Contribution to a Methodology and a Co-Simulation Framework assessing the impact of Lean on Manufacturing Performance

Jalal Possik

THÈSE PRÉSENTÉE
POUR OBTENIR LE GRADE DE

DOCTEUR DE L'UNIVERSITÉ DE BORDEAUX

ECOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE L'INGÉNIEUR
SPECIALITÉ: PRODUCTIQUE

Par Jalal Joseph POSSIK

CONTRIBUTION TO A METHODOLOGY AND A CO-SIMULATION FRAMEWORK ASSESSING THE IMPACT OF LEAN ON MANUFACTURING PERFORMANCE

Sous la direction de Bruno VALLESPIR
Co-directeur: Gregory ZACHAREWICZ
Co-encadrante: Aicha AMRANI

Soutenue le 16 Décembre 2019 à 15:30

Membres du jury:

Mme. FRYDMAN, Claudia Professeur des Universités	Université d'Aix-Marseille	Rapporteur
M. GRABOT, Bernard Professeur des Universités	ENIT - Tarbes	Rapporteur
M. VALLESPIR, Bruno Professeur des Universités	Université de Bordeaux	Directeur
M. ZACHAREWICZ, Gregory Professeur de l'Institut Mines Telecom	IMT - Mines d'Alès	Co-directeur
Mme. AMRANI, Aicha Maître de Conférences	Université de Bordeaux	Co-encadrante
M. DUCQ, Yves Professeur des Universités	Université de Bordeaux	Examinateur

Titre: Contribution à une Méthodologie et un Environnement de Co-Simulation pour évaluer l'impact du Lean sur la Performance de l'Entreprise

Résumé: Au-delà des compétences humaines et managériales nécessaires pour développer une entreprise, le bon déploiement du Lean peut jouer un rôle important dans la réduction des gaspillages et la maximisation de l'efficacité. Ces avantages dépendent fortement de l'intégration adéquate des techniques Lean. L'un des principaux obstacles auxquels font face les entreprises est la difficulté de choisir les outils Lean qui correspondent le mieux à leurs contextes et qui sont les mieux adaptés à l'atteinte de leurs objectifs.
Dans cette étude, nous avons proposé un environnement de co-simulation basé sur HLA avec une plateforme digitale basée sur Java pour permettre à différents fédérés (simulations à évènements discrets) qui représentent les outils opérationnels Lean de fonctionner simultanément en parallèle. Les mécanismes de gestion du temps de HLA sont nécessaires pour réguler l'avancement des fédérés pendant le cycle de simulation. Un exemple d'entreprise aéronautique est utilisé pour démontrer l'utilité de cet environnement de co-simulation. Six modèles de configuration Lean sont étudiés par rapport au modèle actuel de l'entreprise simulé sans l'application du Lean, et ce sous l'influence de la fluctuation du marché, de la diversification de la demande et de l'incertitude des ressources.

MOTS CLES: Co-Simulation, HLA, Simulations à Évènements Discrets, Lean Manufacturing, KPI

Title: Contribution to a Methodology and a Co-Simulation Framework assessing the impact of Lean on Manufacturing Performance

Abstract: Aside from the human and managerial skills necessary to propel any business, the right Lean deployment can play a big role in reducing waste and maximizing efficiency. Capturing these benefits is highly dependent on adequate Lean techniques integration. One of the major hurdles companies face is the difficulty to choose the Lean tools that best fit their contexts and that are best tailored towards reaching their objectives. In this study, we proposed an HLA based Co-Simulation framework with a Java-based digital platform to allow different federates (discrete event simulations), representing the operational Lean tools, running simultaneously in parallel. Time management mechanisms of HLA are required for regulating the advancement of the federates during the simulation run. An example of an Aeronautic company is used to demonstrate the usefulness of this co-simulation framework. Six Lean configuration models are investigated under market fluctuation, demand diversification, and uncertainty of resources contexts compared with an actual model simulated as a Lean free scenario.

KEYWORDS: Co-Simulation, HLA, Discrete Event Simulation, Lean Manufacturing, KPI

Laboratoire de l'Intégration du Matériau au Système
[IMS, CNRS UMR 5218, Bât. A31 - 351 Cours de la Libération - 33400 Talence]

Acknowledgment

Undertaking this Ph.D. has been a life-changing experience for me, and I am really thankful for the support and guidance that I received from many people. This thesis took place within the IMS (Integration: from Material to Systems) laboratory of the University of Bordeaux.

First and foremost, my sincere thanks go to my supervisors Dr. Aicha AMRANI, Prof. Gregory ZACHAREWICZ, and Prof. Bruno VALLESPIR for having trusted me and accepted to supervise my thesis. Their corrections, advices, and professionalism were essential to the realization of this work and made my Ph.D. experience more motivating and productive.

I would like to express my deepest gratitude for the members of the Jury Prof. Claudia FRYDMAN, Prof. Bernard GRABOT, and Prof. Yves DUCQ for accepting to examine and review my thesis.

I would also like to thank all the doctoral students, researchers, and staff at the IMS laboratory for their kindness. I would particularly like to thank Simon and Yousra for the good times we had together in Bordeaux. I wish to thank Ralph, Imad, Christelle, Joseph, Fadi, Edwin, Charbel, and Elie, which were always ready to support me during difficult times.

My acknowledgement would be incomplete without thanking the biggest source of my strength, my family. I would like to thank my parents, Joseph and Odile, for their confidence, love, and guidance during all these years, my sisters, Christelle and Elite, for their love and unending support, and most importantly, my loving and supportive wife, Soha, and my two wonderful children, Céline and Joe, who stood by me in good and difficult times and provided endless inspiration.

Jalal Joseph Possik
University Of Bordeaux
November 2019

Table of Contents

List of Figures

List of Tables

Acronyms and Abbreviations

ATO	Assemble to Order
BBN	Bolt, Beranek, and Newman
BPMI	Business Process Management Initiative
BPMN	Business Process Model and Notation
BPR	Business Process Reengineering
CODP	Customer Order Decoupling Point
DARPA	Defense Advanced Research Projects Agency
DES	Discrete Event Simulation
DIS	Distributed Interactive Simulation
DoD	Department of Defense
DS	Distributed Simulation
ETO	Engineer to Order
FMI	Functional Mock-up Interface
FOM	Federation Object Model
GUI	Graphical User Interface
HLA	High Level Architecture
HRM	Human Resource Management
JIT	Just In Time
LM	Lean Manufacturing
LP	Lean Production
MIT	Massachusetts Institute of Technology
MTO	Make to Order
MTS	Make to Stok
OEM	Original Equipment Manufacturer
OM	Operations Management
OMG	Object Management Group
OTD	On-time delivery
P2P	Peer to Peer
PC	Personal Computer
PME	Process and Manufacturing Engineering
QMS	Quality Management System
RTI	Run-Time Infrastructure
SCOR	Supply Chain Operation Reference
SME	Small and Medium Enterprises
SMED	Single-Minute Exchange of Die
SPC	Statistical process control
TOC	Theory Of Constraints

TPM	Total Productive Maintenance
TPS	Toyota Production System
TQM	Total Quality Management
TSO	Time Stamp Order
TTM	Time To Market
UML	Unified Modelling Language
VM	Visual Management
VSM	Value Stream Mapping
WIP	Work In Progress
WS	Workstation

INTRODUCTION

The rise of Lean thinking was first seen in Japan with Toyota Production System (TPS), right after the Second World War, when Japan was left defeated. The term “Lean” was conceived by John Krafcik from Massachusetts Institute of Technology (MIT) and James Womack in the United States. The Lean thinking in TPS has helped rebuild the existing economy and advance the goods industry, especially, the automobile industry. The techniques that emerged from Lean were developed over the years and currently constitute a wide panel.

Lean techniques have improved the performance of companies in the manufacturing sector (goods production) as well as the services sector (banks, hospitals, etc.). In fact, many studies have yielded satisfying results, but in contrast, many other researches and publications of case studies have showed that Lean techniques have failed, and in some cases have carried many difficulties in its implementation process. More details and references are provided in the literature review of Chapter I. Companies are tricked by the results and are doubtful about Lean’s implementation. The examination of Lean application raises an important common point: no instructions are provided, and the effectiveness of Lean depends mainly on the knowledge of the manager. Lean principles are well established and Lean techniques widespread but no method of Lean implementation yet exists. Many questions arise as well: Which Lean technique should be implemented first? Which technique suits each context best? There exists no definitive answer since the economic context, the production constraints, and the evolution of demand are disregarded when it comes to decision making. The complexity and variability of the actual context regarding industrial systems impose an accurate and dynamic vision in order to meet the changing needs of the clients, the limitations of the production constraints, and the economic context concerning the evolution of demand. Meanwhile, existing production systems find themselves in charge of different situations where they often find it difficult to adapt to circumstances. Efficient production systems become unreachable if Lean tools are implemented in a hazardous, opportunistic and intuitive manner.

