Creating the bulk from outside is a scenario where the products are to be delivered from suppliers to the firm. The SCM component is the primary interface that corresponds to the requirement of suppliers and deals with delivery details via LIS. SCM also performs operational planning, execution of procurement and completion with the purchase department, which are sub-operations of inter-departmental communication [171]. ECU collects data transmitted by SCM and LIS from the central information centre, and then, after processing that data, invokes a call to deliver the right batch to the warehouse with all the delivery details.

In the second approach of *Breaking the bulk*, the trigger is initiated from the front-end customer side. The CRM component is the main interface for dealing with the requirement of upcoming orders from the customer side. CRM coordinates with the department of sales and marketing and posts the data to the central information centre [171]. Then the information related to a particular shipment, in liaison with the sales department, is transmitted to the warehouse side. Section 8.1.1 highlights details of the test cases under study in this chapter, for all the possible triggers initiated, in a routine day on an hourly basis in a distribution warehouse.

From the perspective of mapping Industry 4.0 standards to SMEs, three particular aspects, as mentioned below, are usually recommended through the use of RFID technologies, mobile user interfaces and auto/predictive control of inventory management [148]:

- Smart Logistics, providing connected units with predictive features;
- Smart Production, providing sensor-based environments within production plants; and
- Organizational/ Managerial model, providing comprehensive control to managerial staff.

AOSF framework takes all these recommendations into account and provides a comprehensive layout not only for organisation and modelling of an SC network but also for how it works in maintaining vertical, horizontal and end-to-end integration, which is an important factor to keep the whole system updated.

AOSF framework is based on a Cloud-based CPS architecture that provides the flexibility and scalability of adding Big Data features as needed in the future. At the moment, most SMEs are not considering data as a source of added value [17]. Also, the

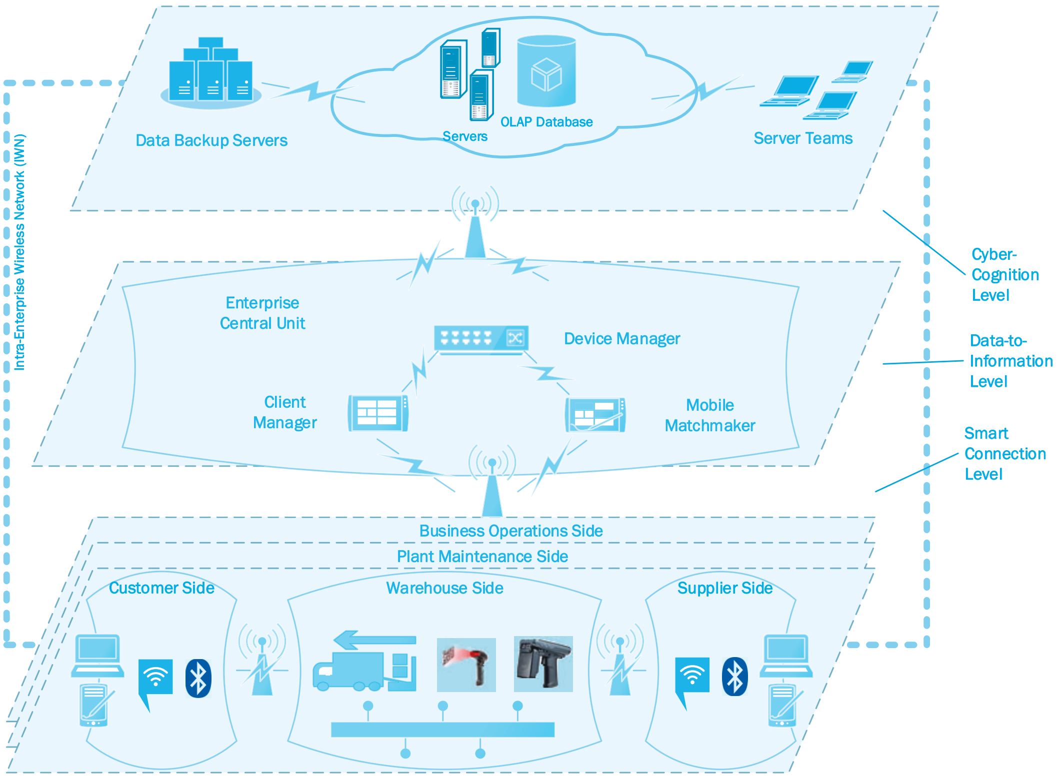


Figure 8.2: Extended View of AOSF Framework

extensive use of collaborative robots is not exploited by SMEs yet and does not seem possible in the near future because of the high infrastructural cost involved with such automation [14]. Recalling the concepts of the tier-based AOSF framework and its extended view in Figure 8.2, this architecture better caters to the cases discussed in a linear SC structure as it provides a proper integration mechanism though an Intra-Enterprise Wide Network (IWN), which also provides three dimensional enterprise integration (detailed in Chapter 4). The traditional SC elements such as the SCM, CRM, plant side, business operation side and warehouse side, with all the smart devices, are part of the Smart Connection Layer, which further provides connectivity to ECU. ECU in AOSF architecture is considered a focal point which serves as a middle layer sensing the data and transforming it into decisive information. All the backup and monitoring facilities are set up at the Cyber Cognition Layer which provides overall cognitive abilities to the system. Such a three-dimensional structure of the AOSF framework also helps in maintaining a proper back-up at the cloud layer, while keeping all the constituent elements updated concurrently. Agent orientation also gives the AOSF framework a further benefit of flexibility where agents interact with each other for a particular resource constraint and themselves are helped by utilising their own local inference engine and belief sets. AOSF framework recommends standard classification of reflex agents, utility-based agents and goal-based agents, i.e. Smart Device Agent (SDAs), User Side Agent (UAs) and Mediator Agents (MAs), in order to provide decentralised decision making, thus making operation seamless and robust.

8.1.1 Dataset and Test Cases

In order to provide a solution to improve warehouse management in SMEs, the AOSF framework recommends its associated AOSR-WMS mechanism with its *6-Feature* strategy [43], which is prototyped in JADE [81], as detailed in Chapter 3. For a thorough validation of this system, the data used to evaluate the test cases, for different categories of products in different scenarios, is represented in Tables 8.1 and 8.2. All the data categorisation of the products applied to the AOSF framework and its associated AOSR algorithm are taken as a test/example case, which can be modified as per business need. The details of different classes and categorisation of products in these test cases are extracted from the online source provided by DGI Global [57] and Eurosped [45] warehousing and logistics companies. In order to build a comprehensive dataset that includes maximum variation and can be considered as a representative for a large scale applicability, several different features are included such as product classes, their characteristics, SKUs and different situations of product delivery and shipment. The data used to validate this system, is stored categorically within the highlighted constraints as detailed in Tables 8.1 and 8.2. This comprehensive data set does not only include one type of product category, it consists of the information of several characteristics of products e.g SKUs, quantity and products classes, from several different industrial sectors e.g., electronics industry, medical industry, textile firms, paint and glass industry.

Table 8.1 takes 32 different triggers into account and categorises them into the aforementioned generic scenarios: *Scenario 1A (Creating the bulk from inside), Scenario 1B (creating the bulk from outside) and Scenario 2 (breaking the bulk)* (in column 2), with their initiator SC-unit in column 3. The first 15 cases are related to the case *Creating the Bulk*, both from inside and outside, and the others reflect the scenario of *Breaking*

Table 8.1: The dataset used for hourly *Creating/Breaking the Bulk* Test Cases

Tr. #	Case	Initiator	Hours	Product ID	Quantity
1	1A	ECU	1^{st}	P-9001	17
2	1A			P-9002	13
3	1B			k-9804	23
4	1A			K-2098	27
5	1B		2^{nd}	L-3092	17
6	1B			K-9803	33
7	1A			F-9210	47
8	1A			L-2801	33
9	1A		3^{rd}	F-2830	23
10	1B			C-3921	27
11	1B			R-3392	67
12	1B			R-1292	43
13	1B		4^{th}	P-8372	53
14	1A			K-3269	27
15	1B			R-3390	67
16	2	CRM		P-9001	5
17	2		5^{th}	P-9002	25
18	2			k-9804	33
19	2			K-2098	67
20	2			L-3092	53
21	2		6^{th}	K-9803	23
22	2			F-9210	37
23	2			L-2801	73
24	2			F-2830	47
25	2		7^{th}	C-3921	33
26	2			R-3392	27
27	2			R-1292	43
28	2			P-8372	17
29	2		8^{th}	K-3269	23
30	2			R-3390	17
31	2			P-9001	27
32	2			P-9002	13

the Bulk. This data set is segregated into 8 divisions (as represented in Column 4) with respect to working hours and details of shipment and delivery within the warehouse for each hour. For the sake of clarity and uniformity only 4 triggers per hour are considered; there are usually, 0-6 data transactions per hour depending upon the size of the enterprise [45, 57], so this is realistic. Every product is assigned a unique *Product Id* (as represented in Column 4), which encapsulates all the details related to the characteristics of a particular product. In column 6, the quantity represented is a random number (in the range from 1-100, as per available industrial data) of products being requested by any of the SC components from the network, e.g. ECU or CRM, which corresponds to a particular delivery or shipment respectively (see Section 8.1.3 for a further examination of these random values).

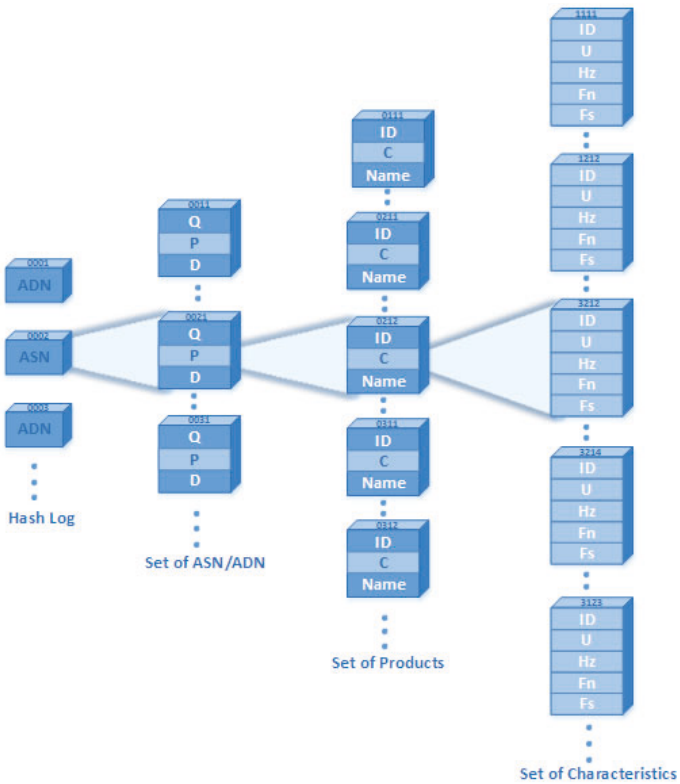


Figure 8.3: 4-Level Storage of Knowledge Structures

This dataset comes with combinations of multiple possibilities such as delivery and shipment instances with varying products details (based on product category, characteristics, SKU, quantity and due date); hence it is stored in four levels of data abstraction (we call them 4-Level knowledge structures). Figure 8.3 explains the details of these knowledge structures utilised by the AOSF framework. These main knowledge structures are contiguous and continuous logs of shipment and delivery details, where each log is related to a particular case. The AOSF framework stores these information logs as *HashMaps*, and refers to them as *hash logs* as reflected in Figure 8.3. In an instance of *hash log storage* there could be n possible Advance Shipment and Delivery Notices (ASN/ADN). In all cases, the origin of the details is the product that needs to be dispositioned in a particular Stock Keeping Unit (SKU). For every product, the uniquely identifiable *Product ID* corresponds to a specific category which could be hazardous products, unfinished products or brittle/fragile items. Products could also be distinguished based on their packing units (SKUs) such as Each per Box (E/B), Boxes per Case (B/C), Cases per Pallet (C/P), Barrels, (Br), Cylinders (Cyl) or Single Pallets (SP).

Table 8.2: Categorisation/Classification of Products with respect to Characteristics

Sr. No	Product ID	Product Name	Characteristics			
			SKU	Hazard	Fast	Finished
1	P-9001	Small Electronics	E/B	✗	✓	✓
2	M-1001	Medical Supplies	B/C	✓	✓	✓
3	P-9002	Household/Hygiene	B/C	✗	✓	✓
4	k-9804	Large Electrical App.	B/C	✗	✗	✓
5	K-2098	Textile Items	B/C	✓	✗	✓
6	L-3092	Crops Prot. Materials	B/C	✓	✗	✗
7	K-9803	Glass Bars	B/C	✓	✓	✗
8	F-9210	Paints/Chemicals	B/C	✗	✗	✗
9	L-2801	Oils/Lubricants	B/C	✗	✓	✗
10	F-2830	Chalking Material	C/P	✗	✗	✓
11	C-3921	Spare Parts	C/P	✗	✓	✓
12	R-3392	Stationary/Paper Logs	C/P	✗	✗	✗
13	R-1292	Industrial Goods	C/P	✗	✓	✗
14	P-8372	Dyes Pallets	C/P	✓	✗	✓
15	K-3269	Large Mechanical Parts	C/P	✓	✓	✓
16	R-3390	Pest Control Powder	C/P	✓	✗	✗
17	P-9003	Household Equipment	C/P	✓	✓	✗
18	R-3292	Alkaline Substances	Br	✓	✓	✓
19	K-4940	Large Liquid Containers	Cyl	✓	✓	✓
20	K-9805	Long Glass Screens	SP	✓	✓	✓

The categorisation of products applied to the AOSF framework and its associated AOSR algorithm are highlighted in the dataset represented in Table 8.2. As highlighted above this data is taken as a test/example case, which can be modified as per business need. To apply the data set to AOSF and AOSR strategy, the products are classified as per four parameters: their SKU, Hazard category, Movement (slow or fast) and Finished or Unfinished. In order to provide comprehensive testing, six different types of SKUs (E/B, B/C, C/P, Br, Cyl, SP) and six different types of characteristics (binary values of hazard, fast and finished classification) are considered with 20 different classes/categories of products. The AOSF framework and AOSR strategy, both provide the flexibility to accommodate such variability of products. The details about how AOSR, with its *6-Feature* strategy, provides the scalability for the same or different categorisation of products are explained in detail in Chapter 6. So, the number of product possibilities catered by AOSF and AOSR can be represented as below:

$$n^2 * Categories * Characteristics * SKUs. \tag{8.1}$$

Hence, the possibility of finding a certain product x in n number of ASNs and ADNs with any particular category out of 20, belonging to any characteristic out of 6 and having any SKU out of 6, can be represented by the following equation:

$$P_{(\mathrm{x})} = \frac{1}{n^2 * 20 * 6 * 6} \tag{8.2}$$

In a complex warehouse environment, sometimes finding product possibilities helps in adjusting the space and product allocation with the designated areas. A simplified solution to find a product possibility in a certain warehouse region can help to improve the searching ability and efficiency of a warehouse solution [1]. As AOSR WMS keeps the solution more generic, it becomes easier to apply different set of requirements as per business need.

8.1.2 Results and Discussion

After applying the aforementioned test cases to the AOSF and AOSR strategy in comparison with a standard WMS strategy (explained in detail in Chapter 6), the results are represented in Figure 8.4, which reflects the inclination of both strategies with three performance metrics. In order to bring clarity in results and to provide better recommendations, this research is constrained to three very important key performance indicators (KPIs):

1. number of products stored in racks;
2. number of products kept at receiving area (RA); and
3. the number of products placed in expedition areas (EA).

Low performance in managing these three parameters results in basic WMS issues such as receiving area overloading, demarcation lines vanishing, manual re-slotting and wandering/lost items [23]. Literature has often mentioned persisting SC and WMS issues, and the main reasons behind such problems are mostly the unmanaged receiving and expedition areas [156] and unmanaged storage capacity [110]. A higher number of products within the racks is usually considered as a performance metric for efficiency in warehousing [58].

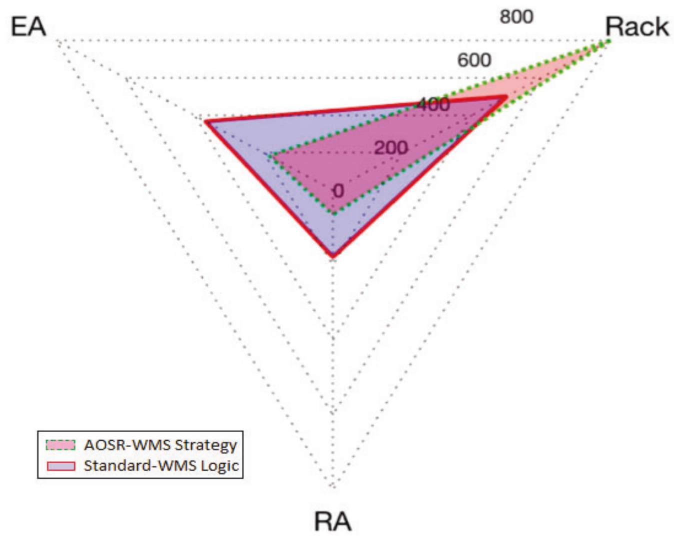


Figure 8.4: Performance Inclination of AOSF and AOSR strategy

Figure 8.4 demonstrates how the focus of the AOSF recommended AOSR strategy is different from a standard WMS in a linear SC model. In the graph, all three aforementioned KPIs are set to be the preference points, which reflect the number of products in the three main areas of the warehouse, RA, EA and Racks. As per the concentration of data points, the graph shows more tendency towards a certain corner. The deflection in the shaded areas reflects the condensation of data points, which shows the preference of the strategy. For example the placement of upcoming products within the defined racks is the main priority of AOSF-AOSR strategy (represented in the orange shaded region) so the deflection of data is towards the point *'Rack'*, while the standard approach uses a balancing approach (represented by the purple shaded region) and reflects a balanced data deflection for all the three points. In the case of AOSR, the graph explains the deviation of data towards the higher number of products at *'Rack'* point with almost 800 products out of 1080 total products. The manual method of sorting and identifying the proper location of received products takes almost 41% of the time and effort in a standard SC warehouse [58]. Conversely, AOSF framework and the *6-Feature Strategy* of AOSR recommends the BPR-based proactive approach of sensing the ASNs and ADNs (advance shipment and delivery notices) by utilising its cognition features and fully integrated environment, which is why it maintains a very low number of products in EA and RA (almost 100 on average in this scenario as compared to 300 using the standard approach).