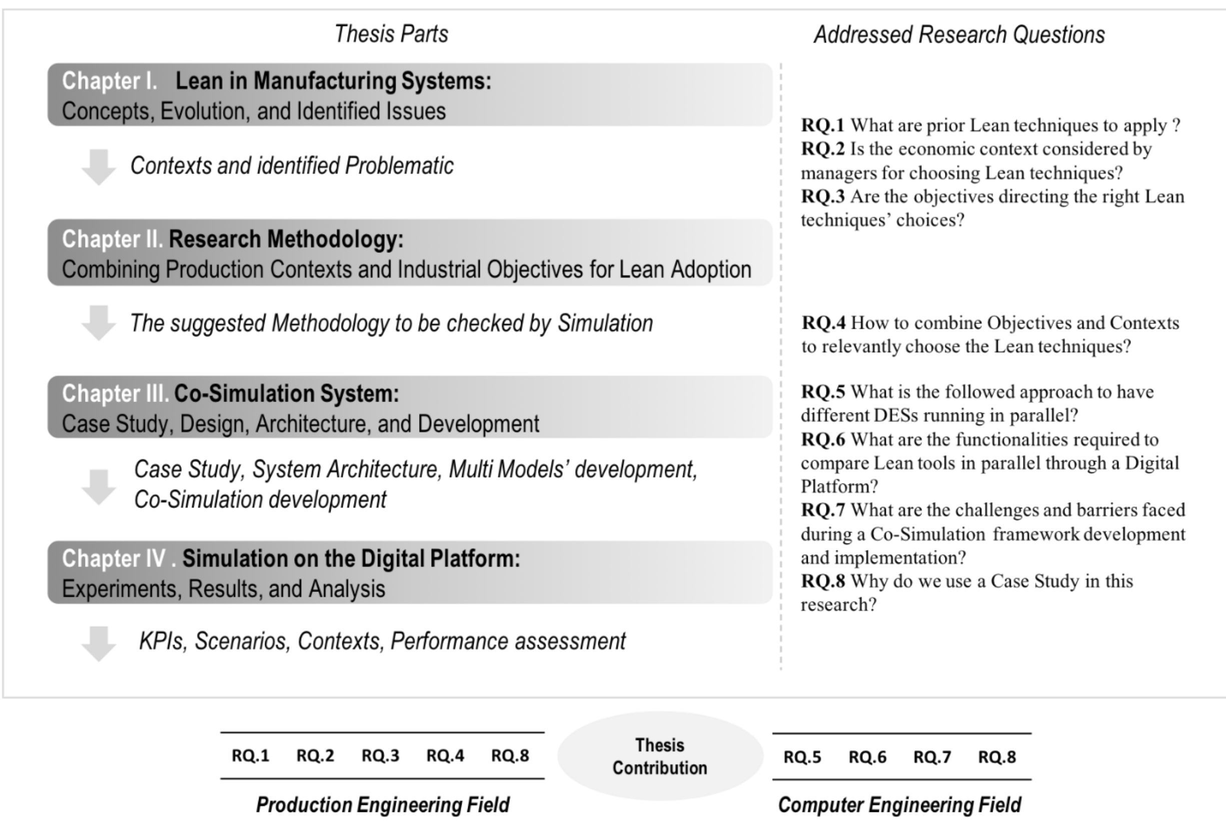
The choice of Lean techniques lacks genericity, which makes Lean approaches less advantageous if the wrong tools were implemented. The perception of the economic context and the constraints of production, if properly grasped, become important assets that aid managers in deciding which production techniques to adopt.

In a highly competitive environment, an accurate application of Lean, with the correct implementation of its techniques, becomes a crucial ingredient to the success of manufacturing firms. A clear methodology provided to manufacturers becomes an essential enabler as well, to make the appropriate decisions based on their situations.

In recent years, many studies have helped in the development of Lean and Six Sigma approaches in companies. On the other hand, there are studies that measured the contribution of Lean techniques to companies in terms of cost, quality, delay, agility, and flexibility (Antony et al. 2012; Bhamu et al. 2014; Fadly Habidin et al. 2013; Karlsson et al. 1996). In addition, some works showed the key success factors that improve the deployment of Lean projects in companies. Differently, very few studies have built models or methodologies of decision support based on simulation findings (Villarreal et al. 2016; Božičković et al. 2012; Gurumurthy et al. 2011; Detty et al. 2000; Greinacher et al. 2016). Through analyzing the fact that very few tools have been developed to reach an adequate contextual matching, we have sought in our research the possibility to define a methodological solution that will improve decision support for choosing Lean technique, based on the combination of the economic context and industrial objectives.

The proposed contribution must be applicable to all production systems in complex environments (i.e. number of products, number of disruptions in production, and other parameters that will be identified later). This interconnection between variable contexts and objectives is mainly what can make the choice of Lean techniques easier. As a result, the simulation of the production flow becomes an interesting tool to verify and examine the behavior of the production system. Modeling and simulation consist of establishing a conceptual representation of reality. Production flow simulations are extremely powerful and have been used for decades by manufacturing systems (Rymaszewska 2016).

The following synoptic represents the structure of the manuscript based on numerous scientific questions:



Thesis Synoptic (Possik 2019)

The first chapter is devoted to displaying the context of Lean deployment in industries and analyzing the research problem. We will present different key concepts regarding Lean principles, tools, and key success factors, as well as barriers that might cause implementation failure. In particular, we will show the impact of Lean approaches on the performance of firms according to the existing literature. The suitability of Lean techniques in a given industrial context is not an easy and obvious combination. A literature review will focus on showing the contribution of different approaches that are globally or partially involved in the adequacy of Lean tools implementation. Through different articles and literature reviews dealing with high number of companies' samples, we extract different research interests that Lean research is concerned about.

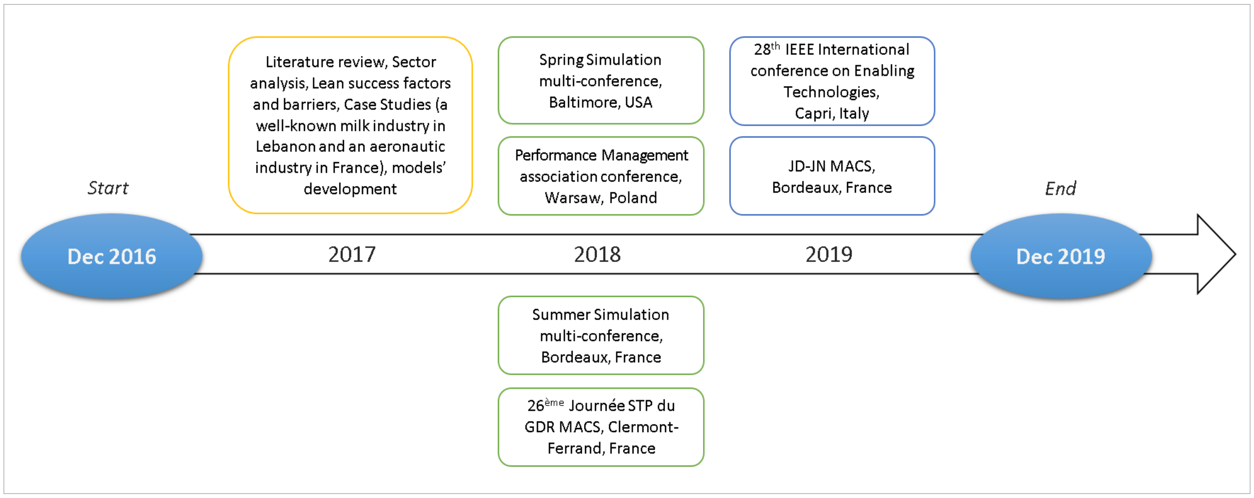
After analyzing our findings from the first chapter, we will define in Chapter II the industrial objectives on which we will base our methodology. We will go through existing references written by different authors and we will select the objectives adequate to our study. In fact, the industrial objective, regardless of its nature (economic, human, or ecological), is the path to follow in order to reach the target. Furthermore, the economic context has to be identified in a more thorough way to guide the analysis of adequacy that will be later studied. In this chapter, we will establish a link between different combinations and analyze different possibilities to map the overall context. Positioning a company could then facilitate the prioritization of Lean tools according to the typologies of the context identified.

In Chapter III, we will discuss several approaches that are applicable following the development of the case study. The use of the High-Level Architecture (HLA), an IEEE standard for distributed simulations, broadens our horizons and opens the door to multiple models' development, data exchange, interoperability, reusability, and communication to external systems. The system's architecture, the development, and the simulation synchronization process will also be presented in this chapter. In addition, we will study the impact of the combination between context, objective, and the choice of Lean tools, to verify the resulting performance on the case study.

This study, via the simulation platform, has two objectives:

- First, to evaluate the reaction of different Lean tools applied to the same actual model based on different contexts.
- Second, to co-simulate different hypotheses, in parallel, in order to achieve real time visualization of the hypotheses' impact on the same case study.

In the last chapter, we will evaluate the performances based on a case study of an aeronautical equipment manufacturer. Results of the co-simulation will be observed and interpreted to avoid revisable decisions that are time and money consuming.



Discussions on the research work

The progress of this research has been the subject of publications in different conferences. Bibliographic research was performed during the year 2017 while communications and discussions began in early 2018 and persisted even during the thesis writing process. Detailed references of the papers are provided in the bibliography. The target will be to publish the results of the thesis in a scientific journal in early 2020, soon after the defense.

CHAPTER I. Lean in Manufacturing Systems: Concepts, Evolution, and Identified Issues

Manufacturing companies in the current economic world face challenges of two different natures. On one hand, the management of customers became more and more demanding; customers nowadays are requesting a variety of products and an uncompromising quality. On the other hand, the internal budget management of the companies imposes a tight budget to carry out its production and marketing. Elements related to uncertainty and variety of products became an essential part of the industry's realities, which requires a production with shorter lead times, smaller lot sizes, and an agile adaptation to the changing environment (Tersine et al. 2000; Ho et al. 2005). Moreover, the market competition is increasing which forces the manufacturers to act quickly to survive (Bhasin et al. 2006; Mishra et al. 2006). This creates a significant evolution and expansion of Lean adoption even outside its automotive origins (Womack et al. 1990). Thus, in the last two decades, companies from different industries, sectors, and services have adopted Lean management, which allowed them to improve, in many cases, their performance and competitiveness (Behrouzi et al. 2011; Radnor et al. 2006; Bhasin 2012a; Alaskari et al.; Lande et al. 2016). However, the implementation of Lean in the industry, whether at the level of the supply chain, production workshops, or engineering departments, is a complex task that requires a good understanding of the fundamentals and principles of the TPS. We will now discuss these foundations and the confronted complications since it is becoming quite common that Lean is facing many obstacles (Scherrer-Rathje et al. 2009). In addition, we will discuss the research questions and formulate the thesis' approach relatively to the existing literature approaches.