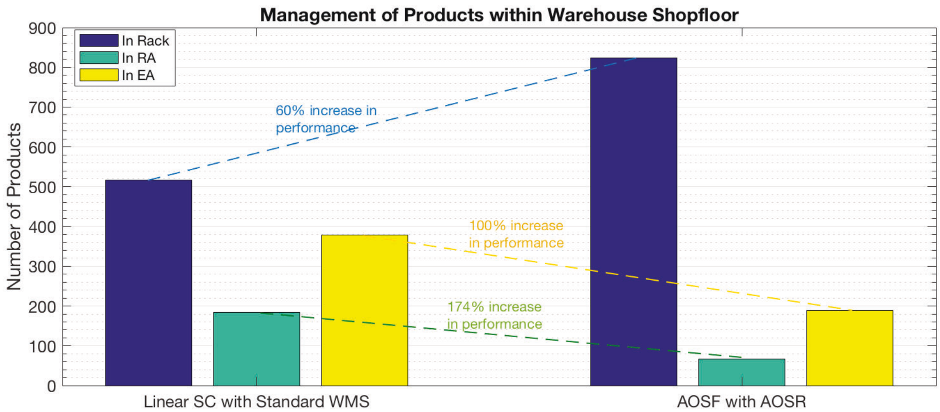


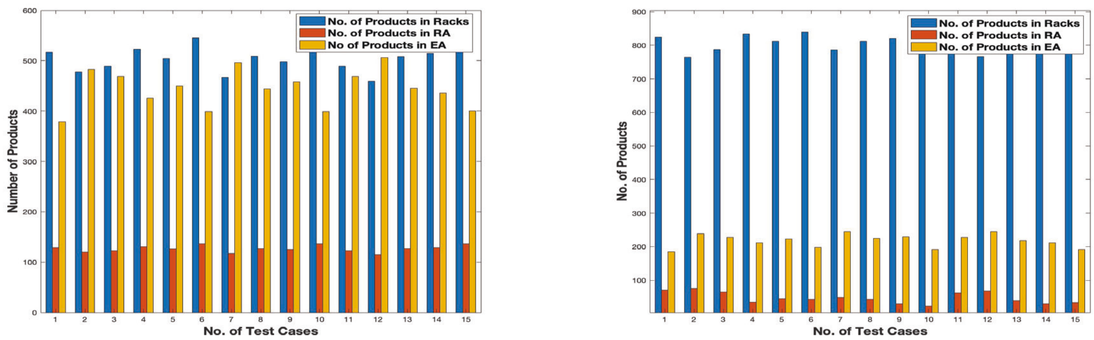
Figure 8.5: Linear SC Network/Standard WMS vs AOSF-AOSR Strategy

Based on the results shown in 8.4, which highlight the preference of AOSR strategy in comparison with a standard WMS, the results shown in Figure 8.5 represent the performance increase while utilising AOSR strategy. A constrained test case of around 1,000 products in rotation is applied to both approaches. Following the AOSF architecture and recommendations of the AOSR strategy [43], results are better in all three of the aforementioned performance metrics. There is a significant increase of almost 60% in the products stored in racks by using AOSR recommendations. In order to adjust the upcoming products, AOSR-WMS strategy provides a comprehensive zoning plan within the shop-floor to cater to a wide range of products (presented in Table 8.2) with several different characteristics. This provision of volatility in different zones provides flexibility and stability to cater to any future change in business operations. The proactive and predictive nature of AOSF-AOSR strategy, as discussed in this case, reduces the number of products in EA and RA to half and less than half respectively, leading to improvements of 100% and 174% in these areas. Section 8.2 will further address details about how this increase in performance is achieved.

8.1.3 Aggregated Results with Random Values

The general applicability of a system can be validated by analysing its performance over a wide range of data. As explained in the previous section, a diverse case study with a broad scope helps indicate the performance efficiency of AOSF and AOSR strategy. In order to confirm its validity over a wide range of possible scenarios this section describes several tests applied to the AOSF with AOSR strategy in comparison with the standard approach.

The dataset described in Section 8.1.1 has been modified with different random values for the quantity, to help ensure this generic case study is more widely applicable. The details of the values used are attached in the appendix B.



(a) Standard WMS Strategy (b) AOSF with AOSR Strategy

Figure 8.6: Performance Results with Random Data

Figure 8.6 represents the detail of the first fifteen (out of thirty) test cases applied to the subject strategies. For clearer visualisation only the first fifteen cases are displayed in the graph, with details of all cases included in Appendix B. A closer look to Figure 8.6 demonstrates that there is a similar trend for this wider range of data as there was for the case study described in Section 8.1.2. Figure 8.6 demonstrates the number of products in all three areas (Racks, EA and RA). The products in racks are represented by blue bars, products in EA with orange and the products in RA are represented with red bars. The standard WMS strategy tends to balance between all the three aforementioned KPIs (number of products in racks, RA and EA), while for the AOSF recommended strategy, the main priority is to manage the maximum number of products in the racks. Figure 8.6a, on the left, shows that, out of 1080 products, at most around half of them are in racks and, from the remaining products, a major proportion are stored in EA.

Furthermore, approximately one-quarter of the total products seem to be stuck in the receiving area. This trend can be observed in all test cases. As previously stated, such a situation leads towards the problems of mismanagement onto the shopfloor and causes the concerns of the unmatched stock count, missing or wandering items and extended lead time of order processing.

On the other side, in Figure 8.6b the major proportion of the total products, almost 80%, are maintained in the racks, in almost all test cases. As explained in detail in Chapter 6, the AOSF/AOSR strategy is designed to prioritise the placement of as many products in the racks as possible by utilising its slotting and re-slotting strategy to make more space available within the racks for upcoming products. This is why it succeeds in maintaining a very low number of products in RA: around 40 - 50 products as compared to 100 - 150 products when using the standard approach. Also, there is a good difference in the products detained in EA when using AOSF, with around 180 - 210 stored in EA with AOSF as compared to 390 - 440 with the standard approach.

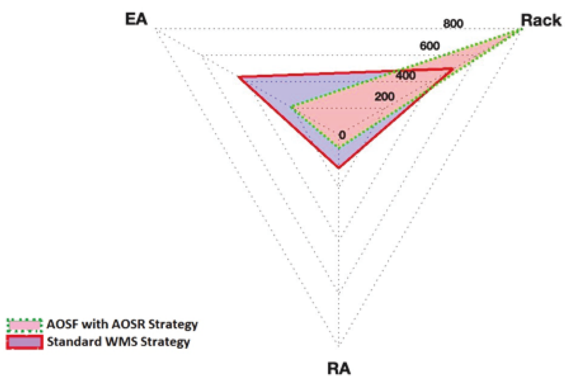


Figure 8.7: Performance Inclination with Random Data

Similar to the visualisation presented for discussing the performance inclination of both strategies in the detailed case study in section 8.1.2, Figure 8.7 represents the average results of the thirty test cases used in this section. The standard WMS strategy, as already discussed in the detailed data value graph in Figure 8.6, tends to maintain the balance between the number of products in all the three sections of shopfloor: racks, EA and RA. It is represented by the purple shaded region in the graph. On average 507 products are in racks, 444 are in EA and 127 are in RA. For AOSF, the focus is to maintain the products in racks, as can be seen by the deflection of the graph towards the corner of 'Rack'. That means there are more data points towards the 'Rack' corner as compared to the others. On average 814 products are stored in racks, 215 are placed in EA temporarily and only 52 are at RA. These number provides quite a fair improvement in efficiency as presented by Figure 8.8.

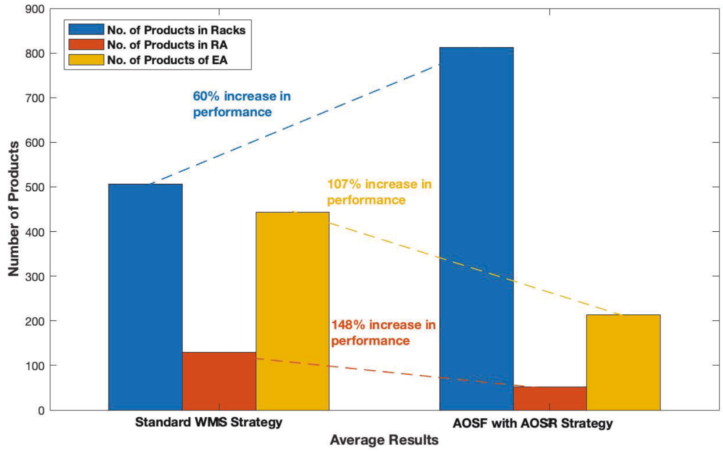


Figure 8.8: Improved Efficiency with AOSR/AOSR Strategy

Figure 8.8 demonstrates the performance improvement over the average of the thirty random test cases. The blue bars represent products in racks, orange bars show the number of products in EA and red bars represent the products in RA. The number of products maintained in racks by utilising the AOSF recommended strategy brings a 60% increase in the number of products stored in racks, a 107% decrease in the number of products in EA and a 148% decrease of items in RA. These numbers are very close to those obtained with the single case study in Section 8.1.2, indicating that the case study is a good representation of typical results. The consistency of performance while utilising AOSF recommended strategy speaks about its validity and broader applicability.

8.2 Test Scenarios and AOSR

The AOSR algorithm provides a simple but overarching solution to the problems of scheduling products and their slotting and re-slotting within a warehouse. It focuses on the main reasons that cause major problems in warehouse management such as attempts to reduce the number of the products in receiving areas (RAs) and expedition areas (EAs) and to maximise the number of products within the defined racks [58]. This section provides the heuristics of AOSR algorithm and discusses the results acquired in different scenarios.

8.2.1 Algorithmic Heuristics of AOSR

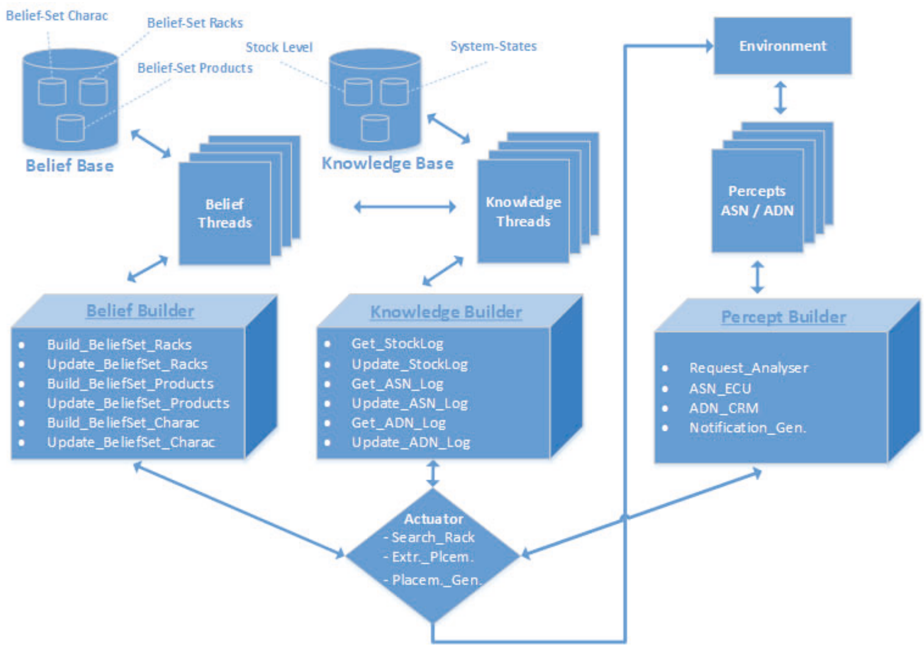


Figure 8.9: Algorithmic Architecture of AOSR

Based on percepts from the environment and descriptive initial states of the system, AOSR generates a comprehensive placement plan, utilising predefined sets of actions. The proactive nature of AOSR strategy provides help in managing the space within the warehouse to cater to upcoming products and its hybrid strategy supports the efficient slotting and re-slotting of products between different locations within the shop-floor.

AOSR algorithm is based on the classical BDI agent model structure [160] and follows the constructs and agent classification of AOSF framework [41], which shapes it into a dynamic solution in order to support operational flexibilities in the future. Figure 8.9 represents a detailed architecture of AOSR algorithmic heuristics. AOSR Planner Agent (PA) utilises the Information-Sets (highlighted in Section 7.1) related to different classification of products, their characteristics and racks, which serve as the *Belief Base* for PA, and information related to current stock levels and systems states serves as the *Knowledge Base* for it, which may be further updated by actuators. For PA, three main segments play the role of actuators: *(i) Placement Generator*, *(ii) Extract Placement*, *(iii) Search Rack.* These actuators sense the percepts coming from the environment and update the *Belief Builder* and *Knowledge Builder.*

There are some other supporting sub-functions within the AOSR algorithm which support the role of actuators, e.g. *Extract Characteristics* and *Generate CharcID* to find the details of the characteristics of a certain product or vice versa, respectively. This composite architecture provides the agent with properties and transforms AOSR strategy into a pure agent-oriented solution. The functionality of these components has been implemented in a prototype using JADE. Algorithms 1-5, described later in this section, and fully detailed in Appendix A, give details of this implementation and are presented in appendices as pseudo-code.

Flexibility and reconfigurability are the features that make AOSR strategy adaptable for any particular implementation environment. All the baseline information sets are stored in a form of Belief-Sets (Belief-Set Products, Belief-Set Characteristics, Belief-Set Racks), which build the Belief Base for AOSR algorithm. Theses Belief-Sets can be modified if needed, which provides volatility and avoids the concerns of hard coded information. In order to provide a variety of different classifications of products in this scenario, AOSR utilises 20 different categorisations of AOSF products as mentioned in Section 8.1.1. The defined Set of Racks in the AOSR-recommended *6-Feature* strategy, as detailed in Chapter 6, provides a flexible capacity. Each zone, in the recommended baseline layout, has different categorisation of racks that can be further divided into different levels. The number of racks and levels are flexible and can be configured initially before launching the setup. As a constraint for experimental purpose in this thesis, all the racks are divided into 3 levels with each level containing space for 5 SKUs, yielding a total of 15 SKUs in one rack, and there are 3 racks in one section and a total of 24 different sections in a baseline scenario. This implies that, for a minimal setup, it provides the storage capacity for more than a thousand products (15*3*24) with 6 different characteristics and 6 different SKUs, excluding other warehouse areas e.g. expedition and receiving areas. It also provides the flexibility to be scaled to a larger set-up, which is possible by just adding new settings to *Belief Sets* of AOSR heuristics because the concept of *Knowledge Builder* and *Concept Builder* makes this system self-configurable. The time efficiency in a scaled scenario may be reduced a bit because of huge memory base but overall the accuracy would remain the same as it is based on AOSR's modular heuristics.

Belief-Builder holds the capability to check on the upcoming exceptions by utilising the *Check-Exceptions* function in certain situations. For example, sometimes a request for fetching data may not be successful because of network failure or an empty record tuple. In that case the dataset may hold no record-tuple, which may cause errors when performing operations through actuators, so Belief-Builder manages such issues before initiating other components as highlighted in Algorithm 1. Belief-Builder converts all the information from the *Belief-Base* into data threads to read through and compare every single data entry thoroughly, and uses *Thread-Reader()* to extract, analyse and combine them into different Belief Streams. For the initial configuration, the Belief-Builder builds the baseline beliefs for Products, Racks and their detailed Characteristics. Similarly, Knowledge-Builder is based on the same strategy to build the pool of knowledge-constructs and maintain a completely updated *Knowledge Base*.

The feature of sensing from the environment keeps AOSR updated at all times, which helps in planing actions in a timely manner. Algorithm 2 represents the heuristics of *Percept-Builder*, which is responsible for pooling percepts from the environment. It builds its beliefs and knowledge from *Belief-Builder* and *Knowledge Builder*, respectively. Based on knowledge threads related to products' locations within the warehouse, it builds a comprehensive Placement Plan (P), which keeps updating itself whenever a product-batch needs to be shipped or delivered. Two main entities dealing with AOSR-WMS are the SCM component from the back-end supply chain and CRM from the front-end. Both of these components send requests to WMS for any shipment and delivery, corresponding to certain product-batches, receptively. *Percept-Builder* utilises its method of *Request-Analyser()* to identify two of its variations, requests from ECU side and requests from CRM side, and perform tasks accordingly. Because AOSR algorithm is based on the BDI-agent model, it completely complies with the FIPA-Agent Communication Language (ACL) protocol [48]. All messages between different AOSR components follow ACL constructs. *ACLmessageReceiver()* function in Algorithm 2 extracts all the the subcomponents of the request and identifies the information related to product details, e.g. their SKUs, their characteristics or quantity. The method of *Extract-Characteristics()* fetches all the characteristics related to that particular product highlighted in ASN/ADN. This set of characteristics helps in finding the right match to determine a suitable rack to place the product within the warehouse at the right location.

If the request is from ECU, the very first step is to find a suitable rack, as explained in Algorithm 3. Other than matching the characteristics of products with that of racks, capacity is one of the main concerns in order to completely store the batch. In contrast to a standard WMS ([29,110]), AOSR provides an advanced and deeper approach to assign a rack to a product. It does not just randomly assign an available rack to a product but it analyses the list of available racks based on capacity and location and then attempts to consolidate the slot within the rack e.g. by justifying the maximum possible space to completely fill the same rack level, rather than putting the dispersed products into different levels. Although this is not always achievable because of capacity and quantity mismatch, it first tries and then finds the nearest possible rack in order to reduce the total activity-time within the warehouse. If the method of *SearchRack()* cannot find a suitable available rack, only then does it attempt to find an available Expedition Area (EA) while, in parallel, checking for any upcoming delivery orders from its *Knowledge Base*. If, by extracting the data log from *Knowledge Base* for ADN, it perceives that some products need to be delivered within a given threshold (threshold defined by the company e.g. 3-5 days), it initiates a task to re-slot the existing products from racks and put them into the available EA in the quantity specified in the ADNs and place the products coming through ASN into racks, so that they may stay in the rack without any further hassle. After this re-slotting on the delivery day (in 3-5 days), it picks the products from EA and ships them. Thus, through this re-slotting mechanism, the warehouse remains more organised and better managed.