I.1 Origins, Principles and Philosophy

The concept of « Lean » first appeared officially in the article entitled "Triumph of the Lean Production System", an article from MIT by John Krafcik (Krafcik 1988) before James Womack's book "The Machine that changed the world" (Womack et al. 1990) that people most often refer to. However, it is true that a book has a larger vision and aims at a wider audience, which explains why Womack's book has become very popular.

In Table I.1 of (Bozdogan 2010), we can see different enterprise management systems applicable universally to improve enterprise performances. Agile manufacturing and Business Process Reengineering (BPR) approaches are developed management systems introduced since the 1990s. The Lean enterprise system is the oldest of the listed management systems and in spite of that, researchers are always in need of new researches and developments related to Lean manufacturing domain; extending research about Lean sustainability (Marshall 2015), developing new pilots and conceptual models for Lean implementation (Jasti et al. 2015), extending research about factors affecting a successful Lean transformation (Marodin et al. 2013), etc.

Table I.1 Overview of major approaches in production history (Bozdogan 2010)

Approach / Key Dimensions	Lean Enterprise System	Total Quality Management (TQM)	Six Sigma	Theory of Constraints (TOC)	Agile Manufacturing	Business Process Reengineering (BPR)
History	Since late 1940s (emphasis on developments since mid-1990s)	Since early 1980s	Since mid-1980s	Since mid-1980s	Since early 1990s	Since early 1990s
Goal	• Deliver value to multiple stakeholders • Build long-term dynamic network-wide capability for sustained competitive advantage	• Meet customer expectations • Improve profitability and shareholder value	• Increase customer satisfaction • Create economic wealth (higher profitability and shareholder value)	• Maximize throughput • Improve net profits	• Enhance enterprise flexibility and responsiveness • Thrive in a fast-paced, uncertain, environment	• Improve customer satisfaction • Enhance enterprise performance
Defining Feature	• Mutually supportive and reinforcing set of principles, practices and methods for evolving efficient and flexible enterprises as networked systems creating value for multiple stakeholders	• Evolving system of precepts, practices, tools and techniques for improving quality to satisfy customer needs & expectations	• Structured methods, practices and tools for reducing all sources of variation in order to improve quality, satisfy customer needs, and improve the bottom-line	• Set of ordered practices, methods and tools for improving throughput in production systems in order to maximize financial performance, by viewing the production system as "chains of interdependencies"	• Future-looking, aspirational, set of concepts and practices aimed at defining the next industrial paradigm beyond lean enterprise ideas and flexible production systems	• Manifesto for turning the prevailing industrial system on its head; a manifesto for fundamental rethinking and radical redesign of core enterprise processes
Core Concepts	• Adopt a holistic view of the networked enterprise • Stress long-term thinking • Deliver customer-pulled best lifecycle value • Eliminate waste towards the goal of creating value • Ensure stability and synchronized flow • Develop collaborative relationships and mutually-beneficial network-wide governance mechanisms • Foster a culture of continuous learning • Evolve an efficient, flexible & adaptive enterprise	• Understand and fulfill customer expectations • Concentrate on process management to reduce sources of variation • Focus on continuous quality improvement • Ensure heavy leadership involvement • Establish close links to customers & suppliers • Develop an "open" organization • Foster worker training, empowerment and fulfillment	• Adopt customer-focused culture • Reduce all sources of variation • Pursue disciplined, structured, approach to process improvement • Practice proactive, data-driven, management • Emphasize teamwork	• Improve workflow (throughput) in the production system • Concentrate on key leverage points (constraints) offering greatest performance improvements • Protect production line against interruptions • Ensure people learn better and faster	• Anticipate and meet customer needs • Deliver tailored solutions to customers • Evolve adaptive, flexible & efficient enterprise • Establish virtual organizations • Enhance ability to thrive in a fast-paced & uncertain environment	• Reinvent enterprise through fundamental rethinking of enterprise processes • Pursue radical ("clean sheet") redesign of existing business processes • Seek breakthrough process solutions

Despite of the existing differences between the approaches of Table I.1, they almost converge to the same main goal, improving operational performances in order to satisfy customers. Each of these approaches has its unique technique and process oriented tools (Cua et al. 2001; Bozdogan 2010).

The Lean management system represents a more holistic and complete approach (Bozdogan 2010), as it takes into consideration the lifecycle view of the entire production system. It helps organizations in removing wastes and reducing non value-added activities in order to improve the overall productivity and customer experience.

Total Quality Management (TQM) mainly focuses on quality management, cross-functional product design, and customer involvement (Cua et al. 2001). Six Sigma, introduced in mid-1980s, is a strategy that has been developed and implemented by managers and executives in order to eliminate the sources of variation, eliminate product defects, reduce cycle time, and increase the customer satisfaction (Pande et al. 2001). Six sigma primarily focuses on the bottom line performance, it doesn't have a product lifecycle perspective (Bozdogan 2010). The Theory Of

Constraints (TOC) approach provides a clear management paradigm to run an organization (Rahman 1998). This management system aims at maximizing the throughput and improving the organizations' profit, however, it adopts the system view without focusing on the supplier networks (Bozdogan 2010).

Agile manufacturing is a new management system that is gaining popularity. It has been promoted as the 21st century manufacturing system paradigm. It represents an interesting approach that helps in developing a flexible, adaptive, and efficient production in the actual fast-moving and changing market (Yusuf et al. 1999). Agile manufacturing systems use the Lean manufacturing ideas deeply but lack a proper structure and roadmap for adoption and implementation (Hasan et al. 2007; Bozdogan 2010). The BPR is a newly developed approach that aims to rethink, redesign, and restructure the existing company's business processes and make them more efficient in order to enhance the company's performance and improve customer satisfaction (O'Neill et al. 1999). BPR focuses on a complete replacement of existing processes by pursuing fundamental changes in order to have coherent business processes without taking any small or careful steps. BPR has theoretical limitations and lacks conceptual means to manage complexity. In addition, BPR does not focus on the cultural and human factor issues (Bozdogan 2010).

Several approaches exist for performance improvement, focusing on process management to have significant impact on operational flexibility, efficiency, and responsiveness. Companies are facing an increasingly market competition and managers are always searching for new methods, approaches, and strategies to compete. For this study, the focus will be on "Lean Management" approach, the others: BPR, TOC, TQM, Six Sigma are beyond the scope of this manuscript.

I.1.1 Lean origin: From Japan to USA

Japan has established itself as an emergent hub in Lean production. This is due to the fact that some problems and limitations were caused by mass production at that time (in the 1950s). Japan first tried to adopt mass production that was inefficient regarding the needs and specificities of the country. It was characterized by its smaller domestic market with a strong demand for a variety of cars; small and large, simple and luxurious, etc. In addition, the Japanese labor force had a strong bargaining power, there was not enough financial resources to purchase the latest production technologies, and foreign car manufacturers were reluctant to establish their operations in Japan (Sugimori et al. 1977). Toyota, a car manufacturer, proved to be a pioneer in launching a production model that would fill the gaps of the mass production and meet the needs of Japanese customers (Rymaszewska 2016). It has been found that producing small rather than huge batches would cost less (Womack et al. 1990). This discovery has been a significant step towards cost reduction and quality improvement through the elimination of unnecessary inventories and early detection of errors.

Toyota has progressively built the famous TPS to instill the basics of this new organizational approach in the production workshops. In addition, after becoming deeply rooted in the production process, Toyota has disseminated its successful initiatives in the process of developing new products based on subsystems integration (Liker 2016) and based on an iterative and collaborative concept rather than following inflexible steps (Blank 2013).

I.1.2 Lean definition, pillars and principles

The Lean approach has been defined in different ways (Dahlgaard et al. 2006; Stentoft-Arlbjørn et al. 2013). Cherrafi et al. (2016) recall that there is a difficulty to get a consensual definition of "Lean" since the concept is still evolving (Hines et al. 2004; Shah et al. 2007). Stentoft-Arlbjørn et al. (2008) and Ghosh (2013) identify three levels of Lean thinking: philosophy, principles, and tools and techniques. We will start by defining the Lean Manufacturing (LM), its system's pillars, and its five principles.

Understanding the foundation of Lean is closely linked to the customer satisfaction. The system must be built in such a way to respond and stick to the value from the end-customer's point of view (Paez et al. 2004). In Lean thinking, any business or activity unable to create value from customer's perspective is a waste to be removed or minimized (Womack et al. 1996; Myers et al. 2002). The core concern of Lean is to improve the customer value (Radnor 2000; Hines et al. 2004; Shah et al. 2007) and to eliminate wastes.

Radziwill (2013) suggests that "Lean is principally and notably a system, in essence an assimilated sequence of portions with a noticeably defined objective". Lean manufacturing can be defined as a systematic approach to recognize and eradicate wastes in order to fulfill customer demand (Shahidul et al. 2011).

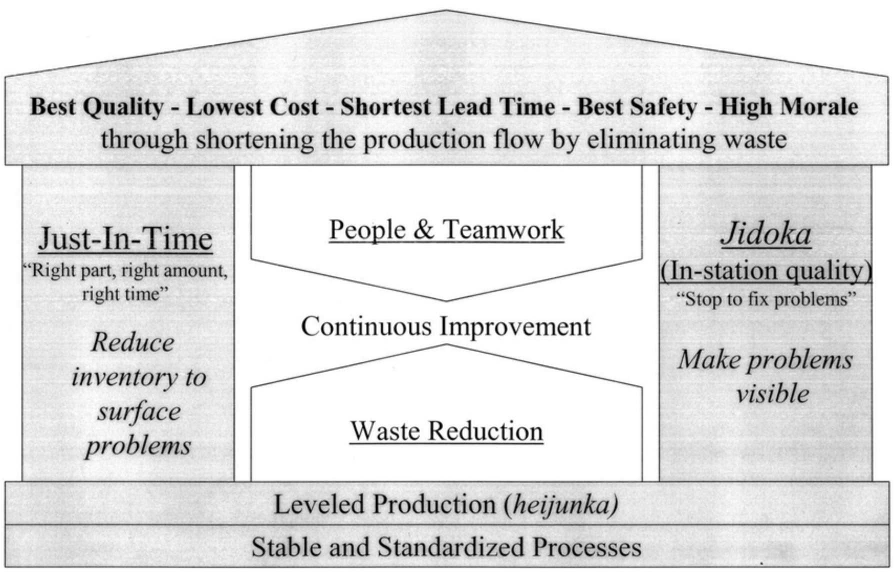


Figure I.1 TPS House

The elimination of all kinds of waste in all phases, from the order till the delivery process, should be done (Seppälä et al. 2004). Lean approach focuses on the reduction of the following eight types of wastes, called "Mudas" in Japanese TPS. These types of wastes are discussed by Dimitrov et al. (2012) and listed as per the following: Transportation, over production, waiting, over processing, motion, defects, inventory, and unused talent.

The goal of Lean is to produce better products or services, at the lowest cost and in the least time, by eliminating waste (Liker 1997; Dennis 2002). When considering the TPS and its features, it is simpler to think about it as a "house". The house of TPS is built per Figure I.1, two main pillars of TPS are very well known in the literature. The "Just In Time" (JIT) and "Jidoka" (quality at the source). Pull the flow using dynamic JIT and prevent defects at a very early stage are the main ideas of TPS in order to efficiently and quickly produce end-products that satisfy customer requirements. The Continuous Improvement (i.e. Kaizen) and the respect for people are also enablers that make a Lean system run smoothly.

Lean Production (LP) requires active, innovative, multi-skilled, and continuously motivated employees to suggest improvements in the process and in the production methods (Seppälä et al. 2004). The Lean tools and techniques are designed to make it simple to *see problems, resolve them easily, and learn from mistakes* (Mor et al. 2016).

The five key principles of Lean defined by Womack et al. (1996) became widespread in the literature and are listed as follows: value, value stream, flow, pull, and continuous improvements. These Lean principles range from identifying non-added value activities to continuously pursuing for improvements with constructive involvement of workers (Lyons et al. 2013).

I.1.3 Lean techniques supporting Lean implementation

In general, understanding the basics, foundations, pillars and principles of LM is a powerful driver of successful Lean deployment in the field. The Lean vision guides decision-makers and helps in improving operational and organizational practices in order to optimize the product value.

The success of any Lean philosophy is not limited to Lean tools, it is also important to optimize product flows in industrial systems. Improvements in the industrial processes' performances must be based on techniques developed and tested in various scientific contributions. The different practices and/or tools of Lean have been discussed and studied by various authors. We present an adapted synthesis of the work of Cherrafi et al. (2016). We can clearly notice the diverse nature of the Lean tools called also Lean practices in the literature (see Table I.2).

The literature reveals different Lean techniques, understanding these practices in our work strongly determines their use in the different production systems. These elements are also being employed in the various procurement, production, and delivery processes. It is obvious that authors use different terminologies for these tools and techniques depending on their study needs and the way they extract from the literature.

Table I.2 Adapted from Cherrafi et al. (2016) – Scientific publications dealing with Lean Techniques

Lean Tools	References
5S	(Fliedner 2008; Vais et al. 2006; Langenwalter 2006; Wilson 2010; Torielli et al. 2011a; Ecology 2007; Vinodh et al. 2011a; Pojasek 1999; Chiarini 2014b; Bae et al. 2007)
Kaizen	(Fliedner 2008; Pampanelli et al. 2014; Pampanelli et al. 2011; Vais et al. 2006; Miller et al. 2010; Rothenberg et al. 2001; Soltero et al. 2002; Nahmens 2009; Ecology 2007; Zhang et al. 2014; Wilson 2010; Vinodh et al. 2011b)
VSM	(Sobral et al. 2013; Langenwalter 2006; Torielli et al. 2011b; Park et al. 2008; Maskell et al. 2008; Ecology 2007; Aguado et al. 2013; Vinodh et al. 2011b; Ng et al. 2015; Chiarini 2014b; Bae et al. 2007; Marudhamuthu et al. 2011)
Kanban/Pull	(Fliedner 2008; Herrmann et al. 2008; King et al. 2001; Kováčová 2013; Longoni et al. 2011; Ng et al. 2015; Rothenberg et al. 2001; Sobral et al. 2013; Vinodh et al. 2011b; Ecology 2007)
Cellular Manufacturing	(Chiarini 2014b; Fliedner 2008; Vinodh et al. 2011b)
TPM	(Chiarini 2014b; Fliedner 2008; Longoni et al. 2011; Marudhamuthu et al. 2011; Pojasek 1999; Sobral et al. 2013; Vais et al. 2006; Vinodh et al. 2011b)
SMED	(Chiarini 2014b; Kováčová 2013; Marudhamuthu et al. 2011; Moreira et al. 2010; Ng et al. 2015; Ecology 2007)
Supplier relationship	(Corbett et al. 2006; Fliedner 2008; Miller et al. 2010; Simpson et al. 2005; Vinodh et al. 2011b)
Six Sigma	(Calia et al. 2009; Fliedner 2008; Kadry 2013; Pojasek 1999; Vinodh et al. 2011b; Wilson 2010)
Statistical Process Control (SPC)	(Garza-Reyes et al. 2014; Torielli et al. 2011b; Wilson 2010)
Visual management (VM)	(Herrmann et al. 2008; Sobral et al. 2013; Vinodh et al. 2011b)
Analysis Tools	(Garza-Reyes et al. 2014; Langenwalter 2006; Maskell et al. 2008; Ng et al. 2015; Ecology 2007)
Standardized work/Qualification	(Chiarini 2014b; Herrmann et al. 2008; Kováčová 2013)
Plant layout reconfiguration	(Ecology 2007; Aguado et al. 2013)

Bhasin (2015) in his book provides 52 tools of Lean Manufacturing and outlined 25 of the essential tools used in the industry stating that the importance of the application and the type of Lean tools to be applied depends on the stage of Lean implementation attained by the company.

In Table I.3, it is worth noting the multitude and variety of these Lean practices. For some authors, these optimization techniques issued from Lean philosophy are called: “Lean Tools” (Arunagiri et al. 2014; Chiarini 2014a; Melton 2005). Other authors talk about “Lean Practices” (Hofer et al. 2012; Jasti et al. 2015) applied in the workshops and enterprises. In some other studies Lean Practices are called “Lean Factors” (Büyüközkan et al. 2015), and several other terms have been identified. Concerning the multitude of these terms, we can conclude that it is necessary to treat these "elements" as parameters for study and analysis. Whatever is the name of these elements, the goal remains the same, optimizing the operational performance of the company to improve the overall productivity and customer experience. These tools have a high importance for the flow optimization.

It is certainly necessary to develop a philosophy and consistent Lean thinking with the key concepts presented above. But one should not be reluctant to use the word "Lean tool". In fact, these are configurable and testable elements that could be analyzed for their technical improvements, their contributions, and their impact on performance. However, vigilance must be exercised in order to differentiate between problem solving tools, system management tools and applicable tools. Leandro-Elizondo (2018) stated that in the absence of a standard definition, several inconsistencies were found in different publications. For example, practitioners often confuse between managerial systems (Total Quality Management (TQM), Total Productive Maintenance (TPM), and JIT) and tools like 5S, Kanban, and Value Stream Mapping (VSM).