If *Request-Analyser* receives a request from CRM, the *Request-Analyser* utilises the functionality of *RetrieveLocation()*, highlighted in Algorithm 4 (line *11*), and builds a list of possible locations for the required product and quantity. Similar to AOSR product-placement-strategy, location-retrieval-strategy also ensures it consolidates the racks by finding the minimum possible products to be fetched in order to clear a rack for upcoming products. If it is not possible to consolidate, it identifies the nearest possible location where the product can be picked in order to reduce the total activity time. The method of *RetrieveLocation()* returns a failure notification without crashing only if the required product is not in stock in the desired quantity.

Every location in an AOSR recommended-warehouse is given a unique name, so that it can be uniquely identified and easily managed without any ambiguities and hassle. Every location has a placement code that is comprised of rack number, rack level and characteristics (i.e. finished/raw, fast/slow and hazardous or not). The algorithmic overview of *Placement Generator()* is highlighted in Algorithm 5, which first identifies the available space using the heuristics of *SearchRack()* and then extracts all details from the *Knowledge Base*.

Recalling from Chapter 3, to implement agent-oriented heuristics, there are several tools available, such as Jack [70], Jadex [20] and JaCaMo [78], which provide advanced features such as implementing environment programming and emotion handling (but these are not the focus areas of this research). For testing purposes, we have explored the design mechanisms provided by available tools but the features provided by JADE (as detailed in Chapter 3) are much simpler and suitable for the AOSR strategy as compared to the aforementioned tools. Hence, all the algorithmic heuristics of AOSR have been implemented in JADE [81], which provides simplicity with flexibility to design multiple agents and facility of sniffer agent interfaces to monitor the overall agents' activity. Constraint based tests are applied to acquire results by applying AOSR strategy in contrast to a standard WMS approach (discussed in detail in Chapter 6, to see if the issues in warehouse management can be reduced by employing a moderate level semi-autonomous AOSR solution.

8.2.2 Results and Discussion

The prototype developed in JADE to validate AOSR strategy utilises the aforementioned test data sets in Table 8.1 and Table 8.2. The *6-Feature* strategy of AOSR [43] attempts to provide a solution to SMEs' problems in warehousing, such as wandering items/picking lists [23,60], inaccurate stock value at runtime [149], unmanaged receiving and expedition areas [156], unmanaged storage capacity [110] and inappropriate retrieval scheduling [105].

The AOSR strategy is validated using multiple test cases. In order to provide clarity and increase readability the test scenarios are subdivided as per two major validating parameters: **Performance Efficiency** and **Time Efficiency**, as detailed below.

Performance Efficiency of AOSR Strategy

The performance efficiency of AOSR strategy is tested based on three KPIs as highlighted in Section 8.1.1, in three different types of scenarios:

- Number of Products stored in Racks;
- Number of Products stuck at RA; and
- Number of Products placed in EA.

In order to provide a clear comparison of AOSR strategy with the other standard approaches, the experiments are performed by combining the standard zoning logic with two widely acceptable warehousing logics: *(i)* Zoning Logic with First In First Out (FIFO) Logic [86], which picks and puts the products with the preference of first arrival and *(ii)* Zoning Logic with Pick/Put from/to the Fewest Logic [150], which picks and places products with the preference of clearing the space. The test cases to validate the performance of AOSR are segregated further in two different states of the system: Initial Static State (*State (0)* and Regular Dynamic State (*State (1)*). *System State (0)* is a preliminary state where there are no products in the warehouse when shipment notices start to arrive for products to be shipped to the warehouse. *System State (1)* is a normal running-system state where there are already some products stored in the warehouse and both the ASNs and ADNs are being received for products to be shipped and delivered within the same time interval. All the these test cases are explained below.

Scenario of Products in Racks

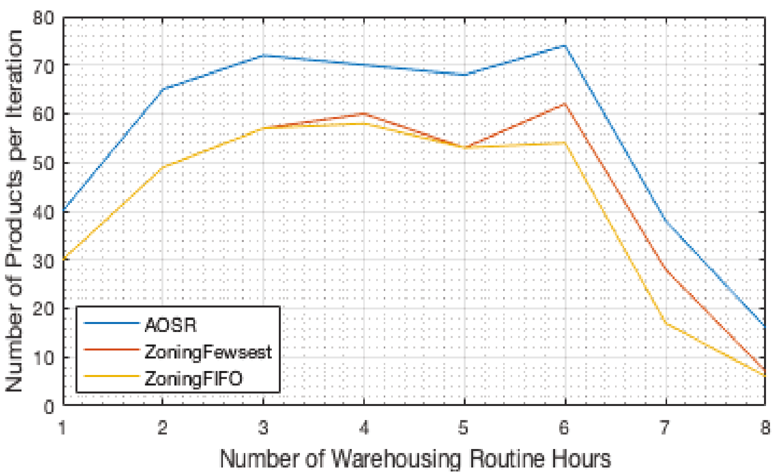


Figure 8.10: Comparison of Multiple Logics with AOSR for Products in Racks in State0

Figure 8.10 represents the results taken from *State 0 (Initial Static State)*. The results reflect the difference between these approaches for the applied test data set for a full routine day. The number of transaction/iterations are divided into hours (represented on the x-axis) and the number of products being shipped or delivered to warehouse (represented on the y-axis). The graph reflects that there is no major difference in the two standard warehousing logics, Zoning with FIFO Logic and Zoning with Fewest Logic. Although the results generated by AOSR represent the same pattern, the situation is better than the other two approaches as there is a higher number of products within the racks, which is considered as a performance metric for efficiency in warehousing [58]. AOSR follows the same trend as the other techniques (with better performance than the others because it follows the strategy to maintain a very low number of products in RA) as it uses the hybrid logic and is not offered a situation where its re-slotting strategy can be utilised since no delivery operations are performed. In the *System State (1)* the performance improvement of AOSR over the other techniques can be easily noticed in Figure 8.11.

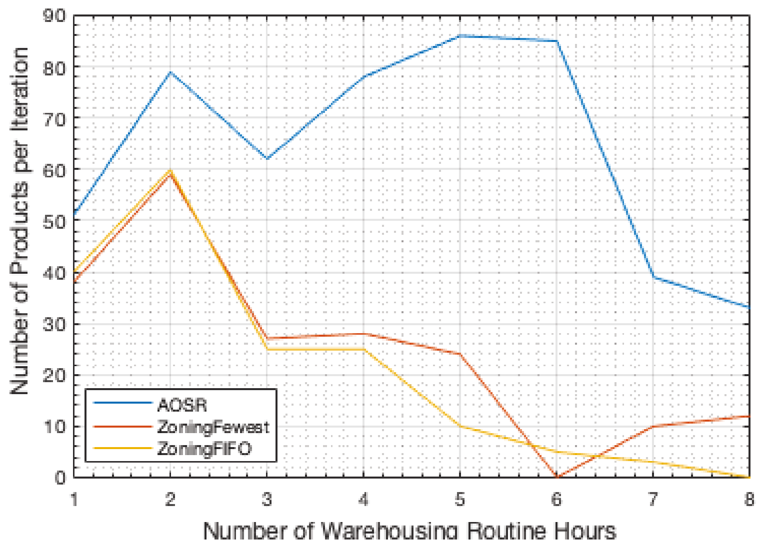


Figure 8.11: Comparison of Multiple Logics with AOSR for Products in Racks State1

Figure 8.11 provides a comparison of the two standard approaches with the hybrid strategy of AOSR. As can be seen from the graph, for the first 4 hours, the two standard approaches: Zoning with Fewest Logic and Zoning with FIFO logic have only marginal difference because both of the approaches follow the same pattern of leaving one quarter of the products in RA. However, the AOSR strategy is based on enterprise integration concepts of AOSF framework and takes the ASN/ADN into account prior to the arrival of products. Hence, the time taken to identify the products and find the right placement is already reduced because of the proactive nature of AOSR. After the fourth hour the performance gap and difference in trend can easily be noticed as AOSR utilises its re-slotting strategy to make space available for upcoming products in order to maintain the maximum number of products within racks. For the fifth hour a difference can be seen between the two other standard approaches; Fewest Logic performed comparatively better than FIFO logic as it tries to consolidate the space to make more availability for new products to be stored within the rack. As the AOSR strategy utilises a combination of these approaches, it is more successful and yields better results than the other two individually. A clear performance difference can be seen during the sixth hour, which ultimately reduces for the seventh and eighth hour as the number of total products is reduced in upcoming shipment and delivery notices.

Scenario of Products in RA

In a standard SC warehouse, a manual method of sorting the received products and identifying the proper location takes almost one quarter of the total time and operational effort [58]. The case of products in RA is different while utilising the AOSF framework and the *6-Feature Strategy* of AOSR, which recommends the BPR-based proactive approach of sensing the ASNs and ADNs and maintains a very low number of products in RA having prior knowledge of upcoming products. The results are shown in Figures 8.12 and 8.13 for *State 0* and *State 1* respectively.

A difference between the two standard approaches is slightly noticeable in Figure 8.12, particularly after the third hour. The 'Zoning with Fewest Logic' has taken more time in sorting and identifying a proper space for the products than the FIFO Logic, which is why it has detained more products in RA than the FIFO Logic. In the case of AOSR strategy there are very few or no products in RA because of its proactive approach, which lets the AOSR Planner Agent make prior plans, so it makes the RA clear for upcoming products for auto-identification [43].

The difference between the two standard approaches becomes unnoticeable in *State 1* as shown in Figure 8.13 up until the fifth hour. Before hour 5, there is more space available in the warehouse, so both approaches take less time and effort to identify

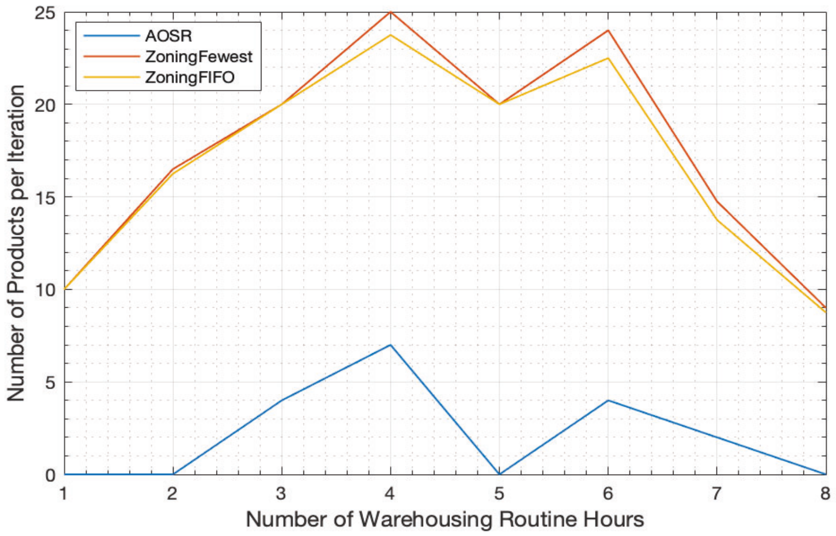


Figure 8.12: Comparison of Multiple Logics with AOSR for Products in RA State0

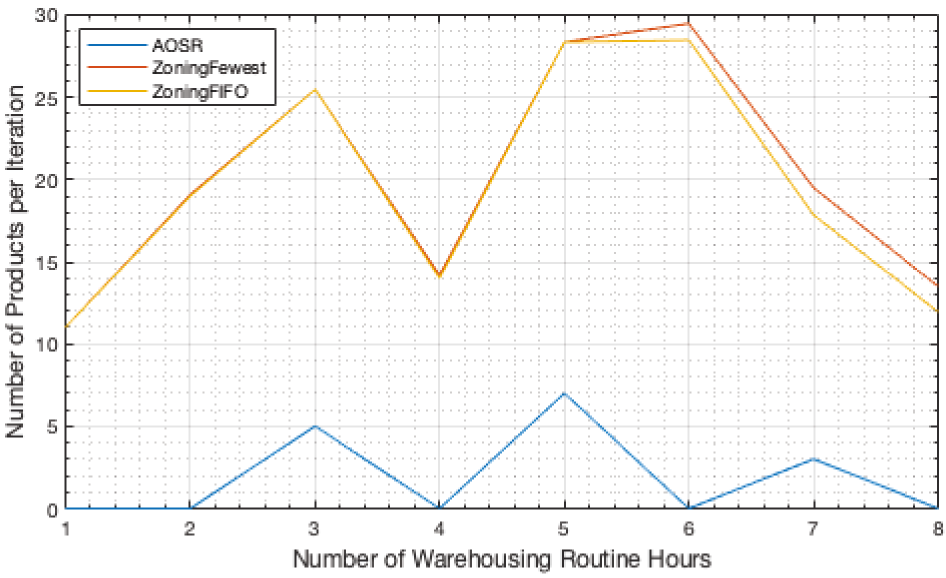


Figure 8.13: Comparison of Multiple Logics with AOSR for Products in RA in State1

the space. This is why there is almost the same number of products in RA in both cases. However, when the same products start repeating themselves in upcoming ASNs and ADNs after hour 5 then 'Zoning with the Fewest logic' takes more products to RA to identify the available space than the FIFO Logic. In this scenario, the AOSR recommended strategy takes the lead and provides better results by incorporating its cognitive and integrative approach to support the warehouse activities with a pro-active utilisation of its slotting and re-slotting capabilities.

Scenario of Products in EA

For the results acquired in the scenario of products in EA, there is a very slight difference in the scenarios of *State 0* and *State 1*, but they yield considerably different results. The results shown in Figure 8.14 represent almost no difference in both of the standard logic approaches and *AOSR Hybrid Logic* for first three hours as, in *State 0*, there are no ADNs, which means the products are only received at the warehouse but no product is being delivered.

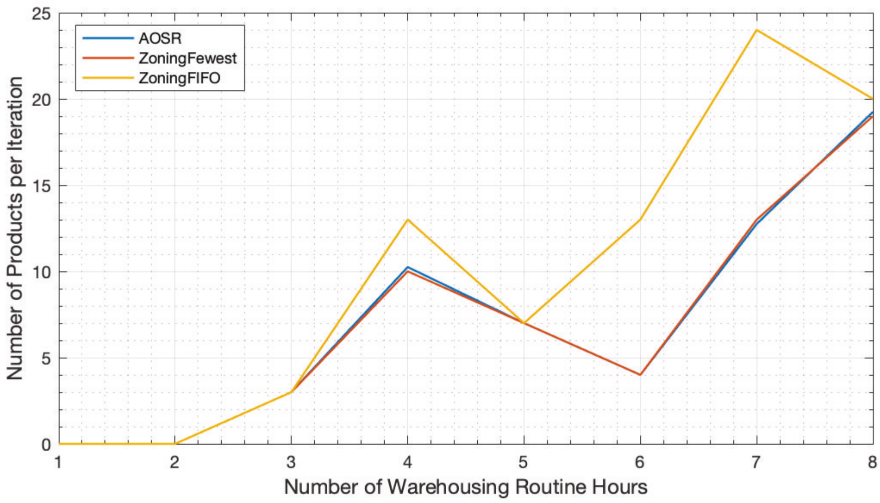


Figure 8.14: Comparison of Multiple Logics with AOSR for Products in EA in State 0

So, in that case, the AOSR planner algorithm has no opportunity to utilise its re-slotting strategy and shows almost the same pattern as the standard logics do. From hours 1 through 3, as there are no products in the racks, all three strategies can easily find the capacity to store products within racks, and the extra products that exceed the total capacity are placed in EA. The difference can be noticed in iterations after hour 3, as from hour 4 onwards, other than hour 5, the *Zoning with FIFO Logic* has placed more products in EA because of its failure to incorporate consolidation logic like *Zoning with Fewest* and AOSR do. For hour 5 the products appearing in ASNs have different categories to those already stored in the racks so the same number of products are placed in EA by all of the three strategies.

The results in *State 1*, as represented in Figure 8.15, clearly spell out the performance gap between the approaches. Because in *State 1* both the ASNs and ADNs are being received, the AOSR can utilise its re-slotting strategy when needed. In hours 1 and 2,

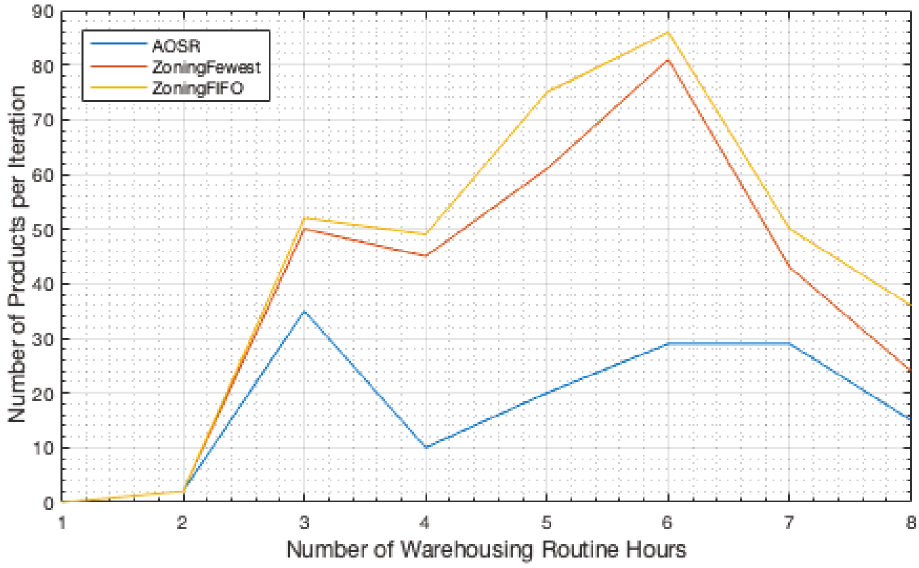


Figure 8.15: Comparison of Multiple Logics with AOSR for Products in EA

there is plenty of space and the products are easily stored within the racks, so the number of products is the same for all of the three strategies but after hour 2, when the number of products is higher than capacity and when products with the same characteristics appear in upcoming ASNs and ADNs in hour 4 then the difference between AOSR and standard strategies is quite visible. The AOSR has maintained a comparatively lower number of products in the EA throughout the routine hours for all the random test scenarios, which can help reduce the issues of wandering/lost items and unmanaged inventory.