Table I.3 Various Lean technics Lean techniques found in the literature

Authors	Lean Techniques	Named	Main outcomes
(Bortolotti et al. 2015)	Equipment layout, JIT, Kanban, setup time reduction, statistical process control, autonomous maintenance	*Lean Constructs*	In order to have a successful implementation of LM, it is essential to go beyond the Lean constructs and technicalities by developing an appropriate Organizational Culture profile
(Büyüközkan et al. 2015)	Setup time reduction, Pull production/Kanban, small lot size, inventory level, continuous flow, Value Stream Map. process flow improvement, preventive maintenance, cellular manufacturing, 5S (order and cleanness in the plan), root cause analysis/5 Why analysis, employee involvement, continuous improvement/Kaizen, error proof/Poka Yoke, waste elimination	*Lean Factors*	Seven Lean factors were studied, and achievements were analyzed based on the flexibility, quality, reliability, and time operations performance indicators Lean techniques combinations have financial and non-financial consequences on the business performance
(Arunagiri et al. 2014)	9 Lean tools: 5S, OEE, 8Do, Pareto analysis, waste elimination, kaizen, setup reduction, process mapping, VSM	*Lean Tools*	It exists more than thirty Lean tools that can be applied in production. Each organization type uses a particular Lean tool to solve an existing particular problem. A survey of 91 samples in automotive industries has been conducted to find the most effective Lean tools
(Chiarini 2014b)	VSM, 5S, cellular manufacturing, SMED, TPM	*Lean Tools*	In this research, authors measured the environmental impacts after the implementation of five Lean tools. Quantitative results showed that VSM, 5S, cellular manufacturing, and TPM have improvements in the environmental impacts. However, there are no improvements in the environmental impacts after SMED implementation.
(Hofer et al. 2012)	Supplier feedback, supplier JIT, supplier development, customer involvement, Pull system, continuous flow, setup time reduction, statistical process control,	*Lean Practices*	Lean practices affect the financial and inventory performances. Implementation of concurrent internal/external Lean practices leads to a better performance than the selective LP employment.

	employee involvement, Total Productive Maintenance		
(Melton 2005)	Force Field Diagram, IPO diagram, process flow mapping, time value mapping, spaghetti diagram, 5 Whys, 5S, risk assessment, Kaizen, Kanban	*Lean Tools*	Lean is not only about Lean Tools and some changes in the manufacturing processes, it is also about people. Lean is a revolution; Lean tools are now applied in all over the world within different types of industry.
(Jasti et al. 2015)	Value Stream Mapping, setup time reduction, Kaizen, Kanban, Pull production, small lot size, JIT purchasing, elimination of waste, supplier involvement, Total Quality Management, Standardization of work, flexible information system, JIT, Takt Time, continuous flow, employee commitment, multifunctional employees, long-term supplier and customer relationship, top management commitment, Total Productive Maintenance, customer involvement, uniform workload, visual factory, cellular layout	*Lean Practices*	Lean practices should be applied to the whole activities of the organization and not only to the manufacturing field. Many of the organizations used some Lean practices to avoid few wastes instead of working to avoid all existing seven wastes. Academicians and professionals should collaborate to get more and better results for successful Lean implementations. Organizations need a systematic methodology to implement Lean Practices across all their activities.

Whatever terminology is used to qualify different Lean approaches (tools, practices, factors or techniques), the authors highlight the enablers to speed up the flow and regulate the materials through the added value chain from raw materials to final product. Value Stream Mapping, setup time reduction, Kaizen, Kanban, Pull production, small lot size, JIT purchasing, elimination of waste, supplier involvement, TQM, Standardization of work, flexible information system, JIT, takt time, continuous flow, employee commitment, multifunctional employees, long-term supplier and customer relationship, top management commitment, TPM, customer involvement, uniform workload, visual factory, cellular layout are the main Lean techniques that we can find in the literature (Jasti et al. 2015). For instance, the VSM is defined as being a Lean factor by Büyüközkan et al. (2015), Lean tool by Melton (2005), Arunagiri et al. (2014), and Chiarini (2014b), and Lean practice by Jasti et al. (2015). In our study, we will keep the Lean techniques vocabulary to qualify the different Lean tools.

It is interesting to highlight the results of the literature shown in Table I.4 and Table I.5. In the literature review of Jasti et al. (2015), which addressed their synthesis based on a sample of 546 scientific articles, and Marodin et al. (2013) that used a sample of 102 articles, there are Lean tools that are frequently studied and mostly tested in various works. As an example, in both studies, the pull system and the setup time reduction almost occupy the top ranking of these works. Those results reflect the link between the tools and the speed up of the flow. Both, setup time necessary to change batches and flow pulling to reduce the stock are directly responsible for the physical quantifiable gain in “time” and “storage level”.

Table I.4 Frequency distribution of Lean Elements mentioned in the articles shown according to the authors' sample (n=546) (Jasti et al. 2015)

LP tools	Conceptual	Descriptive	Empirical	Exploratory cross sectional	Exploratory longitudinal	Total	%
Value stream mapping	2	72	53	51	1	179	32.78
Set-up time reduction	2	73	47	48	1	171	31.32
Kaizen	2	59	61	43	1	164	30.04
Kanban	3	69	40	52	0	164	30.04
Pull production	1	62	37	36	1	137	25.09
Small lot size	1	53	33	47	0	134	24.54
JIT purchasing	0	59	31	37	1	128	23.44
Elimination of waste	2	57	23	45	1	126	23.08
Supplier involvement	1	50	19	52	1	123	22.53
Total quality management	0	65	15	39	0	119	21.79
5S	2	63	17	33	0	115	21.06
Standardisation of work	0	55	24	33	0	112	20.51
Flexible information system	0	66	11	30	1	108	19.78
JIT production	2	47	18	31	1	99	18.13
Takt time	0	44	34	12	0	90	16.48
Continuous flow	0	45	10	34	0	89	16.30
Employee commitment	0	39	8	38	1	86	15.75
Multifunctional employees	0	43	11	30	1	85	15.57
Long-term supplier and customer relationship	1	55	6	22	1	85	15.57
Top management commitment	1	38	5	36	0	80	14.65
Total productive maintenance	2	32	7	38	0	79	14.47
Customer involvement	0	32	4	38	1	75	13.74
Uniform work load	0	35	3	33	0	71	13.00
Visual factory	0	32	2	28	0	62	11.36
Cellular layout	1	28	4	28	1	62	11.36

On the other side, VSM is a very widespread approach because of its technical interest found in engineering journals and publications. Almost (32,78%) of the samples studied by Jasti et al. (2015) are mentioning VSM Lean technique. The other dispatching of Lean techniques is represented through the finding in Table I.5. Less attention is obviously given to other techniques, not because they are useless, but probably because they are not always interestingly explained, as well as they are not enough subject to calculation and assessment. Engineering journals probably prefer publishing more technical practices. Indeed, it is often boring for readers to see only the reconfiguration of plants thanks to "cellular manufacturing" which gets only 11,36% of attention in the studied sample (Jasti et al. 2015). However, the cellular layout is definitely an interesting Lean technique that optimizes the flow evolution with successive steps of production range avoiding transportation time waste.

The same remark can be made for the visual management. It is not obvious to find out scientific papers dealing with color code and visual transformations, even useful but not interesting enough to explain and comment as other Lean techniques closer to engineering field.

The simple statement is relating to 5S a very well-known Lean technique and very widespread in industrial reality but not really considered for publication and research community because of its recurrent applicative aspect.

Table I.5 Frequency of Lean practices according to Marodin and Saurin (2013) sample (n=102)

Lean practices	Doolen and Hacker,(2005)	Sezen et al.(2012)	Shetty et al.(2010)	Shah and Ward (2007)	Wan and Chen (2009)	Saurin et al. (2011)	Bhasin (2011a)	Soriano-Meier and Forrester (2002)	Gurumurthy and Kodali (2009)	Lasa et al. (2009)	Abdulmalek and Rajgopal (2007)	Shah and Ward (2003)	Karlsson and Åhlström (1996)	Panizzolo(1998)	Forza (1996)	Niepce and Molleman (1996)	Sum
Pull production / Takt time	X	X	X	X	X	X	X	X	X	X	X	X	X	X			14
Setup reduction	X	X		X	X	X	X	X	X		X	X	X	X	X		13
Total production maintenance	X	X	X	X		X	X	X	X		X	X	X	X	X		13
Smoothed (levelled) production	X		X			X		X	X	X	X	X	X	X	X		11
Teamwork	X			X		X	X	X	X			X	X	X	X	X	11
Cellular manufacturing	X	X	X		X	X	X		X		X	X		X		X	11
Workforce involvement in solving problems		X		X	X	X		X			X	X	X	X	X	X	11
Autonomation (Jidoka)	X	X	X		X	X	X	X	X				X	X			10
One-piece-flow (continuous flow)		X	X	X		X	X		X	X	X	X		X			10
Multi-functionality and cross-training	X			X	X	X		X	X			X	X	X		X	10
Standardized work	X		X		X	X	X		X						X	X	8
Visual management			X		X	X	X	X	X		X		X				8
Lot size reduction	X	X						X	X		X	X	X	X			8
Feedback of performance metrics (e.g., productivity, quality)					X	X	X	X	X				X		X		7
Workforce autonomy / empowerment			X			X	X		X			X		X	X	X	7
Root cause analysis for problem solving		X	X	X		X	X				X	X					7
Housekeeping (5S)		X				X	X		X		X			X			6
Value stream mapping		X	X		X				X	X	X						6
Concurrent engineering	X					X			X					X	X		5
Customer involvement			X	X	X									X	X		5
Suppliers involvement in the long term	X			X	X				X					X			5
Focused factory				X		X			X		X				X		5
Parts standardization / modularization	X								X					X	X		4
Statistical process control (SPC)			X	X					X						X		4
Feedback to suppliers	X			X					X					X			4
Ability and responsiveness to product mix variability							X					X		X	X		4
Workforce recognition and reward			X				X		X		X						4
Design for manufacturability	X								X					X			3
Quality at the source	X								X					X			3
Suppliers delivery JIT				X					X					X			3
Team leader						X		X					X				3
Safety at work							X		X			X					3
Commercial actions to stabilize demand	X													X			2
Line balancing					X				X								2
Multifunctional product development teams														X	X		2
Process standardization							X		X								2
Departments working integrated															X		1

At this step, the idea is to browse the different findings and literature consideration to identify the global Lean literature context. The positioning and the scope of the study will be led in the following sections of Chapter II where we will discuss the choice made to select the appropriate Lean tools in accordance with the objective of this thesis.

I.2 Impact of Lean implementation on industrial performances

The benefits of Lean implementation and its impact on industrial performance are widely studied in the literature. These benefits have been explored in various sectors (Voss 2005; Shah et al. 2007; Dickson et al. 2009) and the positive impact of LM on a firm's operational performance has been strongly argued (Moyano-Fuentes et al. 2012; Fullerton et al. 2014). The main impacts are summarized hereafter*:* *reduction of stocks, reduction in manufacturing costs* (Capraro et al. 2002; Baglin et al. 2000; Kilpatrick 2003; Shah et al. 2003; Melton 2005; Dickson et al. 2009; Demeter et al. 2011), *unnecessary processes elimination, productivity increase, quality enhancement, lead time reduction, cost reduction* (Karlsson et al. 1996; Sohal 1996; Ghosh 2013; Fullerton et al. 2014), *and space used reduction* (Kilpatrick 2003).

Marodin et al. (2013) point that 55% of studies used only operational performance measures, which reflects obviously the technical emphasis of the literature. Other authors explore an extension for the evaluation of the impact beyond the operational performances. The impact can be classified into three categories: operational, administrative, and strategic (Kilpatrick 2003).

Kilpatrick (2003) showed that Lean management led to a significant drop in the number of administrative documents issues and order errors, which proved the administrative gain. Regarding the literature review of Cherrafi et al. (2016), authors outlined that the main impact of Lean on business processes is evaluated financially. Moreover, they explore remarkably the environmental and societal aspects to extend the arguments of Lean benefits. A great interest in the impact of Lean practices on working conditions is argued by Marodin et al. (2013) where authors stated that "Companies have often improved metrics related to working conditions as a result of Lean implementation, although negative impacts have also been detected".

Moreover, the performance evaluation is mainly focused on the operations in the manufacturing phase (shop floors) disregarding the product development and resources management (Marodin et al. 2013). For example, in view of Lean practices, pull production was included in 87% of the methods, while concurrent engineering appeared in only 31% of them. These results can be justified by the fact that the main origin of Lean is linked to the manufacturing. Furthermore, manufacturers and production managers are mainly the main stakeholders of Lean approach until now. As for the product development, it is judged as being adapted more to Agile methods. It has not been well-developed yet in Lean practices because of the existing competitive approaches.

Marodin et al. (2013) suggest a classification into different categories: (a) operational, such as stock levels, quality, worker productivity, and setup time; (b) financial, such as cost, profit, and revenue; (c) human, such as stress, employee commitment, and safety at work; (d) market, such as market share; and (e) environmental, for example, pollution, resource efficiency, and the use of pollutant chemicals.

In Table I.6, we clarify the different categories of performance evaluation based on the literature. We extract the following synthetic overview; there exists obviously a dominance of the financial and operational performances.

However, it is also interesting to state that these findings are gradually moving from the restricted financial operational dimensions to other interesting factors such as staff motivation, quality, etc. This synthetic overview demonstrates that it is important to assess the aforementioned factors beside the cost aspect that is usually examined alone in the literature. This assessment will increase the success of Lean practices implementation.

Table I.6 Lean Implementation Impact on Performance

Category	Lean implementation impact on performance	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)	(17)	(18)	(19)	(20)
		Authors																			
Operational	Quality	X			X		X	X		X	X	X			X	X	X	X		X	X
	Delay		X		X	X	X	X		X		X	X		X	X	X			X	X
	Cost	X	X		X			X		X		X	X		X	X	X	X		X	
	Production efficiency		X								X		X					X			
	Inventory				X	X					X	X	X	X		X	X	X		X	X
	Cycle Time		X		X		X														
	Process											X	X			X					
	Sales turnover				X						X		X								
Administrative	Formalities						X														
Human	Labor Productivity	X	X		X		X				X					X					X
	Safety	X			X				X		X				X						
	Work Condition	X							X		X										
	Employee motivation				X						X								X		
	Employee empowerment			X	X																
	Customer satisfaction			X	X	X										X		X			
	Customer-supplier relationship										X							X			
	Service quality				X					X	X										
Environmental	Work Environment/Cleanness	X							X												
	Energy consumption	X																X			
	Space productivity	X					X									X		X			

(1) Fliedner (2008); **(2)** Shah et al. (2003); **(3)** Dickson et al. (2009); **(4)** Bhasin (2008); **(5)** Wan et al. (2008); **(6)** Kilpatrick (2003); **(7)** Behrouzi et al. (2011); **(8)** Saurin et al. (2009); **(9)** Agarwal et al. (2006); **(10)** Baglin et al. (2000); **(11)** Cortes et al. (2015); **(12)** Pay (2008); **(13)** Demeter et al. (2011); **(14)** Martínez-Jurado et al. (2014); **(15)** Gurumurthy et al. (2009); **(16)** Karlsson et al. (1996); **(17)** Melvin et al. (2008); **(18)** Suri et al. (1986); **(19)** Martinez Sànchez et al. (2001); **(20)** Anand et al. (2008).

I.3 Key success factors for Lean implementation

Many publications have been conducted to investigate the "Key Success Factors" in order to reveal the key points/factors that enable the good Lean conversion and lead to implementation success in the manufacturing systems.

Table I.7 Main key success factors for LP implementation

Authors	Key Success Factors											
	Culture	Leadership	Top Management	Training	Financial capabilities	Rewards and Recognition	Skills and Expertise	Involvement and Commitment	Customer Focus	Tracking and regular Audits	Quality analysis	External experts
Achanga et al. (2006)	√	√	√		√		√					
Jeyaraman et al. (2010)	√		√	√	√	√		√		√		
Kaye et al. (1999)	√	√	√					√	√	√	√	
Taner (2013)		√	√	√	√	√		√		√		
Fadly Habidin et al. (2013)		√	√						√		√	
Lande et al. (2016)	√	√	√	√		√		√	√	√	√	
Antony et al. (2012)			√	√				√	√			
Hibadullah et al. (2014)								√	√			
Netland (2016)		√		√		√		√		√		√
Alaskari et al.	√	√	√	√	√			√		√		
Antony et al. (2002)	√		√	√			√	√	√	√		
PQA (2003)		√	√	√				√	√			
Kundu et al. (2012)	√	√	√	√	√			√		√		
Laureani et al. (2016)	√	√	√		√	√	√	√	√	√		

Martínez-Jurado et al. (2014) qualified the key success factors as facilitators that smooth Lean practice implementation process. A summary with the major success factors is extracted from the literature and shown in Table I.7. Top management involvement, employee commitment, leadership and culture are about the main critical factors to consider. Bhasin et al. (2006) and Marodin et al. (2013) point out *managerial and cultural issues* as the most challenging obstacles in Lean Production (LP) implementation. "*The effective management of these factors, to the possible extent, is critical for successful LP implementation*" (Marodin et al. 2013). However, only one of the studied researches pointed about the importance of the external experts in smoothing the Lean transformation (Netland 2016). 21% of the studied researches consider that the skills, expertise, and quality analysis are critical success factors for a successful implementation.

Regarding the complexity of Lean practices implementation (Lian et al. 2007), one can expect the diversity and variety of possible factors positively influencing the implementation process. However, it is not enough to state and justify the existence of success factors. A broader analysis on why companies are successfully or not implementing Lean goes through investigating the influence of company's contexts on those factors. Moreover, the dynamics and intensity of factors' relationships should also be investigated. Indeed, there is a need for proposing countermeasures that could be used in managing and integrating the factors into the LP implementation methods (Marodin et al. 2013).

Based on the literature, Hu et al. (2015) conclude that there is a need for researching into Lean in the context. The authors address the need for research that spans beyond the boundaries of single organization, extending to supply chains and networks contexts. They recommend exploring the differences between small, medium, and large organizations to recognize how a company's size affects the implementation of Lean.

Obviously, the Lean success factors are playing the role of facilitators to increase the odds of Lean success. However, the roots, principles, and choices of relevant Lean implementation regarding the context, size, and network, seem to be wide and complex areas to explore. This complexity makes it difficult to assign the appropriate implementation to its real time environment.

I.4 Main Lean implementation Hindrances and Pitfalls

Studying Lean from success factors perspective is not significant enough (Kumar et al. 2008; Chin et al. 1993; Achanga et al. 2006; Timans et al. 2012). Barriers should also be tackled and analyzed in order to perceive the implementation troubles perspective (R. Jadhav et al. 2014). Studying both perspectives should lead to a smooth and successful Lean implementation process. Operations Management (OM) scholars discussed several causes of this lack of success, namely, the complexity of Lean implementation (Lander et al. 2007).

Although there are many companies that have adopted Lean production successfully, others have failed in the adoption process and have not achieved their goals (Bhasin et al. 2006). Findings from a study conducted in Chili showed that the main barriers faced by 77 enterprises in the deployment of Lean approach were the lack of time assigned to the implementation process, the lack of training and staff self-criticism, and the necessity of the research for improvement activities (Alarcón et al. 2005). Kumar et al. (2011) claim that any change initiatives will fail, in spite of the LM projects, if the organization is not culturally ready; several companies failed to achieve a superior performance. According to Lyonnet (2010), these disparities could also be related to the specificities of the context in which the Lean approach is deployed.

There are many obstacles that a Lean journey encounters (Henderson 2003). An excellent summary of obstacles is suggested by Bhasin (2015). See the following Table I.8. Browsing the literature, main obvious hindrances justifying the low number of successful Lean implementations are analyzed by Bhasin (2015). We may highlight the following items perceived as hindrances to which the literature suggests taking action: *lack of communication, lack of culture, manufacturing scope reducing the expansion of Lean thinking, expected standardized Lean implementation, complexity of context, reducing Lean to organizational problem, lack of strategic consideration, lack of compatible IT supporting Lean deployment, lack of value notion, and lack of checking objectives.* In front of each quoted hindrance, Bhasin (2015) provides a "rationale forwarded" column to counter the hindrances that can reduce the odds of a successful Lean conversion.