Although the preference in the *Belief Base* of AOSR planner agent is to place a maximum number of products within the racks, when similar products arrive, so that the total is greater than the maximum capacity of the warehouse for that particular product, then it places them temporarily in EA. In parallel, it continuously checks with its *Knowledge Base* for any updates of products to be shipped so that it can place the new products into racks rather than EA and re-slot previous products to EA from the racks. Then, when the delivery date arrives for the re-slotted products, they can be picked from the EA and space can be cleared for future possibilities. This phenomenon can be observed during hours 3 and 4 in Figure 8.15, when AOSR places the products in EA because the number of products in ASN is much larger than the maximum capacity for that product-batch, so it re-slots the products from racks to EA and places the newly arriving products into racks so that they do not need to be moved later and inventory can be managed effectively. Also in hour 6, when products with the same characteristics arrive in ASN, both of the standard logics have placed a very high number of products in EA, while AOSR has placed the products from ADN in EA and upcoming products from ASNs placed in the racks. This is how AOSR's re-slotting strategy helps in maintaining a low number of products in EA.

Time Efficiency of AOSR Strategy

While validating the technical solutions, execution time is always an important factor to be considered. The purpose of testing the time efficiency is to rule out processing time as a bottleneck of the system. In order to evaluate the time efficiency of AOSR strategy, several test cases are applied. These test cases and scenarios are categorised as below:

- Gradually Reducing Search Space;
- Gradual Change in Product Characteristics; and
- Random Cases.

For all these scenarios, the results are acquired from both sides: the ECU side and CRM side. These results with their implications are discussed below.

Scenario of Gradually Reducing Search Space

AOSR maintains a balanced execution time to produce results as shown in Figure 8.16; all of the transactions, performed on the same machine and Operating System (OS): an Intel (R) Core (TM) i5 computer, having 3.7 GHz clock rate and 64-bit MacOS, took less than 0.02s, which reflects its efficiency with respect to time as well. This execution time is comparable with the other standard approaches validated and tested by Waris et al. [188], on a similar hardware configuration (Intel Core i5 with 64-bit OS), where the average execution time for parsing information, in a similar scenario, is 0.021s (This thesis does not include explicit test cases for validating the execution time for other approaches).

A closer look at Figure 8.16 can explain that, with the reduction in search space, it takes less time to compute and to generate results. The ECU component utilises *Percept-Builder* (highlighted in Algorithm 2) and *Actuator search-rack()* (in Algorithm

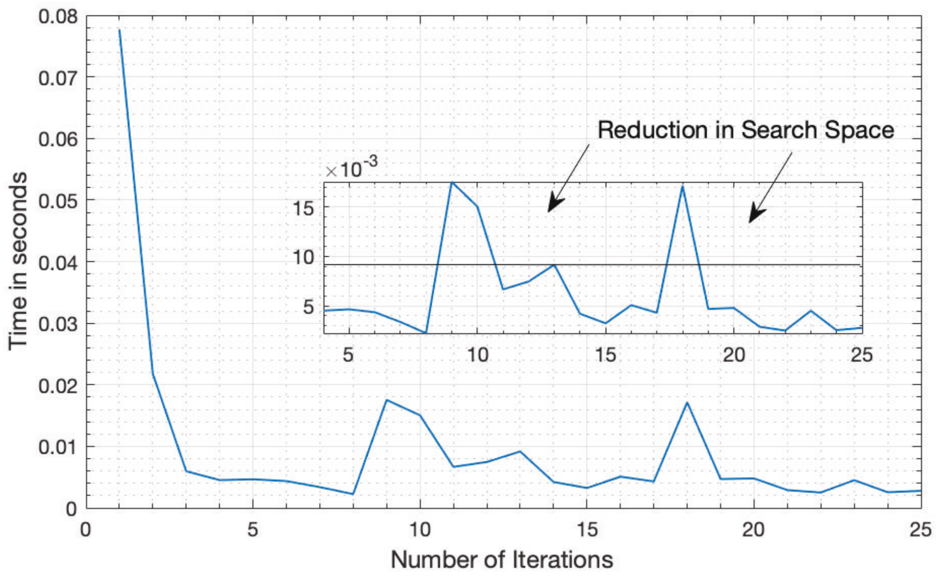


Figure 8.16: Efficiency of ECU side of AOSR Algorithm with Gradual Reducing Search Space

3), which employs caching techniques [13] for finding an appropriate rack for the upcoming products, such as it stores the memory base of similar previous results to compare it with the upcoming request and thus generate a quicker result. In case of changing the set of characteristics it builds a new search space. Hence, AOSR algorithm takes less time when it is in the same iteration and when it switches the iteration the time taken increases abruptly (but not more than 0.02s) and then again reduces gradually in the same iteration. This phenomenon can be observed at iterations 9 and 18.

Similarly, the CRM side of AOSR algorithm also takes the same strategy and almost the same trend in execution time as reflected in Figure 8.17. The time taken to execute CRM side transactions is a bit higher than the transaction time on the ECU side as it utilises a double iterative strategy, as highlighted in Algorithm 4 (line *9-22*), in order to update the inventory as well as the capacity in the stock. Even after performing almost double the number of tasks as its partner side, only one iteration took over 0.04 sec (in iteration 6) with most of them taking around 0.02s. Also the reduction in search space reduces the execution time as well, which can be seen between iteration 5 to 13, 14 to 22 and 23 to 30.

Figure 8.17: Efficiency of CRM side of AOSR Algorithm with Gradual Reducing Search Space

Scenario of Gradual Change in Product Characteristics

The test cases to validate the AOSR strategy include 20 different classifications of products presented in Table 8.2, so a gradual change in characteristics results in a gradual decrease in execution time. The results taken after applying the test case with gradual change in product characteristics are represented in Figure 8.18. *Percept Builder()* (in Algorithm - 2 line *11-17*) attempts to find the local optimal for every product characteristic and, in case of a change in characteristics, it exits the loop and attempts to build a new cache and starts searching for the optimal value again.

As reflected in Figure 8.18, all the iterations take less than 0.01s which is a great execution time for an algorithm like AOSR, which interacts with the environment and computes the plan for the whole warehouse shop-floor. A closer look can explain that even at iterations 5, 14 and 23, when the characteristics change, it took less then 0.01s and all the corresponding iterations were completed within about 0.005s.

Similarly, while executing the test case on the CRM side, as represented in Figure 8.19, the gradual change in characteristics reduces the execution time but leaving an iteration and initiating new caching memory takes a bit more time because of its double iterative strategy. Even when building its memory base, even the first iteration takes less than 0.06s, with all other iterations where the characteristics change having an execution time around 0.03s. All the other iterations take less than 0.02s, which demonstrates the

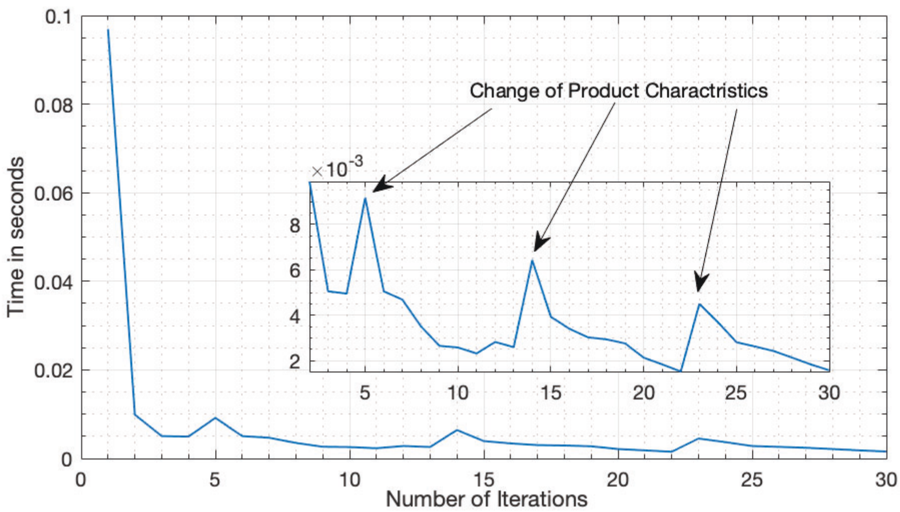


Figure 8.18: Efficiency of ECU side of AOSR Algorithm with Gradual Change in Characteristics

consistency of the overall AOSR strategy.

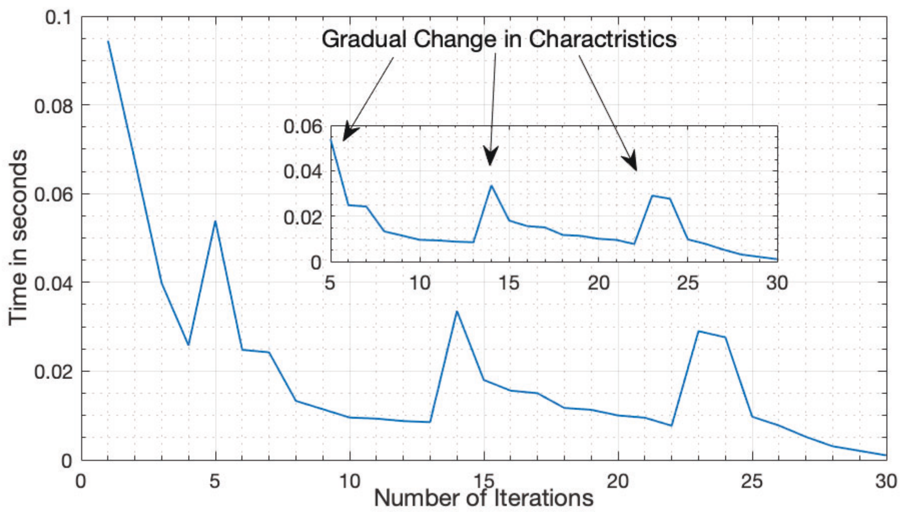


Figure 8.19: Efficiency of CRM side of AOSR Algorithm with Gradual Change in Characteristics

Scenario of Random Cases

For a complete validation, a set of 25 random test cases (with non-sequential product characteristics) is applied to AOSR and performance is seen to be consistent. These test cases include the random data from already existing industrial data set (as used in cases mentioned above) to ensure the non-sequentiality of iterations. The extracted results for

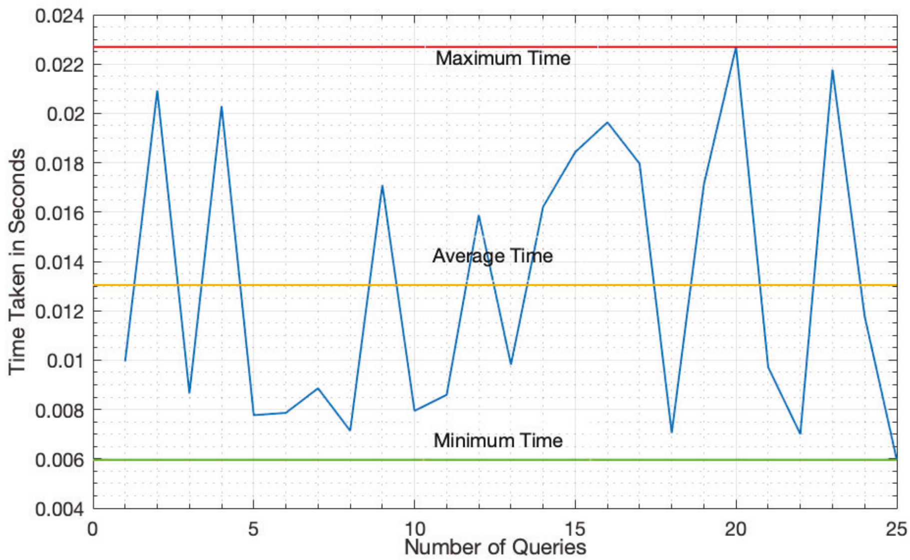


Figure 8.20: Time Taken AOSR with Random Test Cases

the ECU side are shown in Figure 8.20, where the maximum time taken to extract the right information and activate the actuator is 0.022s. This demonstrates the efficiency and consistency of AOSR, even in random test cases. The least taken time is 0.006s, which is finding and allocating a space to a product where there is no product already stored, so the process remains quite simple and quick. On average all transactions took about 0.013s to compute the comprehensive product placement plan.

Similarly, the results extracted from the CRM side by applying the random test cases are reflected in Figure 8.21. In the case of unavailability of space for a certain product, the AOSR utilises its re-slotting strategy where computation is then performed three times (to pool the information from ASN/ADN, re-slotting and then slotting the products if needed) by Algorithm 4 to check and manage if there is a need to re-slot the products. In order to manage the space with efficiency, the CRM side of AOSR algorithm takes more time than the ECU side because it includes searching and updating both the racks and the inventory level. The maximum time taken by the CRM side of AOSR was 0.09s, which is not a very high computation time for the algorithm to build, extract and perform transactional and analytical information. On the CRM side when the required product is in the nearest rack with available matching quantity the time taken is less than 0.02s as shown at iteration 20 in the graph. On average, the CRM side of AOSR took 0.047s to perform the task to satisfy requirements. In literature, several other re-

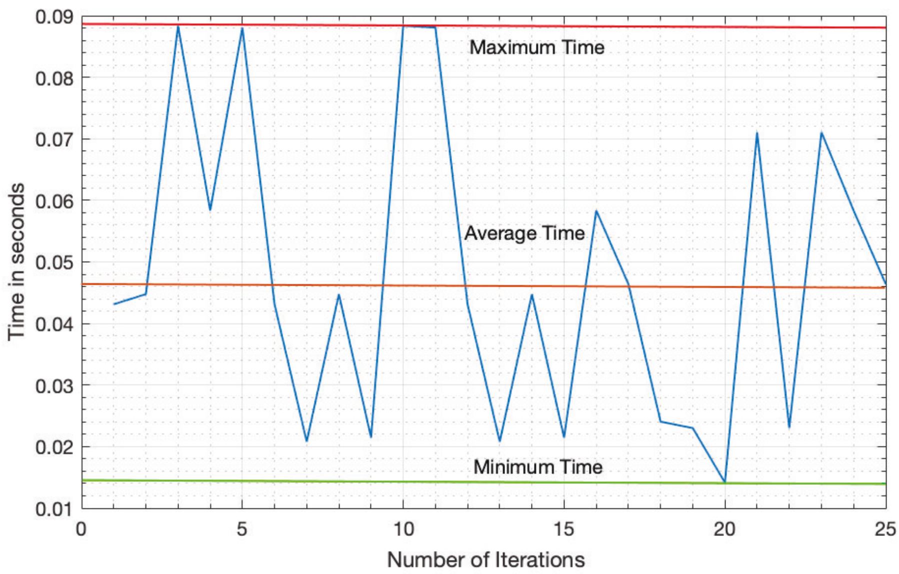


Figure 8.21: Time Taken AOSR with Random Test Cases on CRM side

searchers have computed the standard computation times on Pentium series computers for similar heuristics-based scenarios but in a different context of location allocation for vehicle-routing problems, and the average CPU time lies between 0.52s to 8.57s [185] or 0.62s to 10.21s [195] with different approaches. Research suggests that, on average, less than or equal to one second ($\leq$ 1s) is considered a standard CPU processing time for inbound logistics [165], regardless of hardware configuration. The computation time taken by AOSR strategy falls well inside these limits. Thus, the efficiency and hybrid approach of AOSR makes it suitable for industry where agility and customisation are the main metrics of success.

Other than the validation of this system in JADE, we have also validated this system using real data. For this purpose, we implemented the AOSR strategy in an industrial simulation tool, Demo3D [38]. Demo3D is a product of RockWell Automation [158], which is a US-based firm providing industrial automation solutions. We created a 3D visualisation of a warehouse with AOSR strategy and received comparable results to those discussed in earlier sections. These results are not included as part of this dissertation because of privacy concerns of the organisation's data where it was implemented.

In this chapter, we have discussed results from several different types of test cases to thoroughly validate the AOSR strategy. The test cases analysed the products stored in racks, in EA and in RA, each with respect to two different system states: *State 0* (without conflicts) and *State 1* (with possible conflicts). We also discussed the time efficiency of AOSR strategy in relation to the standard approaches. The successful and positive results, from all the scenarios and test cases, highlight the overall performance efficiency of AOSR algorithm in association with its parent AOSF framework. The next chapter will provide a summary and conclusion for the contribution made by this research, with respect to the research questions identified at the beginning of this thesis.

Chapter 9

Conclusion

The inception of Industry 4.0 in recent years has given a boost to current manufacturing setups in industry, though SMEs are not getting the expected benefits from these standards [5, 8, 119, 133, 151]. With an objective to bridge this gap, this research presents the framework of the Agent-Oriented Smart Factory (AOSF) and its associated Agent-Oriented Storage and Retrieval (AOSR) WMS strategy. The study conducted and the solution presented in this thesis, with the help of a number of peer-reviewed publications, attempts to provide a comprehensive SC architecture for SMEs incorporating CPS-based Industry 4.0 concepts and reduce the common baseline issues in warehouse management.

In order to present a moderate level semi-autonomous solution, this research aimed to provide a two-fold solution for SMEs, including a comprehensive SC architecture and an agent-oriented warehousing strategy. Based on the concepts highlighted in Chapters 1 and 2, it proposed the AOSF framework to map Industry 4.0 standard to SMEs' SC networks in Chapter 4 and the AOSR strategy in Chapter 6 to reduce the highlighted issues in warehouse management. Chapter 5 presented the problem and domain definition for prototyping AOSR strategy. Chapter 3 included details of the selection of a prototyping methodology and development strategy that gives the benefit of a sound agent communication environment to develop and test the proposed system. Chapter 8 provided a detailed validation of the AOSR algorithmic heuristics and presented results in comparison to the standard warehousing strategies.

The importance of the AOSF framework and its associated AOSR WMS strategy can be justified from the point of view of the performance difference and potential benefits that it offers toward the implementation of Industry 4.0, specifically for SMEs. Some of the advantages of this proposed system are as follows:

Comprehensiveness: AOSF framework provides a comprehensive architecture of SC implementation from both ends (the back-end supplier side and front-end customer side). This architecture does not only provide CPS based Industry 4.0 implementation guidelines but also three-dimensional integration across the enterprise to keep all units integrated, interactive and updated all the times.

Flexibility and Customisability: Based on Industry 4.0 standards, the AOSR strategy, which is the implementation of the AOSF framework towards SME-warehouses, provides the flexibility to be customised as per the enterprise setup criteria. It provides flexibility in design, configuration and hierarchy with agent negotiation and coordination strategy. It also provides the flexibility of defining the zones in a warehouse and the characteristics of products in the belief base of agents, which can further be utilised while performing operations.

Efficient, Quick and Systematic: The systematic algorithmic heuristics of AOSR provide fast computational capabilities with efficacy, which makes it better than manual human-operated planning and processing of warehouse operations. With the help of a continuously updating knowledge base and cross-checking with its belief base, AOSR helps in increasing the overall efficiency in the SC network and the warehouse shop-floor.

Reduction in Costs: AOSF framework, with its associated AOSR WMS strategy, provides the potential to reduce infrastructural costs such as installing conveyor belts and investing in high-tech robots in the manufacturing SME industry, which can result in significant cost reduction, specifically in warehouses. The implementation of this proposed system is based on CPS; hence it can play a vital role not only to reduce the issues in warehousing but also in exposing SMEs towards the benefits of Industry 4.0. This framework provides the options of scalability to incorporate Big Data and Cloud Computing in future.

Applicability: Comprehensiveness, flexibility and customisability of AOSF framework make it suitable for the dynamic environment of SMEs, as AOSF's tiered architecture provides a basic layout which can be adopted step by step by adding more layers at the lower level, and provides an option to grow with time without any restructuring of the scalable top layers. The systematic approach of the associated AOSR algorithm also provides a moderate level, semi-autonomous, low-cost solution to apply the business needs as is, rather than moulding the company's business process according to the system requirements. The AOSR algorithm provides customisable settings to be altered as per business need, allowing the solution to fit into an SME-environment.

In the remainder of this chapter, Section 9.1 concludes the work by revisiting the research questions identified in Chapter 1. This is followed by some possible dimensions for future work in Section 9.2

9.1 Revisiting Research Questions

This section throws light on how the research questions, highlighted in Chapter 1, have been handled by the research conducted in this thesis.

9.1.1 Research Question 1

- *Which framework under the umbrella of Industry 4.0 is compatible with SMEs to improve the issues of warehousing? [5, 8, 119, 133, 151]* Does Industry 4.0 standard focus SMEs explicitly?

Recent research claims that the Industry 4.0 standard mainly focuses on large industrial setups [8, 151] and only marginally on SMEs [164]. Several researchers have highlighted the issue of not having a compatible Industry 4.0 framework applicable for SMEs, such as the study conducted by Müller et al. [133], which presents a survey held in 2018 for 68 SMEs in the world's 2nd largest industrial hub, Germany. A similar claim is made by Andulkar et al. in 2018 [5]: that such an approach can possibly endanger the business model of SMEs. Hence, for widespread implementation of industry 4.0, its applicability in SMEs is a crucial aspect [119].

In order to minimise the chances of SMEs becoming victims of Industry 4.0 rather than beneficiaries, the identification of its compatibility with semi-autonomous systems in a timely way is important. The system presented in this thesis provides an overarching solution to this problem by contributing not only a CPS-based end-to-end SC architecture, which incorporates both ends: the back-end SCM side and the front-end CRM side; but also provides the three-dimensional enterprise integration strategy to keep all the sub-systems integrated. The details of the AOSF framework with its integration strategy are presented in Chapter 4. The three-tier architecture of the AOSF framework provides the flexibility of incorporating other units in the future, e.g. procurement, sales and accounts department. It also includes the classification of multiple agents and their negotiation and communication mechanisms in order to provide more rationality and embedded intelligence within the system. The details of its associated warehousing strategy are described in Chapter 6.

9.1.2 Research Question 2

- *Can MAS technologies provide Industry 4.0 benefits to SMEs? [95, 111, 183]* What MAS solutions exits under the umbrella of Industry 4.0 to facilitate the standard? What is the gap that still needs to be filled and what kind of solution can help bridge this gap?

Despite extensive research in the industrial applicability of agents and their level of rationality and intelligence with flexibility, a widespread breakthrough is not yet recorded, where they can contribute a high level of support in Industry 4.0 and CPS-based setups [95]. There is a broad potential to work on MAS based implementations under the umbrella of Industry 4.0 [111, 183]. Some researchers have presented basics from the perspective of intelligent and autonomous industrial systems, including but not limited to the works presented in [52, 89, 167, 168, 184, 189]. Prior research has advanced efforts to provide complete autonomous systems, but none of the works focused in depth on the implementation of an agent-oriented smart factory for Small to Medium Size Enterprises (SMEs) [162, 169, 175].

With the objective of providing a moderate level semi-autonomous but flexible and dynamic solution, the AOSF framework is based on MAS strategy, which provides a robust and ubiquitous environment with self-learning features to update the belief sets and knowledge base of multiple agents. Agent communication and negotiation strategy, which is based on the FIPA-ACL protocol, helps in utilising resources in a better manner. From the categorisation of agents detailed in Chapter 4, mediator agents help maintain the right balance in requests coming from client agents (either software agents or device agents) and available resources.

Chapter 5 provides a detailed definition of the problem and domain for the constituent agents within the system, which presents BNF-grammar-based MA-HTN constructs to define the main tasks utilised by the AOSF agents including non-primitive tasks and their distribution to further actionable primitive tasks. This problem and domain definition is utilised by AOSR-planner agent to resolve the problem of warehouse management (addressed in detail in Chapter 6).

9.1.3 Research Question 3

- *How can issues of warehousing be resolved for SMEs using Industry 4.0? [60, 110, 149, 156]* If SMEs cannot afford the high-tech robo-oriented warehousing system then how the problems of warehouse management can be resolved? Are the existing standard WMS strategies suffice the management requirement of SMEs oriented warehouses?

Previous research presents multiple projects, particularly from a warehouse perspective, such as the work mentioned in [96], related to a hybrid control mechanisms with an architecture of a specific warehouse, which is similar to the ones presented in [52] and [89] related to conveyor belt systems. There are several other warehouse optimisation implementations in the literature, including Flowshop Algorithm [28] and Ensemble Multi-Objective Biography Based Optimisation [114], which address the automation up to the next level where autonomous robots are moving in the picking aisles to pick and place products into the racks. Even utilising the existing standard warehousing strategies, many warehousing issues still persist, such as wandering items/picking lists [60, 149], inaccurate current stock values at run time [149], unmanaged receiving and expedition areas [156], unmanaged storage capacity [110] and inappropriate retrieval scheduling [105, 149].

This thesis presents the concept of the Agent-Oriented Storage and Retrieval (AOSR), associated with its parent AOSF framework. AOSR's recommended *6-Feature* strategy (detailed in Chapter 6) attempts to reduce the aforementioned issues by providing a hybrid strategy for product placement and retrieval. Based on a CPS architecture, it provides overall pro-activeness by sensing the upcoming information from the environment. The AOSR mechanism presents the idea of slotting and re-slotting based on run-time need, and its recommended planner agent builds, maintains and updates the placement of products within racks and expedition areas. With the information pooling and three-dimensional integration system, it aims to reduce the number of products in the receiving area and attempts to maximise the number of products within racks. The general set of experiments and test cases applied to AOSF mechanism, and its comparison with standard WMS strategies clarifies the performance difference; that a hybrid logic-based mechanism helps improve the overall management within the warehouse shopfloor. The results (presented in Chapter 8) suggest a great improvement, specifically in increasing the number of products within racks and reducing the numbers both in receiving and expedition areas, which helps in reducing the hindrances in managing warehousing activities.

A wide variety of real-world experimentation of this research, utilising several different test cases and scenarios, justifies the applicability of this scalable system. The flexibility and volatility of AOSF and AOSR strategy provide a fair margin for customisability to adapt to SME-oriented setups. The comprehensiveness offered by this system does not only provide the overall guideline for SC framework but also a specific WMS strategy to reduce the baseline issues which cause further problems in managing a warehouse in semi-automated structure.

9.2 Future Directions

The framework presented in this thesis provides a support mechanism to expose SMEs towards the benefits of Industry 4.0. However, it is impossible for a single solution to be universally applicable. Similarly, the presented system also has some limitations. For example, performance has only been tested in the prototype, not in a real-time distributed cloud architecture, where results may vary slightly. Furthermore, this system does not include in-built cloud server security, which is another rich area of research. Although the AOSF/AOSR framework caters to the requests coming from smart-devices, connecting manual industrial hardware components to this system may raise some more areas of optimisation. There is always a margin for future work in almost every solution; some of the possible future extensions of this project are also identified and discussed in upcoming sections.

In order to provide more robustness within the system, some features from the other state of the art concepts of Big Data and Cloud Computing can also be incorporated.

Although, the AOSF framework includes its own hybrid planning strategy for its associated AOSR algorithm, pure multi-agent planning strategies such as TALplanning [101], IXTET-EXEC [104] or DOMAP [26] can also be implemented as an alternative in future. Some of the other possible future works for this research are highlighted below:

9.2.1 Cloud Architecture for Multiple Sites

This thesis specifically addresses the need for having a compatible CPS-based Industry 4.0 framework that can support SMEs. On the other hand, for distributed industrial setups, the idea of maintaining a cloud network becomes more important (which was not the focus area for this thesis and comes with the concerns of security and privacy). For maintaining distributed enterprise setups, the recommended model of the AOSF framework can still be utilised for inter-enterprise integration as it includes OLAP based systems and server architectures on the cloud layer. The AOSF framework includes a complete top layer for this purpose, but could be scaled further to cater for the concerns of privacy and security. The purpose of this thesis was to determine the feasibility of applying Industry 4.0 based architecture to SMEs with better management at the warehouse level, which is achieved by performing test cases and scenarios on a single site, which can be scaled to cater to requirements for multiple sites. In that case, the top two layers of the AOSF framework would remain the same, as detailed in Chapter 4, with the addition of one or more smart connection layers as highlighted in the test scenario in Chapter 8.

9.2.2 Big Data Analytics

Further to incorporating a scaled Cloud Network, utilising features of Big Data may provide this system with more cognitive abilities in order to provide intelligence, based on past data trends. Although most SMEs do not currently consider data as a source of added value [17], it could be a valuable addition in the future. The AOSF framework presents CPS-based provision for storing and maintaining historical data, for the purpose of predicting future trends and providing flexibility to incorporate data analytics in future. The ideas contributed by Voss et al. in [181] related to incorporating Big Data analytics in logistics can also be a part of this system to enhance it for future purposes.

Similarly, handling tasks with the same priority can also be a value addition to the work presented in this thesis. Chapter 6 provides an overview of the AOSR-algorithmic heuristics and details in Chapter 8 how the knowledge extraction feature of AOSF-agents helps in building, maintaining and updating the belief base and knowledge base of agents.

9.2.3 Addition to FIPA-ACL

The system prototype presented in this thesis complies with FIPA-ACL protocol, which provides the standard communication language for messages exchanged between agents. It utilises the basic ACL standard for agent interaction, though some recent protocols could also be added in this regard to provide an extra feature to the system for managing emergency situations in production systems, such as the work conducted by Hassan and Hun-Heh in [63]. This thesis mainly focuses on reducing warehousing issues, though Chapter 4 includes the details of smart connection for adding plant side in this system in the future. Also, in addition to basic FIPA-ACL functionality, the AOSF framework with its associated AOSR algorithm can be implemented with other tools, such as JaCaMo, to experience the features of environment and organisation programming with agents.

9.2.4 Corporate Cutting Edge Features

The experiments performed and discussion of results in Chapter 4, 6 and 8 help assure that the AOSF framework and its associated AOSR strategy work in reducing the issue of warehouse management by reducing the number of products in EA and RA and maximising the possibility of products being stored in racks by utilising its recommended *6-Feature Strategy* and its hybrid logic-based mechanism. There could be some more dimensions to work upon in the future such as movement within the warehouse shopfloor using forklift trucks, utilising collapsible racks or small-scale drones (as some of the industry is already providing a high-tech robo-oriented solution such as GrayOrange [186] and Unleashed [187]). These solutions provide nice cutting edge features but come with an additional infrastructure cost. The solution presented in this thesis can also be used for incremental improvements such as by employing this system, basic SC and warehouse management issues can be reduced; and later on, if needed, the automated features such as conveyor belts and picking machines can also be added into the system. AOSF framework provides a generic and dynamic solution, which can be customised in future as required.

9.2.5 Industry Interests

As discussed in Chapter 8, we have implemented AOSR strategy in Demo3D [38] in liaison with a local industry, Glenvern Associates [56], which offers consulting services to a diverse industrial clients from Australia, South East Asia and North America. Glenvern's management has shown great interest in the AOSR strategy and we are in the process of getting more involved with them in offering this solution to a range of their clients.

Appendices - A - Overview of Algorithms

These appendices (Appendix *A1*-*A5*) include the overview of Algorithms 1 - 5 as discussed in Chapter 6. The algorithm in Appendix *A1* presents an overview of Belief Builder to initiate/update the belief-base for AOSF agents. Appendix *A2* includes the algorithmic overview of Percept-Builder for ECU side, which provides heuristics to sense from the environment and create respective notifications. Algorithm 3 in Appendix *A3* provides overview of Search-Rack process to match the product quantities with respect to available space within the warehouse shop-floor. Algorithm 4 in Appendix *A4* is about dealing with the percepts generated from CRM side. Appendix *A5* includes Algorithm 5 which provides an overview of Product-Placement-Generation based on available space. All these algorithm in these appendices work integrated to carry-out the whole process for AOSF-AOSR strategy.

A.1 Algorithm 1

Algorithm 1 Overview of Belief Builder Heuristics

```
procedure BELIEF-BUILDER
    CheckExceptions()
    Inititate ThreadReader()
top:
    if updateThreadReceived then
        UpdateBeliefBase()
    end if
    BeliefStream[] ← extractBeliefBase()
do:
    BeliefThreadProduct[i] ← BeliefStreamProduct[i]
    BeliefThreadCharac[i] ← BeliefStreamCharac[i]
    BeliefThreadRack[i] ← BeliefStreamRack[i]
while (BeliefStream.hasNext())
    CheckUpdates
    if UpdatesAvailable then
        goto top
    end if
end procedure
```

A.2 Algorithm 2

Algorithm 2 Overview of Percept Builder-ECU Heuristics

```
procedure REQUEST-ANALYSER-ECU
    Initiate BeliefBuilder()
    Initiate KnowledgeBuilder()
    PlacementPlan[] ← KnowledgeThreadPlan[]
    request ← ACLmessageReceiver()
    P ← request.requiredProduct
    Q ← request.requiredQuantity
    C[] ← Extract-Charactristics(P)
    if request is from ECU then
        AvailableRacks[] ← Search-Rack(P , c[] , Q , Plan[])
        if true (AvailableRacks[]) then
            FewestAvailableRacks[] ← FindFewest(AvailRacks[])
            NearestAvailableRack ← FindNearest(FewestAvailableRacks[])
            GeneratePlacement(P, Q, NearestAvailableRack)
            UpdateBeliefBuilder()
            UpdateKnowledgeBuilder()
            Notification-Generator(SUCCESS, ECU)
        end if
        if  true(AvailableEA()) then
            if  true(CheckReslottingNeed()) then
                p ← ExtractADNlogProduct()
                q ← ExtractADNlogQuanitity()
                GeneratePlacement(p, q, EA)
                GeneratePlacement(p, q, ExtractPlacement(P, Q)
                UpdateBeliefBuilder()
                UpdateKnowledgeBuilder()
                NotificationGenerator(SUCCESS, ECU)
            end if
        end if
        else
            NotificationGenerator(FAILURE, ECU)
    end if
    CheckUpdates.
end procedure
```

A.3 Algorithm 3

Algorithm 3 Overview of Actuator-SearchRack Heuristics

```
procedure ACTUATOR-SEARCHRACK
    Initiate Belief-Builder()
    Initiate Knowledge-Builder()
foreach P in ASN:
    matchCharactristics(KnowledgeThread(Rack, P))
    matchCapcity(KnowledgeThread(Rack, P))
    if matched then
        AvailableRacks[] ← KnowledgeThread(RackNo)
        AvailableRackLevels[] ← KnowledgeThread(RackLevel)
    end if
    goto loop
    CheckConsolidation(AvailableRacks[], AvailableRackLevels[]))
    if consolidation possible then
        AvailableRacks[] ← fewest(AvailableRacks[], AvailableRackLevels[]))
    else
        AvailableRack ← nearest(AvailableRacks[], AvailableRackLevels[]))
    end if
    return AvailableRack
end procedure
```

A.4 Algorithm 4

Algorithm 4 Overview of Percept Builder-CRM Heuristics

```
procedure Request-Analyser-CRM
    Initiate BeliefBuilder()
    Initiate KnowledgeBuilder()
    PlacementPlan[] ← KnowledgeThreadPlan[]
    request ← ACLmessageReceiver()
    P ← request.requiredProduct
    Q ← request.requiredQuantity
    C[] ← Extract-Charactristics(P)
    if request is from CRM then
        if true P with Q inStock then
            PossibleLocations[] ← RetrieveLocation(C[], PlacementPlan[])
            FewestAvailable[] ← FindFewest(PossibleLocations[])
            NearestAvailable ← FindNearest(FewestAvailable[])
            ExtractPlacement(P, Q, NearestAvailable)
            UpdateBeliefBuilder()
            UpdateKnowledgeBuilder()
            NotificationGenerator(SUCCESS, CRM)
        else
            NotificationGenerator(FAILURE, CRM)
        end if
    end if
    CheckUpdates
end procedure
```

A.5 Algorithm 5

Algorithm 5 Overview of Actuator-PlacementGen Heuristics

```
procedure ACTUATOR-PLACEMENTGEN
    AvailableRack ← SearchRack()
    if matched in KnowledgeBase then
        RackNo ← KnowledgeThreadRackNo(AvailableRack)
        RackLevel ← KnowledgeThreadRackLevel(AvailableRack)
    end if
    Characteristics ← ExtractCharac(BeliefThreads(P.Charac))
    Location ← GenerateLocation(Characteristics, RackLevel, RackNo)
    return Location
end procedure
```

Appendices - B - Dataset with Random Variation

These Appendices (Appendix *B1* and *B2*) include the data used for performing extra validation with aggregated results. The data presented in Appendix *B1* represents the number of products stored in Rack, in EA and in RA using both of the subject strategies (Linear SC with Standard WMS Strategy and AOSF with AOSR Strategy). The data presented in Appendix *B2* includes the actual number of products used in different cases (case 1-30). This data is used in the experiments explained in Section 8.1.3.