Table I.8 Lean implementation hindrances/ Rationale Forwarded (Bhasin 2015)

Literature explanations for the low numbers of successful implementations

Rationale forwarded	Literature sources
Improve the internal communications systems; required to aid empowerment and to adopt the principles of Lean	• Angelis et al. (2011) • Camp (2013) • Eisenhardt and Martin (2010) • Hines et al. (2008)
Need to observe Lean more than a manufacturing improvement strategy and allow its remit to surpass outside manufacturing	• Koenigsaecker (2005) • Liker (2004) • Shook (2010) • Spear (2004)
Effectively manage the sub-cultures; no company has a homogeneous culture and it is important to retain focus upon the Lean mission and vision	• Stefanie et al. (2012) • Wincel and Kull (2013) • Angelis et al. (2011)
Recognise that every Lean journey is distinctive; there does not exist a stable formula to achieve Lean success; and the respective companies commence with a dissimilar arrangement of constituents (or influences and restrictions)	• Sim and Rodgers (2009) • Johnston (2009) • Laureani and Antony (2012) • Bartels (2005) • Campell (2006)
Customised accounting procedures need to be adopted; both standard costing or activity-based systems are unable to accommodate the complexities of Lean. Preferably, value stream/product-based costing taking into consideration product development whilst vending alongside production and supplier costs is required; in this way, the personnel involved within the value stream are able to detect if they are influencing a greater degree towards value instead of costs	• Neely et al. (2005 • McVay et al. (2013) • Schonberger (2008) • Singh et al. (2010) • Tangen (2005) • Saurin et al. (2011) • Baggaley (2006)
Promoting the Lean paybacks; there is a sketchy record of organisations treating Lean as an business initiative	• Gremyr and Fouguet (2012) • Cocolicchio (2008) • Doolen and Hacker (2005)
Lean has to considered as a long-term venture and one whereby the benefits may not be obvious within the first year	• Wheatley (2005) • Cross (2012) • Fullerton and Wempe (2009)
Companies are required to adopt appropriate compatible IT systems; there exists a need to link the operational level to effective enterprise software proceeding to extend it to the customers' value chain	• Cross (2012) • Marksbury (2012) • Montgomery (2010) • Williams and Duray (2012)
Adapt the organisational structures; a definite requirement exists to shape in line with the "*value streams*" concentrating upon the customer and product groupings	• Radziwill (2013) • Mehta and Shah (2005) • Montgomery (2010) • Jones (2009)
A need to sustain the Lean momentum; it is essential that the company intermittently elucidates objectives for individual value streams whilst deducing the accomplishment disparity between the customers' requirements and the actual provision	• Wilson (2010) • Motley (2005) • Pullin (2005) • Ransom (2008) • Camp (2013)

As for Bhasin (2015), it is important to know that there exists no generic formula to follow during Lean implementation process and the main key elements that should be considered in order to counter the Lean implementation hindrances are:

- Refining the internal communications system
- Handling the sub-cultures efficiently
- Adopting custom accounting procedures
- Promoting Lean benefits which might not be evident to the company at first
- Employing the required compatible software systems
- Adjust the organizational structures
- Ensure Lean sustainability

Considering those potential hindrances of Lean implementation may constitute initial alerts. These alerts can make the managers aware about the key points to consider at an early stage of the implementation in order to avoid mistakes and disruptions occurrences during the Lean deployment process. Considering those key points early may be as beneficial as considering success factors, and even better because excluding risky situations will definitely increase the possibility of successful Lean transformation.

A good internal communication is required for Lean principles adoption. In other words, the oral or written dissemination of correct information, the sharing of opinions, the establishment of a good communication, and the creation of a win-win situation are required. Considering these points smoothly increase the chance of Lean project sustainability. Moreover, creating a culture based on Lean philosophy leads to a positive and nice environment within the department or the company. After the Lean culture initiation, it becomes easier to adjust the organization structure. Therefore, the big groups initiating the Lean culture create an appropriation of the global Lean system (called: PES - Peugeot Excellence System, BPS - Bosch Production System, EPS - Eurocopter Production System). The good initiation of Lean culture strategically determines the global success of Lean implementation.

Lean management is more than just an elimination of waste; it requires a new approach, in which the customer and the maximum added value are taken into accounts. To achieve an organization that thinks about continuous improvement day after day, it is necessary to have sufficient creativity in the organization; moreover, this creativity and innovativeness must be properly guided. Furthermore, employees should get involved all together to achieve the objectives of the team and the entire organization. One of the difficult issues faced during the implementation is the employees' behavior. To influence the behavior of employees, managers should take the skills of the individual and the team elements into consideration. Skills and behavior of individuals are interrelated. With good employees' skills, the team will get involved in common objectives. Moreover, employee's self-criticism and responsibility are essential and should be taken into consideration at an early stage of the implementation in order to avoid implementation mistakes.

I.5 Multiple Case Studies in different Sectors

Table I.9 shows multiple case studies on Lean implementation in multiple sectors and different countries; three case studies on aerospace products, three on automotive products, and five others on different other sectors (Textile, health, electronics and food). Based on these studies, we have realized that company's culture, top management commitment, and employee involvement were the most critical success factors for a Lean implementation. Vinodh et al. (2011c) and Hodge et al. (2011) considered that management commitment had to be ensured before the implementation of a proposed Lean project. Thomas et al. (2016) and Kumar et al. (2006) found that convincing top management was the most difficult task during the implementation. Herron et al. (2008) agreed stating that management support and willingness to accept the change are the most important factors to be considered. Managers' resistance to change was a noticed factor in many studies (Kumar et al. 2006; Hodge et al. 2011; Antony et al. 2009) and in some cases, this required the introduction of new vice presidents to change the leadership attitude (Ferdowsi et al. 2002b) consisting in some cases of a barrier leading to implementation failure (Thomas et al. 2016; Vlachos 2015).

From the other side, employee involvement is also an important factor for implementation success; employees should be motivated and involved in Lean implementation. They need a human resource environment that permits and encourages freedom of thought, involvement in ambitious goals, and individual expertise contribution to the Lean implementation (Ferdowsi et al. 2002a). In order to reach such an environment; trainings, resources, knowledge, and authority to solve problems are crucial factors for the success of the project (Wang et al. 2012). In addition, involving employees as much as possible in the planning, implementation and evaluation of changes as well as developing superior capabilities will ensure survival of the firm in the long term providing the employees with a feeling of job security (Sohal 1996). In many implementations, employees were resistant to change (Kumar et al. 2006; Hodge et al. 2011; Vinodh et al. 2011c), because they thought that the implementation of the new strategies could endanger their job opportunities especially in cases of poor performance (Kumar et al. 2006).

In the listed case studies of Table I.9, we note that VSM, 5S, and Kanban/Pull strategy tools were the most frequently used techniques. We can clearly see that whatever sector or country Lean is implemented in, there are common points or factors that affect Lean implementation and have a significant part in leading the Lean implementation to success or failure. *"Lean is a journey, not a destination*" (Bhasin 2015). Therefore, a successful Lean implementation in any country or sector needs a continuous respect of the principles. It also needs a significant effort to convince both management and shop floor personnel that Lean is an effective approach that requires time and commitment. In addition, few benefits are noticed in the short run, substantial results are possible in the long term (Sohal 1996). The results are noticed in the long-term run as the implementation often requires a change in culture and considerable experience (Vinodh et al. 2011c). Sohal (1996) and Ferdowsi et al. (2002a) considered that a Lean Champion is needed to drive the change initiatives and to provide the leadership for Lean transformation.

Table I.9 Case Studies in multiple different sectors

Sector	Authors	Key Success Factors	Tools
Aeronautic	(Ferdowsi et al. 2002a; Thomas et al. 2016; Ferdowsi et al. 2002b)	Leadership Top Management Human resources Communication Teamwork Employee empowerment Employee involvement Training Pilot projects	VSM, TPM, 5S, Kanban/Pull System, Standardized work, Kaizen Continuous Improvement, On-Time Delivery (OTD), DMAIC
Automotive	(Kumar et al. 2006; Sohal 1996; Vinodh et al. 2011c)	Culture Top Management Employee involvement Work environment Communication Training	VSM, 5S, TPM, Kanban, JIT, Kaizen, SMED, Cellular manufacturing, Statistical process control, DMAIC, 5 Why's Analysis
Textile	(Hodge et al. 2011)	Culture Top Management Employee involvement Training	VSM, 5S, TPM, Kanban, SMED, Cellular Manufacturing, Kaizen, Jidoka, Poka-Yoke, DMAIC, PDCA
Health	(Crema et al. 2015)	Culture Leadership Managers commitment Continuous application Training	5S, Kanban, One-piece flow, PDCA, 5 Why's analysis
Electronics	(Wang et al. 2012)	Culture Managers commitment Human factor Training	VSM, 5S, 5 Why's analysis, DMAIC
Food	(Vlachos 2015; Lehtinen et al. 2005)	Business Culture Leadership Top management Expert Knowledge Operational easiness	VSM, Kanban/ Pull Strategy, Activity Process Map

I.6 Lean Soft and Hard practices

Yang et al. (2012) sent a questionnaire to 620 companies implementing Lean techniques to build a structured hypothesis about Lean techniques adopted in different industries. 151 of these companies answered this questionnaire. Based on companies' answers, authors revealed that different Lean techniques was implemented in these companies and some of these companies are only focusing on the technical approaches of Lean disregarding the Human factor. In this survey, the motorcycle and automobile industries did a significant effort to apply human resources practices and obtained good results. However, most of the other industries are ignoring the human elements of Lean Manufacturing. For managers and executives, results showed that it is essential to go beyond Lean technical aspects by adopting the human-related practices in order to have a successful Lean transformation (Yang et al. 2012; Mamat et al. 2015). Those results were grouped by Yang et al. (2012) into nine categories, four categories are directly related to the technical elements used during Lean implementation and four categories are related to the human elements that should be taken into consideration during any Lean implementation. See Table I.10.