B.1 Aggregated Results with Random Variation in Data

	Linear SC with Standard WMS			AOSF with AOSR		
Case #	In Rack	In RA	In EA	In Rack	In RA	In EA
1	517	129	379	824	71	185
2	478	120	483	765	76	239
3	489	122	469	787	65	228
4	523	131	426	834	34	212
5	504	126	450	812	45	223
6	545	136	399	839	43	198
7	467	117	496	786	49	245
8	509	127	444	812	43	225
9	498	125	458	820	30	230
10	545	136	399	865	23	192
11	489	122	469	790	62	228
12	459	115	506	767	68	245
13	508	127	445	823	39	218
14	515	129	436	838	30	212
15	544	136	400	855	33	192
16	523	131	426	834	41	205
17	534	134	413	847	39	194
18	546	137	398	867	22	191
19	489	122	469	769	93	218
20	503	126	451	810	41	229
21	476	119	485	767	82	231
22	489	122	469	799	53	228
23	498	125	458	778	84	218
24	509	127	444	823	46	211
25	507	127	446	816	68	196
26	512	128	440	835	41	204
27	534	134	413	856	31	193
28	508	127	445	802	66	212
29	498	125	458	785	75	220
30	504	126	450	809	55	216
Average	507	127	444	814	52	215

B.2 Data used for Test Cases with Random Values

		Quantities used in Different Cases									
Sr. No	Product ID	Case1	Case2	Case3	Case4	Case5	Case6	Case7	Case8	Case9	Case10
1	P-9001	5	22	14	29	34	12	5	21	16	34
2	P-9002	39	14	26	19	43	26	19	34	19	30
3	k-9804	42	29	34	47	23	19	23	19	5	12
4	K-2098	23	34	19	35	42	24	39	25	39	36
5	L-3092	34	20	32	12	34	54	55	39	12	21
6	K-9803	67	31	32	45	39	56	43	29	37	43
7	F-9210	23	29	20	33	21	15	58	47	39	28
8	L-2801	5	56	43	29	36	49	33	59	46	35
9	F-2830	34	37	28	38	16	32	45	29	43	18
10	C-3921	56	20	42	49	55	53	26	43	20	66
11	R-3392	39	55	19	22	31	46	54	34	43	37
12	R-1292	39	36	38	43	19	43	35	22	38	42
13	P-8372	44	34	41	34	59	23	62	40	55	28
14	K-3269	63	56	22	50	43	30	19	35	19	45
15	R-3390	20	62	59	45	69	55	61	48	50	59
16	P-9001	18	15	33	39	20	31	43	26	39	27
17	P-9002	2	25	29	57	51	48	36	51	22	40
18	k-9804	49	45	36	20	33	23	29	43	14	11
19	K-2098	71	33	44	35	41	50	47	25	42	55
20	L-3092	53	52	20	43	35	21	30	58	41	38
21	K-9803	21	20	59	31	19	36	14	49	33	47
22	F-9210	54	46	68	55	44	58	42	37	63	50
23	L-2801	43	51	49	20	11	24	30	22	39	31
24	F-2830	37	36	21	33	46	19	47	41	51	42
25	C-3921	28	44	33	26	45	38	18	19	26	22
26	R-3392	54	32	56	44	37	41	14	49	63	49
27	R-1292	31	39	34	20	26	39	26	16	37	27
28	P-8372	7	25	18	31	12	17	19	22	25	18
29	K-3269	43	29	28	22	31	23	37	31	19	34
30	R-3390	12	5	29	34	21	19	22	18	37	18
31	P-9001	5	19	33	27	18	25	34	22	18	21
32	P-9002	19	29	21	13	26	31	15	27	30	16

Figure 1: Product Quantities Used for Test Cases 1-10

		Quantities used in Different Cases									
Sr. No	Product ID	Case11	Case12	Case13	Case14	Case15	Case16	Case17	Case18	Case19	Case20
1	P-9001	13	9	25	11	20	16	19	26	10	21
2	P-9002	27	18	18	23	33	26	34	14	18	19
3	k-9804	19	31	20	34	21	19	12	28	9	23
4	K-2098	32	23	31	41	25	20	26	21	26	31
5	L-3092	20	16	27	19	38	54	41	39	29	16
6	K-9803	67	31	30	39	39	56	39	29	47	38
7	F-9210	33	29	21	41	41	15	44	47	31	23
8	L-2801	18	39	39	35	36	49	51	52	28	35
9	F-2830	39	37	44	38	29	32	45	29	51	18
10	C-3921	44	41	42	50	48	53	26	43	39	49
11	R-3392	31	55	26	22	31	46	54	34	31	37
12	R-1292	23	36	38	43	51	43	35	22	38	42
13	P-8372	44	28	41	34	42	23	62	50	28	28
14	K-3269	63	56	41	50	36	30	43	35	33	45
15	R-3390	29	62	59	45	22	55	59	28	39	46
16	P-9001	18	34	33	41	36	31	43	46	26	27
17	P-9002	30	25	29	57	50	48	36	51	11	50
18	k-9804	39	42	36	32	43	23	29	43	61	34
19	K-2098	51	33	44	35	41	50	47	29	29	55
20	L-3092	45	50	18	43	35	21	40	58	44	38
21	K-9803	21	38	59	31	40	36	14	52	36	47
22	F-9210	54	46	50	55	44	58	33	33	63	48
23	L-2801	43	51	39	20	11	24	18	24	39	31
24	F-2830	37	36	21	33	46	19	47	41	51	42
25	C-3921	28	32	33	26	55	38	18	20	26	36
26	R-3392	57	28	56	44	37	41	12	47	66	49
27	R-1292	31	39	34	27	26	39	29	18	37	27
28	P-8372	39	25	20	12	18	17	23	24	37	29
29	K-3269	43	29	28	32	23	23	37	29	22	34
30	R-3390	19	16	29	27	31	19	16	15	15	25
31	P-9001	15	25	33	18	19	25	30	33	32	16
32	P-9002	8	20	16	22	13	31	18	20	28	21

Figure 2: Product Quantities Used for Test Cases 11-20

		Quantities used in Different Cases									
Sr. No	Product ID	Case21	Case22	Case23	Case24	Case25	Case26	Case27	Case28	Case29	Case30
1	P-9001	19	22	10	5	18	21	15	9	16	31
2	P-9002	10	31	26	19	24	34	22	16	29	14
3	k-9804	23	18	31	11	31	7	18	32	21	23
4	K-2098	28	23	22	35	16	26	31	21	34	30
5	L-3092	35	28	27	28	38	36	45	39	13	18
6	K-9803	55	34	41	31	46	28	39	29	47	38
7	F-9210	33	41	26	41	34	33	44	47	31	23
8	L-2801	26	29	39	35	22	19	51	52	28	35
9	F-2830	39	37	51	38	41	32	45	29	56	24
10	C-3921	27	41	42	46	53	53	26	43	39	49
11	R-3392	31	49	26	37	31	46	41	34	26	37
12	R-1292	23	36	38	43	47	43	35	31	38	42
13	P-8372	44	28	54	34	29	23	59	46	23	28
14	K-3269	58	37	41	26	36	49	43	35	33	45
15	R-3390	29	55	31	49	22	55	63	28	45	46
16	P-9001	37	34	28	36	36	31	43	46	26	27
17	P-9002	46	49	29	44	41	48	22	57	39	43
18	k-9804	20	42	36	28	39	23	34	43	56	34
19	K-2098	41	33	44	51	41	42	40	29	29	55
20	L-3092	33	29	18	43	35	35	38	58	44	38
21	K-9803	21	22	59	31	59	36	26	52	36	47
22	F-9210	54	32	50	63	44	58	33	33	63	51
23	L-2801	43	51	39	40	32	24	18	24	39	31
24	F-2830	37	37	21	33	46	31	47	41	51	42
25	C-3921	28	34	33	26	37	38	32	32	26	36
26	R-3392	61	28	50	44	35	45	12	41	60	39
27	R-1292	38	41	34	27	29	39	29	18	37	27
28	P-8372	44	25	30	41	24	29	23	11	19	29
29	K-3269	21	37	16	30	23	23	37	29	22	34
30	R-3390	32	20	29	31	18	36	16	31	15	25
31	P-9001	28	31	38	14	22	25	30	26	25	18
32	P-9002	16	26	21	20	31	12	23	18	14	21

Figure 3: Product Quantities Used for Test Cases 21-30

Bibliography

[1] H. M. Abu Al-Rejal, J. D. Abu Doleh, L. M. Salhieh, Z. M. Udin, and S. Mohtar, "Barriers of Supply Chain Management Practices in Manufacturing Companies in Republic of Yemen: Pre-War Perspective," *International Journal of Supply Chain Management*, vol. 6, no. 3, pp. 246–251, 2017.

[2] M. K. Adeyeri, K. Mpofu, and T. A. Olukorede, "Integration of agent technology into manufacturing enterprise: A review and platform for industry 4.0," in *International Conference on Industrial Engineering and Operations Management (IEOM).* IEEE, 2015, pp. 1–10.

[3] J. Aguilar, M. Cerrada, F. Hidrobo, J. Chacal, and C. Bravo, "Specification of a multiagent system for planning and management of the production factors for automation based on the SCDIA framework and MASINA methodology," *WSEAS Transactions on Systems and Control*, vol. 3, no. 2, pp. 79–88, 2008.

[4] J. Albrecht, R. Dudek, J. Auersperg, R. Pantou, and S. Rzepka, "Thermal and Mechanical Behaviour of an RFID based Smart System Embedded in a Transmission Belt Determined by FEM Simulations for Industry 4.0 Applications," in *16th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE).* IEEE, 2015, pp. 1–5.

[5] M. Andulkar, D. T. Le, and U. Berger, "A multi-case study on Industry 4.0 for SMEs in Brandenburg Germany," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018, pp. 4544 –4553, http://hdl.handle.net/10125/50463.

[6] R. J. Arend and J. D. Wisner, "Small Business and Supply Chain Management: Is There a Fit?" *Journal of Business Venturing*, vol. 20, no. 3, pp. 403–436, 2005.

[7] B. Arinze and M. Anandarajan, "A Framework for using OO Mapping Methods to Rapidly Configure ERP Systems," *Communications of the ACM*, vol. 46, no. 2, pp. 61–65, 2003.

[8] C. Arnold, D. Kiel, and K.-I. Voigt, "How the Industrial Internet of Things changes Business Models in different Manufacturing Industries," *International Journal of Innovation Management*, vol. 20, no. 08, pp. 1–25, 2016.

[9] A. M. Atieh, H. Kaylani, Y. Al-abdallat, A. Qaderi, L. Ghoul, L. Jaradat, and I. Hdairis, "Performance Improvement of Inventory Management System Processes by an Automated Warehouse Management system," *Procedia CIRP*, vol. 41, pp. 568–572, 2016.

[10] Automation, Rockwell, "Converged Plantwide Ethernet (CPwE) Design and Implementation Guide," 2011.

[11] F. Bacchus and F. Kabanza, "Using Temporal Logics to Express Search Control Knowledge for Planning," *Artificial Intelligence*, vol. 116, no. 1-2, pp. 123–191, 2000.

[12] W. Bank, "Small to Medium Size Enterprises (SMEs) Finance," https://www.worldbank.org/en/topic/smefinance, 2020, [Online; accessed 02-July-2020].

[13] N. Bansal, N. Buchbinder, and J. Naor, "Randomized competitive algorithms for generalized caching," in *Proceedings of the fortieth annual ACM symposium on Theory of computing*, 2008, pp. 235–244.

[14] T. Becker and D. Wagner, "Identification of Key Machines in Complex Production Networks," *Procedia CIRP*, vol. 41, pp. 69–74, 2016.

[15] F. Bergenti and A. Poggi, "Leap: A fipa platform for handheld and mobile devices," in *International Workshop on Agent Theories, Architectures, and Languages*. Springer, 2001, pp. 436–446.

[16] A. Berre, B. Elvesæter, N. Figay, C. Guglielmina, S. Johnsen, D. Karlsen, T. Knothe, and S. Lippe, "The ATHENA Interoperability Framework," in *IESA*. Springer, 2007, pp. 569–580.

[17] Z. Bi and D. Cochran, "Big Data Analytics with Applications," *Journal of Management Analytics*, vol. 1, no. 4, pp. 249–265, 2014.

[18] J. Blackhurst, M. J. Rungtusanatham, K. Scheibe, and S. Ambulkar, "Supply Chain Vulnerability Assessment: A Network based Visualization and Clustering Analysis Approach," *Journal of Purchasing and Supply Management*, vol. 24, no. 1, pp. 21–30, 2018.

[19] M. Bratman, *Intention, Plans, and Practical Reason.* Harvard University Press Cambridge, MA, 1987, vol. 10.

[20] L. Braubach, W. Lamersdorf, and A. Pokahr, "Jadex: Implementing a BDI-infrastructure for JADE agents," 2003, [Online; accessed 18-August-2017]: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.5.3969.

[21] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, and J. Mylopoulos, "Tropos: An Agent-Oriented Software Development Methodology," *Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 3, pp. 203–236, 2004.

[22] M. Brettel, N. Friederichsen, M. Keller, and M. Rosenberg, "How Virtualization, Decentralization and Network Building Change the Manufacturing Landscape: An Industry 4.0 Perspective," *International Journal of Mechanical, Industrial Science and Engineering*, vol. 8, no. 1, pp. 37–44, 2014.

[23] Business2Community, "Issues in Warehouse Management Systems," https://www.business2community.com/product-management/top-5-warehouse-management-problems-solve-02027463, 2018, [Online; accessed 17-July-2018].

[24] J. Butler and H. Ohtsubo, "ADDYMS: Architecture For Distributed Dynamic Manufacturing Scheduling," *Artificial Intelligence Applications in Manufacturing*, pp. 199–214, 1992.

[25] C. Cardoso and H. Bordini, "A Multi-Agent Extension of Hierarchical Task Network," in *10th Workshop-School on Agents, Environments, and Applications (WESAAC)*, 2016.

[26] R. C. Cardoso and R. H. Bordini, "A Distributed Online Multi-Agent Planning System," *Distributed and Multi-Agent Planning (DMAP-16)*, p. 15, 2016.

[27] CArtAgO, "Common Artifact based Architecture for Agent open," http://cartago.sourceforge.net, 2017, [Online; accessed 18-August-2017].

[28] P. Centobelli, G. Converso, T. Murino, and L. Santillo, "Flow Shop Scheduling Algorithm to Optimize Warehouse Activities," *International Journal of Industrial Engineering Computations*, vol. 7, no. 1, pp. 49–66, 2016.

[29] J. C. Chen, C.-H. Cheng, P. B. Huang, K.-J. Wang, C.-J. Huang, and T.-C. Ting, "Warehouse Management with Lean and RFID Application: A Case Study," *The International Journal of Advanced Manufacturing Technology*, vol. 69, no. 1-4, pp. 531–542, 2013.

[30] Z. Chen and M. Xing, "Upgrading of Textile Manufacturing based on Industry 4.0," in *5th International Conference on Advanced Design and Manufacturing Engineering, Atlantis Press*, 2015, pp. 2143–2146, https://www.atlantis-press.com/php/download_paper.php?id=25840478.

[31] T. M. Choi, Y. Li, and L. Xu, "Channel Leadership, Performance and Coordination in Closed Loop Supply Chains," *International Journal of Production Economics*, vol. 146, no. 1, pp. 371–380, 2013.

[32] Cisco Inc, "Cisco Ehternet to the Factory (EttF) Architecture: Overview," https://www.cisco.com/c/dam/enus/solutions/industries/docs/manufacturing/c22-403859ettfoView.pdf, [Online; accessed 1-Aug-2019].

[33] College of Trocaire, "Standard Research Methodologies and Techniques," *https://library.trocaire.edu/services/studentservices/researchassistance/10steps/*, [Online; accessed on 8 April 2019].

[34] J. Daaboul, C. Da Cunha, A. Bernard, and F. Laroche, "Design for Mass Customization: Product Variety vs. Process Variety," *CIRP Annals-Manufacturing Technology*, vol. 60, no. 1, pp. 169–174, 2011.

[35] S. De Clercq, K. Bauters, S. Schockaert, M. Mihaylov, M. De Cock, and A. Nowé, "Decentralized computation of Pareto Optimal Pure Nash Equilibria of Boolean games with privacy concerns," in *Proceedings of the 6th International Conference on Agents and Artificial Intelligence-Volume 2*. SCITEPRESS-Science and Technology Publications, Lda, 2014, pp. 50–59.

[36] R. De Koster, T. Le-Duc, and K. J. Roodbergen, "Design and control of Warehouse Order Picking: A Literature Review," *European journal of operational research*, vol. 182, no. 2, pp. 481–501, 2007.

[37] P. M. Deane, *The First Industrial Revolution*. Cambridge University Press, 1979, Second Edition.

[38] Demo3D-Simulation-Tool, "Demo3D, A Product of SOLIDWORKS and Rockwell Automation Industry, USA, for Simulating Industrial Setups," https://www.demo3d.com/home/, 2019, [Online; accessed 31-May-2019].

[39] B. Denkena, M. Zwick, and P.-O. Woelk, "Multiagent-Based Process Planning and Scheduling in Context of Supply Chains," in *Holonic and Multi-Agent Systems for Manufacturing*, V. Mařík, D. McFarlane, and P. Valckenaers, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 100–109.

[40] F. U. Din and S. Anwer, "ERP Success and Logistical Performance Indicators A Critical View:," *International Journal of Computer Science Issues http://www.ijcsi.org/papers/IJCSI-10-6-1-223-229.pdf*, pp. 223–229, 2013.

[41] F. U. Din, F. Henskens, D. Paul, and M. Wallis, "Agent-Oriented Smart Factory (AOSF): An MAS based framework for SMEs under Industry 4.0," in *KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications*. Springer, 2018, pp. 44–54.

[42] F. U. Din, F. Henskens, D. Paul, and M. Wallis, "Formalisation of Problem and Domain Definition for Agent Oriented Smart Factory (AOSF)," in *2018 IEEE Region Ten Symposium (Tensymp)*. IEEE, 2019, pp. 265–270.

[43] F. U. Din, F. Henskens, D. Paul, M. Wallis, and M. A. Hashmi, "AOSR-WMS planner associated with AOSF framework for SMEs, under Industry 4.0," p. In Review, 2019.

[44] H. ElMaraghy, G. Schuh, W. ElMaraghy, F. Piller, P. Schönsleben, M. Tseng, and A. Bernard, "Product Variety Management," *CIRP Annals-Manufacturing Technology*, vol. 62, no. 2, pp. 629–652, 2013.

[45] EuroSped, "Dataset Information for Warehouse and Logistics," http://www.eurosped.bg/en/eurolog-warehouse-logistics-4pl/, 2018, [Online; available 25-Jul-2018].

[46] M. E. Fayad and D. C. Schmidt, "Lessons Learned Building Reusable OO Frameworks for Distributed Software," *Communications of the ACM*, vol. 40, no. 10, pp. 85–87, 1997.

[47] T. Finin, R. Fritzson, D. McKay, and R. McEntire, "KQML As an Agent Communication Language," in *Proceedings of the Third International Conference on Information and knowledge management.* ACM, 1994, pp. 456–463.

[48] FIPA ACL, "FIPA ACL Message Structure Specification," *Foundation for Intelligent Physical Agents, http://www. fipa. org/specs/fipa00061/SC00061G. html (30.6. 2004)*, 2002.

[49] FIPA Foundation, "The Foundation for Intelligent Physical Agents," http://www.fipa.org, 2019, [Online; accessed 10-April-2019].

[50] M. S. Fox, M. Barbuceanu, and R. Teigen, "Agent-Oriented Supply Chain Management," *International Journal of Flexible Manufacturing Systems*, vol. 12, no. 2-3, pp. 165–188, 2000.

[51] C. Freeman and F. Louçã, *As Time Goes By: From the Industrial Revolutions to the Information Revolution.* Oxford University Press, 2001, First Edition.

[52] D. Frey, P. Woelk, T. Stockheim, and R. Zimmermann, "Integrated Multi-Agent-based Supply Chain Management," in *Enabling Technologies: Infrastructure for*

Collaborative Enterprises, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on. IEEE, 2003, pp. 24–29.

[53] B. O. G. Ekren, A. Erkollar, "Research Trends in Industry 4.0: A Content Analysis," *International Symposium on Production Research*, vol. 55, no. 12, pp. 13–15, 2017.

[54] J. P. Gagliardi, J. Renaud, and A. Ruiz, "Models for Automated Storage and Retrieval Systems: A Literature Review," *International Journal of Production Research*, vol. 50, no. 24, pp. 7110–7125, 2012.

[55] German Industry 4.0, "SMEs in Genramny, German Trade and Industry (GTAI)," *https://www.gtai.de/GTAI/Navigation/EN/Invest/Business-location-germany/Economic-profile/economic-structure,t=-economic-backbone-small-and-mediumsized-enterprises,did=214024.html*, [Online; accessed on 7 Feb 2019].

[56] Glenvern-Associates, "Glenvern Associates, An Australian Industrial Consultancy Firm," http://www.glenvernasc.com.au, 2019, [Online; accessed 31-May-2019].

[57] D. G. I. Global, "Warehouse Prducts Classes," http://www.dgiglobal.com/classes, 2018, [Online; available 25-Jul-2018].

[58] A. Golovatova and Z. Jinshan, "Optimization of Goods Incoming Process," Master's thesis, University of Boras, Sweden, Sweden, 2010.

[59] E. H. Grosse, C. H. Glock, and W. P. Neumann, "Human Factors in Order Picking: A Content Analysis of the Literature," *International Journal of Production Research*, vol. 55, no. 5, pp. 1260–1276, 2017.

[60] J. Gu, M. Goetschalckx, and L. F. McGinnis, "Research on Warehouse Operation: A Comprehensive Review," *European Journal of Operational Research*, vol. 177, no. 1, pp. 1 – 21, 2007.

[61] S. L. Hakimi, "Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems," *Operations research*, vol. 13, no. 3, pp. 462–475, 1965.

[62] C. Harland, N. Caldwell, P. Powell, and J. Zheng, "Barriers to Supply Chain Information Integration: SMEs Adrift of eLands," *Journal of Operations Management*, vol. 25, no. 6, pp. 1234–1254, 2007.

[63] M. K. A. Hassan and Y.-H. Chen-Burger, "ER-Agent Communication Languages and Protocol for Large-Scale Emergency Responses," in *Agents and Multi-Agent Systems: Technologies and Applications 2018*, G. Jezic, Y.-H. J. Chen-Burger, R. J. Howlett, L. C. Jain, L. Vlacic, and R. Šperka, Eds. Cham: Springer International Publishing, 2019, pp. 134–143.

[64] W. H. Hausman, L. B. Schwarz, and S. C. Graves, "Optimal Storage Assignment in Automatic Warehousing Systems," *Management Science*, vol. 22, no. 6, pp. 629–638, 1976.

[65] W. He and L. Da Xu, "Integration of distributed enterprise applications: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 35–42, 2014.

[66] A. Hector, F. Henskens, and M. Hannaford, "A Software Engineering Process for BDI Agents," *Agent and Multi-Agent Systems: Technologies and Applications*, pp. 132–141, 2008.

[67] K. Henning, W. Wolfgang, and H. Johannes, "Recommendations for Implementing the Strategic Initiative Industrie 4.0," *AcaTech–National Academy of Science and Engineering*, vol. 1, no. 1, pp. 13–57, 2013.

[68] M. Hermann, T. Pentek, and B. Otto, "Design Principles for Industrie 4.0 Scenarios," in *49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2016, pp. 3928–3937.

[69] E. Hofmann and M. Rüsch, "Industry 4.0 and the Current Status as well as Future Prospects on Logistics," *Computers in Industry*, vol. 89, pp. 23 – 34, 2017.

[70] N. Howden, R. Rönnquist, A. Hodgson, and A. Lucas, "JACK Intelligent Agents-Summary of an Agent Infrastructure," in *5th International Conference on Autonomous Agents*, 2001.

[71] M. Huget, J. Odell, and B. Bauer, "The AUML Approach," in *Methodologies and Software Engineering for Agent Systems.* Springer, 2004, pp. 237–257.

[72] D. Hutchins, *Just In Time.* Gower Publishing, Ltd. England, 1999, Second Edition.

[73] IIC-Object-Management-Group-Inc., "US Industry 4.0 Plan - Industrial Internet Consortium," *https://www.iiconsortium.org*, [Online; accessed 11-Feb-2019].

[74] Industry in Germany, "German Trade and Industry (GTAI)," *https://www.gtai.de/GTAI/Navigation/EN/Invest/industrie-4-0.html*, [Online; accessed on 14 Nov 2017].

[75] D. Ivanov, A. Dolgui, B. Sokolov, F. Werner, and M. Ivanova, "A Dynamic Model and an Algorithm for Short-term Supply Chain Scheduling in the Smart Factory Industry 4.0," *International Journal of Production Research*, vol. 54, no. 2, pp. 386–402, 2016.

[76] D. Ivanov, B. Sokolov, and A. Dolgui, "The Ripple Effect in Supply Chains: Trade-off 'Efficiency-Flexibility-Resilience' in Disruption Management," *International Journal of Production Research*, vol. 52, no. 7, pp. 2154–2172, 2014.

[77] S. Izza, "Integration of industrial information systems: from syntactic to semantic integration approaches," *Enterprise Information Systems*, vol. 3, no. 1, pp. 1–57, 2009.

[78] JaCaMo-Framework, "Framework for Jason, CArtAgO and Moise," http://jacamo.sourceforge.net, 2017, [Online; accessed 18-August-2017].

[79] JACK, "JACK Project: Multi-Agent Development Environment Framework," http://aosgrp.com/products/jack/, 2019, [Online; accessed 10-April-2019].

[80] Jason, "Java based interpreter for Agents using AgentSpeak," http://jason.sourceforge.net/wp/, 2017, [Online; accessed 18-August-2017].

[81] Java Agent Development Framework, "JADE Open Source Project: Java Agent Development Environment Framework," http://jade.tilab.com/, 2017, [Online; accessed 18-August-2017].

[82] N. Jazdi, "Cyber Physical Systems in the Context of Industry 4.0," in *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*, May 2014, pp. 1–4.

[83] T. M. Jeng, S. C. Tzeng, C. W. Tseng, G. W. Xu, and Y. C. Liu, "The Design and Fabrication of a Temperature Diagnosis System for the Intelligent Rotating Spindle of Industry 4.0," *Smart Science*, vol. 4, no. 1, pp. 38–43, 2016.

[84] N. R. Jennings, "On Agent-based Software Engineering," *Artificial Intelligence*, vol. 117, no. 2, pp. 277–296, 2000.

[85] G. Jezic, Y.-H. J. Chen-Burger, R. J. Howlett, L. C. Jain, L. Vlacic, and R. Šperka, "Agents and multi-agent systems: Technologies and applications 2018," in *Conference proceedings KES-AMSTA*, vol. 18. Springer, 2018, p. 253.

[86] M. M. Jones, M. O. Juneja, K. Gnanamurthy, K. Kandikuppa, J. Y. W. Sheu, E. R. V. William, G. R. Hadagali, S. S. Rawat, V. Berry, D. Agrawal *et al.*, "Consigned Inventory Management System," Mar. 31 2016, US Patent App. 14/499,372.

[87] S. Joosten and S. Purao, "A Rigorous Approach for Mapping Workflows to Object-Oriented is Models," *Journal of Database Management (JDM)*, vol. 13, no. 4, pp. 1–19, 2002.

[88] N. Julka, I. Karimi, and R. Srinivasan, "Agent-based Supply Chain Management—2: A Refinery Application," *Computers & Chemical Engineering*, vol. 26, no. 12, pp. 1771–1781, 2002.

[89] N. Julka, R. Srinivasan, and I. Karimi, "Agent-based Supply Chain Management—1: Framework," *Computers & Chemical Engineering*, vol. 26, no. 12, pp. 1755–1769, 2002.

[90] I. Kacem, S. Hammadi, and P. Borne, "Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic," *Mathematics and computers in simulation*, vol. 60, no. 3-5, pp. 245–276, 2002.

[91] P. Kadera and P. Novák, "Performance Modeling Extension of Directory Facilitator for Enhancing Communication in FIPA-Compliant Multiagent Systems," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 688–695, 2017.

[92] K. B. Kahn, "Market orientation, interdepartmental integration, and product development performance," *Journal of Product Innovation Management*, vol. 18, no. 5, pp. 314 – 323, 2001.

[93] C. Kallina and J. Lynn, "Application of the Cube-Per-Order Index Rule for Stock Location in a Distribution Warehouse," *Interfaces*, vol. 7, no. 1, pp. 37–46, 1976.

[94] H. S. Kang, J. Y. Lee, S. Choi, H. Kim, J. H. Park, J. Y. Son, B. H. Kim, and S. Do Noh, "Smart Manufacturing: Past Research, Present Findings, and Future Directions," *International Journal of Precision Engineering and Manufacturing-Green Technology*, vol. 3, no. 1, pp. 111–128, 2016.

[95] S. Karnouskos and P. Leitão, "Key Contributing Factors to the Acceptance of Agents in Industrial Environments," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 696–703, 2017.

[96] B. Kim, S. S. Heragu, R. J. Graves, and A. St Onge, "A Hybrid Scheduling and Control System Architecture for Warehouse Management," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 991–1001, 2003.

[97] R. Kishore, H. Zhang, and R. Ramesh, "Enterprise integration using the agent paradigm: foundations of multi-agent-based integrative business information systems," *Decision Support Systems*, vol. 42, no. 1, pp. 48 – 78, 2006.

[98] M. Kofler, A. Beham, S. Wagner, M. Affenzeller, and W. Achleitner, "Re-warehousing vs. Healing: Strategies for Warehouse Storage Location Assignment," in *3rd IEEE International Symposium on Logistics and Industrial Informatics*. IEEE, 2011, pp. 77–82.

[99] M. Kofler, A. Beham, S. Wagner, M. Affenzeller, and C. Reitinger, "Reassigning Storage Locations in a Warehouse to Optimize the Order Picking Process," in *Proceedings of the 22nd European Modeling and Simulation Symposium EMSS*, 2010, pp. 77–82.

[100] R. D. Koster, A. L. Johnson, and D. Roy, “Warehouse Design and Management,” *International Journal of Production Research*, vol. 55, no. 21, pp. 6327–6330, 2017.

[101] J. Kvarnström and P. Doherty, “TALplanner: A Temporal Logic Based Forward Chaining Planner,” *Annals of mathematics and Artificial Intelligence*, vol. 30, no. 1, pp. 119–169, 2000.

[102] J. Lee, B. Bagheri, and H.-A. Kao, “A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems,” *Manufacturing Letters*, vol. 3, pp. 18 – 23, 2015, https://doi.org/10.1016/j.mfglet.2014.12.001.

[103] P. Leitão and S. Karnouskos, *Industrial Agents: Emerging Applications of Software Agents in Industry.* Morgan Kaufmann, 2015.

[104] S. Lemai-Chenevier, “IXTET-EXEC: planning, plan repair and execution control with time and resource management,” *PhD thesis*, 2004.

[105] L. Li, *Supply chain management: Concepts, techniques and practices: Enhancing value through collaboration.* World Scientific Publishing Co Inc, 2007.

[106] Y. Liao, F. Deschamps, E. F. R. Loures, and L. F. P. Ramos, “Past, Present and Future of Industry 4.0: A Systematic Literature Review and Research Agenda Proposal,” *International Journal of Production Research*, vol. 55, no. 12, pp. 3609–3629, 2017.

[107] K. C. Lin, J. Z. Shyu, and K. Ding, “A Cross-Strait Comparison of Innovation Policy under Industry 4.0 and Sustainability Development Transition,” *Sustainability*, vol. 9, no. 5, p. 786, 2017.

[108] C. H. Liu, “A Coordinated Scheduling System for Customer Orders Scheduling Problem in Job Shop Environments,” *Expert Systems with Applications*, vol. 37, no. 12, pp. 7831 – 7837, 2010, https://doi.org/10.1016/j.eswa.2010.04.055.

[109] M. Llonch, M. Bernardo, and P. Presas, “A Case Study of a Simultaneous Integration in an SME: Implementation Process and Cost Analysis,” *International Journal of Quality & Reliability Management*, vol. 35, no. 2, pp. 319–334, 2018.

[110] W. Lu, V. Giannikas, D. McFarlane, and J. Hyde, "The Role of Distributed Intelligence in Warehouse Management Systems," pp. 63–77, 2014.

[111] Y. Lu, "Industry 4.0: A Survey on Technologies, Applications and Open Research Issues," *Journal of Industrial Information Integration*, vol. 6, pp. 1–10, 2017.

[112] A. Luder, A. Klostermeyer, J. Peschke, A. Bratoukhine, and T. Sauter, "Distributed automation: PABADIS versus HMS," *IEEE Transactions on Industrial Informatics*, vol. 1, no. 1, pp. 31–38, 2005.

[113] M. Lützenberger, T. Küster, N. Masuch, and J. Fähndrich, "Multi-agent Systems in Practice: When Research Meets Reality," in *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, 2016, pp. 796–805.

[114] H. Ma, S. Su, D. Simon, and M. Fei, "Ensemble Multi-Objective Biogeography-Based Optimization with Application to Automated Warehouse Scheduling," *Engineering Applications of Artificial Intelligence*, vol. 44, pp. 79–90, 2015.

[115] A. A. Majeed and T. D. Rupasinghe, "Internet of things (IoT) Embedded Future Supply Chains for Industry 4.0: An Assessment from an ERP-based Fashion Apparel and Footwear Industry," *International Journal of Supply Chain Management*, vol. 6, no. 1, pp. 25–40, 2017.

[116] V. Marik, D. McFarlane, and P. Valckenaers, "Holonic and Multi-agent Systems for Manufacturing," in *First International Conference on Industrial Applications of Holonic and Multi-Agent Systems*. Springer, 2003, pp. 165–174.

[117] F. Martínez, J. Aguilar, and C. Bravo, "Multiagent Systems for Production Planning in Automation," *Holonic and Multi-Agent Systems for Manufacturing*, pp. 143–152, 2011.

[118] MAS Programming Contest, "Multi-Agent Systems Programming Contest," https://multiagentcontest.org, 2019, [Online; accessed 28-Feb-2019].

[119] R. Masdefiol, M. del Mar, and F. Stävmo, "Industry 4.0–Only Designed to Fit the German Automotive Industry: A Multiple Case Study on the Feasibility of In-

dustry 4.0 to Swedish SMEs," Master's thesis, Jönköping University, International Business School, Sweden, 2016.

[120] MATSim, "MATSim: Open Source Framework for Large Scale Agent-Based Transport Simulation," https://matsim.org, 2019, [Online; accessed 28-February-2019].

[121] D. D. McCracken and E. D. Reilly, "Backus-naur form (bnf)," in *Encyclopedia of Computer Science*. Chichester, UK: John Wiley and Sons Ltd., pp. 129–131.

[122] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, "PDDL: The Planning Domain Definition Language," [Online; accessed 11-December-2017] https://www.csee.umbc.edu/courses/671/fall12/hw/hw6/pddl1.2.pdf.

[123] B. McEleney, G. O'Hare, and J. Sampson, "An Agent Based System for Reducing Changeover Delays in a Job-Shop Factory Environment," in *Proc. of PAAM*, vol. 98, 1998, pp. 591–613.

[124] C. McKinsey, "The Impact and Future of Manufacturing," https://www.mckinsey.com/business-functions/operations/our-insights/the-future-of-manufacturing, 2019, [Online; available 3-April-2019].

[125] L. Minglei, W. Hongwei, and Q. Chao, "A novel planning approach for handling disruption during plan execution," *Applied Intelligence*, vol. 46, no. 4, pp. 800–809, 2017.

[126] K. Miyashita, "CAMPS: A Constraint-based Architecture For Multiagent Planning and Scheduling," *Journal of Intelligent Manufacturing*, vol. 9, no. 2, pp. 147–154, 1998.

[127] A. Moeuf, R. Pellerin, S. Lamouri, S. Tamayo-Giraldo, and R. Barbaray, "The industrial management of SMEs in the era of Industry 4.0," *International Journal of Production Research*, vol. 56, no. 3, pp. 1118–1136, 2018.

[128] Moise, "Moise Organization Programming for Mulit-agent Systems," http://moise.sourceforge.net, 2017, [Online; accessed 18-August-2017].

[129] J. Mokyr, "The Second Industrial Revolution, 1870-1914," *Storia dell'economia Mondiale*, pp. 219–45, 1998.

[130] L. Mönch, M. Stehli, and J. Zimmermann, "FABMAS: An Agent-Based System for Production Control of Semiconductor Manufacturing Processes," in *Holonic and Multi-Agent Systems for Manufacturing*, V. Mařík, D. McFarlane, and P. Valckenaers, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 258–267.

[131] L. Monostori, "Cyber-Physical Production Systems: Roots, Expectations and R&D Challenges," *Procedia Cirp*, vol. 17, pp. 9–13, 2014.

[132] J. Müller, L. Maier, J. Veile, and K.-I. Voigt, "Cooperation strategies among smes for implementing industry 4.0," in *Digitalization in Supply Chain Management and Logistics: Smart and Digital Solutions for an Industry 4.0 Environment. Proceedings of the Hamburg International Conference of Logistics (HICL), Vol. 23.* Berlin: epubli GmbH, 2017, pp. 301–318.

[133] J. M. Müller, O. Buliga, and K.-I. Voigt, "Fortune favors the prepared: How SMEs approach business model innovations in Industry 4.0," *Technological Forecasting and Social Change*, vol. 132, no. C, pp. 2–17, 2018.

[134] NAB Australia and CanStar Company Financial-Comparison, "SMEs contribution to Australia's GDP,," https://www.canstar.com.au/business-loans/small-medium-businesses-contributing-half-australias-gdp/, 2018, [Online; accessed 18-June-2018].

[135] D. Nau, Y. Cao, A. Lotem, and H. Munoz-Avila, "The SHOP planning system," *AI Magazine*, vol. 22, no. 3, p. 91, 2001.

[136] D. S. Nau, T. C. Au, O. Ilghami, U. Kuter, J. W. Murdock, D. Wu, and F. Yaman, "SHOP2: An HTN Planning System," *Journal of artificial intelligence research*, vol. 20, pp. 379–404, 2003.

[137] Netlogo, "A Multi-Agent Programmable Modelling Environment," https://ccl.northwestern.edu/netlogo/, 2019, [Online; accessed 28-February-2019].

[138] H. S. Nwana, D. T. Ndumu, L. C. Lee, and J. C. Collis, "ZEUS: A Toolkit and Approach for Building Distributed Multi-agent Systems," in *Proceedings of the Third Annual Conference on Autonomous Agents*, ser. AGENTS '99. New York, NY, USA: ACM, 1999, pp. 360–361, http://doi.acm.org/10.1145/301136.301234.

[139] Office of United States Trade Representative, "Small and Medium Size Enterprise (SMEs) in USA," *https://ustr.gov/trade-agreements/free-trade-agreements/transatlantic-trade-and-investment-partnership-t-tip/t-tip-12*, [Online; accessed on 7 Feb 2019].

[140] Oracle Official, "Oracle, Small to Medium Business," *https://www.oracle.com/smb/*, [Online; accessed on 18 Feb 2019].

[141] L. Padgham and M. Winikoff, *Developing Intelligent Agent Systems: A Practical Guide*, 13th ed. John Wiley & Sons, 2005.

[142] S. K. Paik, "Supply Management in Small and Medium-sized Enterprises: Role of SME's size," *Supply Chain Forum: An International Journal*, vol. 12, no. 3, pp. 10–21, 2011.

[143] K. A. Patterson, C. M. Grimm, and T. M. Corsi, "Adopting New Technologies for Supply Chain Management," *Transportation Research Part E: Logistics and Transportation Review*, vol. 39, no. 2, pp. 95–121, 2003.

[144] J. Paulo Leitão, Luis Ribeir, "Guest Editorial-Special Section on Smart Agents and Cyber Physical Systems for Future Indutrial Systems," vol. 13, pp. 659–659, 2017.

[145] P. Perez, H. Karp, R. Dieng, O. Corby, A. Giboin, F. Gandon, J. Quinqueton, A. Poggi, G. Rimassa, C. Fietta *et al.*, "Corporate memory management through agents," *E-business: Key Issues, Applications and Technologies*, pp. 360–366, 2000.

[146] C. G. Petersen, "An Evaluation of Order Picking Routing Policies," *International Journal of Operations & Production Management*, vol. 17, no. 11, pp. 1098–1111, 1997.

[147] D. Piasecki, "Warehouse management systems (wms)," *Inventory Operations Consulting LLC, http://www. inventoryops. com/warehouse_management_systems. htm*, 2005.

[148] R. Poklemba, "Mapping Requirements and Roadmap Definition for Introducing I 4.0 in SME Environment," *Advances in Manufacturing Engineering and Materials*, p. 183, 2019.

[149] T. Poon, K. Choy, H. K. Chow, H. C. Lau, F. T. Chan, and K. Ho, "A RFID Case-based Logistics Resource Management System for Managing Order-Picking Operations in Warehouses," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8277 – 8301, 2009.

[150] D. Preuveneers and Y. Berbers, "Modeling Human Actors in an Intelligent Automated Warehouse," in *International Conference on Digital Human Modeling.* Springer, 2009, pp. 285–294.

[151] A. Radziwon, A. Bilberg, M. Bogers, and E. S. Madsen, "The Smart Factory: Exploring Adaptive and Flexible Manufacturing Solutions," *Procedia Engineering*, vol. 69, pp. 1184–1190, 2014.

[152] R. Ravi and L.-C. Wu, "Demystifying industry 4.0 implications of internet of things and services for the chemical industry," Ph.D. dissertation, 2016.

[153] Repast, "Repast: Open Source Project: Agent Based Modelling and Simulation Platform," https://repast.github.io, 2019, [Online; accessed 28-February-2019].

[154] C. Rhodes, "Business Statistics: Number of SMEs in UK," *Business Statistics -Parliament UK (Briefing Paper)*, Dec 2018.

[155] A. Ricci, M. Viroli, and A. Omicini, "Give Agents Their Artifacts: The A&A Approach for Engineering Working Environments in MAS," in *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems.* ACM, 2007, p. 150.

[156] G. Richards, *Warehouse management: a complete guide to improving efficiency and minimizing costs in the modern warehouse.* Kogan Page Publishers, 2017.

[157] Rockwell Automation, "Integrated Control Systems: Overview," https://literature.rockwellautomation.com/idc/groups/literature/documents/br/ia-br005-en-p.pdf, [Online; accessed 1-Aug-2019].

[158] Rockwell-Automation, "Rockwell Automation Industry, USA," https://www.rockwellautomation.com/site-selection.html, 2019, [Online; accessed 31-May-2019].

[159] B. Rouwenhorst, B. Reuter, V. Stockrahm, G.-J. van Houtum, R. Mantel, and W. H. Zijm, "Warehouse design and control: Framework and literature review," *European journal of operational research*, vol. 122, no. 3, pp. 515–533, 2000.

[160] S. Russell, P. Norvig, and A. Intelligence, "A modern approach," *Artificial Intelligence. Prentice-Hall, Englewood Cliffs*, vol. 25, p. 27, 1995.

[161] R. G. Saade and H. Nijher, "Critical Success Factors in Enterprise Resource Planning Implementation: A Review of Case Studies," *Journal of Enterprise Information Management*, vol. 29, no. 1, pp. 72–96, 2016.

[162] N. M. Sadeh, D. W. Hildum, and D. Kjenstad, "Agent-based e-supply chain decision support," *Journal of Organizational Computing and Electronic Commerce*, vol. 13, no. 3-4, pp. 225–241, 2003.

[163] SAP Official, "SAP, Business One," *https://www.sap.com/australia/products/sme-business-software.html*, [Online; accessed on 18 Feb 2019].

[164] R. Schmidt, M. Möhring, R.-C. Härting, C. Reichstein, P. Neumaier, and P. Jozinović, "Industry 4.0 Potentials for Creating Smart Products: Empirical Research Results," in *International Conference on Business Information Systems*. Springer, 2015, pp. 16–27.

[165] T. Schöneberg, A. Koberstein, and L. Suhl, "An Optimization Model for Automated Selection of Economic and Ecologic Delivery Profiles in Area Forwarding based Inbound Logistics Networks," *Flexible Services and Manufacturing Journal*, vol. 22, no. 3-4, pp. 214–235, 2010.

[166] E. Scopus, "Elsvier Documents Search: "Industry 4.0 and SMEs from 2013-2018"," https://www.scopus.com/search/form.uri?display=basic, 2018, [Online; available 06-Nov-2018].

[167] W. Shen, "Distributed Manufacturing Scheduling using Intelligent Agents," *IEEE Intelligent Systems*, vol. 17, no. 1, pp. 88–94, 2002.

[168] W. Shen, Q. Hao, H. J. Yoon, and D. H. Norrie, "Applications of Agent-based Systems in Intelligent Manufacturing: An Updated Review," *Advanced engineering INFORMATICS*, vol. 20, no. 4, pp. 415–431, 2006.

[169] W. Shen, F. Maturana, and D. H. Norrie, "MetaMorph II: An Agent-based Architecture for Distributed Intelligent Design and Manufacturing," *Journal of Intelligent Manufacturing*, vol. 11, no. 3, pp. 237–251, 2000.

[170] D. Singh, L. Padgham, and B. Logan, "Integrating BDI agents with agent-based simulation platforms," *Autonomous Agents and Multi-Agent Systems*, vol. 30, no. 6, pp. 1050–1071, 2016.

[171] P. Singh, M. Van Sinderen, and R. Wieringa, "Smart Logistics: An Enterprise Architecture Perspective," in *CAiSE Forum. 29th CAiSE Conference*, 2017, pp. 9–16.

[172] SMEs, "Small to Medium Size Enterprises (SMEs)," https://www.thebalancesmb.com/sme-small-to-medium-enterprise-definition-2947962, 2020, [Online; accessed 02-July-2020].

[173] L. Sommer, "Industrial Revolution Industry 4.0: Are German Manufacturing SMEs the First Victims of this Revolution?" *Journal of Industrial Engineering and Management*, vol. 8, no. 5, p. 1512, 2015.

[174] V. Strobel and A. Kirsch, "Planning in the wild: modeling tools for PDDL," in *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*. Springer, 2014, pp. 273–284.

[175] J. M. Swaminathan, S. F. Smith, and N. M. Sadeh, "Modeling Supply Chain Dynamics: A Multiagent Approach," *Decision Sciences*, vol. 29, no. 3, pp. 607–632, 1998.

[176] A. Świerczek, "The Impact of Supply Chain Integration on the "Snowball Effect" in the Transmission of Disruptions: An Empirical Evaluation of the Model," *International Journal of Production Economics*, vol. 157, pp. 89–104, 2014.

[177] L. Tesfatsion, "Agent-based computational economics: modeling economies as complex adaptive systems," *Information Sciences*, vol. 149, no. 4, pp. 262–268, 2003.

[178] P. B. Val, M. Garcia-Valls, and I. Estevez-Ayres, "Simple asynchronous remote invocations for distributed real-time java," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 3, pp. 289–298, 2009.

[179] VDI, "Standard for Warehouse Management System," https://infostore.saiglobal.com/en-au/Standards/VDI-4490-2007-743848/, 2017, [Online; accessed 11-December-2017].

[180] G. G. Villa, "The Influence of Different Innovation Enablers on the Adoption of Industry 4.0 by SMEs and Large Companies.: A Comparative Case Study," Master's thesis, Uppsala University, Uppsala, 2018.

[181] S. Voss, H.-J. Sebastian, and J. Pahl, "Intelligent Decision Support and Big Data for Logistics and Supply Chain Management: A Biased View," in *50th Hawaii International Conference on System Sciences*, 2017, pp. 1338–1340.

[182] B. Wagner, I. Fillis, and U. Johansson, "E-business and e-Supply Strategy in Small and Medium sized Businesses (SMEs)," *Supply Chain Management: An International Journal*, vol. 8, no. 4, pp. 343–354, 2003.

[183] S. Wang, J. Wan, D. Li, and C. Zhang, "Implementing smart factory of Industrie 4.0: an outlook," *International Journal of Distributed Sensor Networks*, vol. 12, no. 1, 2016, https://doi.org/10.1155/2016/3159805.

[184] S. Wang, J. Wan, D. Zhang, D. Li, and C. Zhang, "Towards Smart Factory for Industry 4.0: A Self-Organized Multi-Agent System with Big Data based Feedback and Coordination," *Computer Networks*, vol. 101, pp. 158 – 168, 2016.

[185] X. Wang, X. Sun, and Y. Fang, "A Two-Phase Hybrid Heuristic Search Approach to the Location-Routing Problem," in *International Conference on Systems, Man and Cybernetics*, vol. 4. IEEE, 2005, pp. 3338–3343.

[186] G. O. Warehousing, "GrayOrange Autonomous Warehousing," https://www.greyorange.com/butler-goods-to-person-system, 2018, [Online; available 25-Jul-2018].

[187] U. Warehousing, "Information of Warehouse Product Management," https://www.unleashedsoftware.com/product/inventory/product-management, 2018, [Online; available 25-Jul-2018].

[188] M. M. Waris, C. Sanin, and E. Szczerbicki, "Smart innovation engineering: Toward intelligent industries of the future," *Cybernetics and Systems*, vol. 49, no. 5-6, pp. 339–354, 2018, https://doi.org/10.1080/01969722.2017.1418708.

[189] S. Weiming, "Genetic algorithms in agent-based manufacturing scheduling systems," *Integrated Computer-Aided Engineering*, vol. 9, no. 3, pp. 207–217, 2002.

[190] O. Wendt, T. Stockheim, S. Grolik, and M. Schwind, "M.: Distributed Ontology Management Prospects and Pitfalls on Our Way Towards a Web of Ontologies," 10 2011.

[191] D. Weyns, A. Omicini, and J. Odell, "Environment as a First Class Abstraction in Multiagent Systems," *Autonomous agents and multi-agent systems*, vol. 14, no. 1, pp. 5–30, 2007.

[192] WhiteHouse-National-Science-And-Technology-Council and Subcommittee on Advance-Manufacturing, "US Industry 4.0 Strategic Manufacturing Plan, 2018," *https://www.whitehouse.gov/wp-content/uploads/2018/10/Advanced-Manufacturing-Strategic-Plan-2018.pdf*, [Online; accessed 11-Feb-2019].

[193] M. Winikoff, "BDI Agent Testability Revisited," *Autonomous Agents and Multi-Agent Systems*, pp. 1–39, 2017.

[194] M. Wooldridge, N. R. Jennings, and D. Kinny, "The Gaia Methodology for Agent-Oriented Analysis and Design," *Autonomous Agents and multi-agent systems*, vol. 3, no. 3, pp. 285–312, 2000.

[195] T. H. Wu, C. Low, and J. W. Bai, "Heuristic Solutions to Multi-Depot Location-Routing Problems," *Computers & Operations Research*, vol. 29, no. 10, pp. 1393–1415, 2002.

[196] P. Wutthisirisart, "Relation-Based Item Slotting," Ph.D. dissertation, University of Missouri–Columbia, 2010.

[197] G. Xu, B. Dan, X. Zhang, and C. Liu, "Coordinating a dual-channel supply chain with risk-averse under a two-way revenue sharing contract," *International Journal of Production Economics*, vol. 147, pp. 171–179, 2014.

[198] L. Xu and H. Weigand, "The Evolution of the Contract Net Protocol," in *International Conference on Web-Age Information Management*. Springer, 2001, pp. 257–264.

[199] L. D. Xu, E. L. Xu, and L. Li, "Industry 4.0: State of the Art and Future Trends," *International Journal of Production Research*, vol. 56, no. 8, pp. 2941–2962, 2018, https://doi.org/10.1080/00207543.2018.1444806.

[200] R. Zimmermann, O. Hofmann, F. Bodendorf, A. Huber, and G. Görz, "Agent based Tracking and Tracing for Controlling Fulfillment Processes in Supply Chains," in *Proceedings of the Second Workshop on Agent-Based Simulation*, 2001.