Table I.10 Comparison of technical and human elements in Lean (Yang et al. 2012)

Technical Elements	Human Elements
□ Autonomation	□ Utilization of People
• Autonomous Control	• Teamwork (Cross-functional)
• Automation	• Optimized Capability
• Built-in Quality Control	• High Involvement
• Preventive Maintenance (Defect Prevention)	• Multimachine Handling
• Poke Yoke (Error-Proofing)	• Collaboration
□ JIT	□ Flexibility
• Kanban System (Visual Control)	• Multiskilled Workers
• Standard Operations	• Direct Authority
• Production Smoothing (Production Leveling)	• Empowerment
• Quick Setup Times	• Enlarged Responsibility
• Lot Size Reduction	• Decision Making at the Lowest Appropriate Level
• Continuous Flow Production	□ Practices of HRM
• Taking Time	• Employee Education and Training
• Cycle Time Reduction	• Motivation
• Supplier Partnership	• Promotion of Leaders Internally
• Using Few Suppliers	• Development of Lean Leadership at All Levels
• Pull System (Customer Demand Pull)	• Relationships of Mutual Trust and Commitment
• Lowest Inventories	• Job Satisfaction
□ Lean Manufacturing	• Autonomous Decision Making
• U-type Manufacturing	• Work Enrichment
• Cellular Manufacturing	• Ongoing Development of People
• Multifunctional Layout Design	• Decentralized Responsibilities
• Single Minute Exchange of Dies (SMED)	□ Creative Thinking
• Multifunctional Teams	• Creating Value
• Simultaneous Engineering	• Capitalizing on Employees' Ideas and Suggestions
• Vertical Information Systems	• Maintaining Challenges to Existing Processes
• Effective R & D	• Pursuing Perfection
□ Waste Elimination	• Innovative Activities
• Focus on Problem Solving	□ Respect for People
• Group Improvement Activity	• Lifetime Employment
• 5S's	• Pay Graded Steeply by Seniority
• Lowest Inventories	• Treatment of Employees as Family
• 5 Whys	• Employees have Decision-making Power
	• Sharing the Company's Success

Technical and human practices are sometimes referred as hard and soft tools in the literature. In order to have a successful Lean transformation, Lean plants should implement the soft or human practices (i.e., Employee education and training, group problem solving, employee empowerment and involvement, teamwork and collaboration, respect for people, Innovative activities, good working environment, etc.) along with the Hard or Technical practices (i.e., Kanban, SMED, cellular manufacturing, poka yoke, continuous flow production, lot size reduction, simultaneous engineering, 5S, etc.) (Yang et al. 2012).

Shah et al. (2007) showed that Lean success is a result of a complex system of interrelated socio-technical practices, reminding that Lean implementation becomes effective by joining hard and soft practices. Sorooshian et al. (2017) clarify the "soft" Lean tool by defining it as a human-related tool or technique. Some of the soft Lean tools are listed hereafter; worker empowerment, leadership, culture, training, group problem solving, employee involvement, and multi-skills. For the process-related tools defined as hard Lean tools, Sorooshian et al. (2017) listed the 5S, continuous flow, standardized operations, reorder point, and supplier development. Mamat et al. (2015) considered that most of the industries were always focusing on hard Lean tools and disregarding the high importance of human-related tools, known as soft Lean tools. Soft Lean tools should support the hard Lean tools in order to have a successful and sustainable Lean transformation (Hines et al. 2004; Bhasin 2012b; Bortolotti et al. 2015). Moreover, Liker et al. (2011) stated that soft practices are the key factors that allow the company to improve its competitive advantages, especially when companies face a high market competition. (Radam et al. 2008) and (Shah et al. 2007) emphasize the importance of the soft factors such as the human resources management, top management/employee commitment, reward and recognition, and good communication in order to have a successful Lean transformation.

It is interesting to see both existing approaches, the technical/hard and human/soft approaches, that will give us a better vision on the types of existing Lean tools in the industry. This part will help us to make the right choice of tools and developing our thesis hypothesis from an engineering point of view. Actually, our focus is to find the tools that can be technically simulated in order to show their relevancy based on some studied industrial contexts and objectives.

I.7 Main Research Interests in Lean

The objective of this section is to highlight the main research interests in Lean that were recently published in the literature. We chose to treat the five following papers that browsed cumulatively more than 800 papers dealing with Lean practices implementation. We summarize the most important findings of each author in the column concerned. An analysis of common factors among the references reveals some common points necessary to highlight. In Table I.11, we used different symbols to point the similarities between the suggested research interests.

- **(ω)**: Factors that affect Lean implementation
- **(λ)**: Further research and studies suggested
- **(φ)**: Develop new methods to help in the integration of Lean
- **(Ø)**: Developing measures and metrics
- **(σ)**: Further research on the supplier and supply chain
- **(β)**: Expanding Lean implementation to organizations and product development

Table I.11 Main research interests in Lean Implementation

	Marodin et al. (2013)	Jasti et al. (2015)	Marshall (2015)	Panwar et al. (2015)	Cherrafi et al. (2016)
Sample	**n = 102**	**n = 546**	**n = 43**	**n = 104**	**n = 118**
Year	**1996 -2012**	**1988-2011**	**1988-2013**	**1989-2013**	**1990-2015**
Main research interests	Extend research about factors that affect LP implementation including investigation factors and relationships **(ω)** Involve application of methods to provide generalizability **(φ)** Balance the implementation of LP with technical emphasis and the practices that have effect on human, organizational aspects **(ω)** Extend Lean in other areas such as product development and services not only shop floor **(β)** Develop Performance measurement related to different dimensions such as human and financial **(Ø)** Extend research about detailed investigation of LP implementation that had unexpected results **(λ)**	Develop conceptual models related with surveys **(λ)** Extend Lean approach from operation to Lean enterprise **(β)** (337 over 401 companies apply LM instead whole) Extend the research to deal with all types of NVA (wastes) Develop a measure of the model performance **(Ø)**	Extend research about sustainability in Lean**(λ)** LP in service, non- profit organization **(β)** Supplier and supply chain research **(σ)** Use empirical method other than a field research **(λ)**	Develop Lean implementation model for process industry **(φ)** Develop analytical models to quantify the leanness measure of process industries **(Ø)** Extend research to supplier involvement in process industries **(σ)** Extend research to find out which tool result in what effect in process industries. Develop framework to overcome constraints for continuous process industries **(φ)** Conduct further empirical studies **(λ)** Contribution of external factors such as social economic, political and environmental factors **(ω)**	Develop integrated metrics to measure Lean/Six Sigma from *social*, *environmental*, *economic aspects* **(Ø)** Develop integrated model applicable to many sectors **(φ)** Expand research about service industry **(β)** Expand implementation of Lean/ Six Sigma and sustainability to emerging and developing countries Develop pre-implementation phase. **(φ)** (helps to implement Lean/ Six Sigma and sustainability successfully) Expand study to all functions of the supply chain with an analysis of supplier, customer relationship **(σ)** Extend research to motivation, barriers, negative effects of integration **(ω)**

When analyzing the research interests, we may outline the factors that affect the Lean implementation success. In (ω), the factors that influence Lean implementation are evoked. Those factors can be human or organization aspects (Marodin et al. 2013). Panwar et al. (2015) have checked for external factors that influence the implementation such as social, economic, political, and environmental factors. Cherrafi et al. (2016) extend the analysis to the negative factors (barriers) that should be considered in order to have a successful Lean transformation.

The existence of researchers working on the development of new Lean approaches may seem strange for the field community. The whole sample encourages the Lean researcher's community to commit in new developments (λ) of new conceptual models related to surveys (Jasti et al. 2015). For Panwar et al. (2015), "Conducting further empirical studies is welcome". Marshall (2015) indicates that more research is needed in Lean sustainability and that researchers should focus on empirical method and not only on the field research that mainly leads to statistical findings without having solid research hypotheses.

New methods are definitely expected (φ); developing a Lean implementation framework to overcome constraints is well promised area (Panwar et al. 2015). Involving application of methods to provide generalizability is also outlined (Marodin et al. 2013). Developing an integrated model applicable to many sectors and developing a framework for the pre-implementation phase are argued by Cherrafi et al. (2016). In his recent PhD thesis, Leandro-Elizondo (2018) suggested a methodology for evaluating performance of industrial process' continuous improvement.

In addition, the Lean implementation is not an isolated activity. The monitoring is necessary to assess and evaluate the level of Lean maturity implementation. This concern is considered as an important point outlined by Marodin et al. (2013) for developing performance measurement related to different dimensions such as human and financial. Other authors outlined the importance of developing Lean metrics and measures; measures to calculate the model's performance (Jasti et al. 2015), analytical models to quantify the leanness measure of process industries (Panwar et al. 2015), and integrated metrics to measure Lean/Six Sigma in social, environmental, and economical aspects (Cherrafi et al. 2016). The degree of Lean achievement is a growing interest for research community looking for synthetic and representative methods for Leanness degree calculation (Amrani et al. 2018).

In the studied samples (σ), supply chain aspect has also an extreme importance. Lean community needs more studies and researches in the supply chain domain (Marshall 2015). Expanding the research to all functions of the supply chain with an analysis on supplier and customer relationship is highlighted by Cherrafi et al. (2016). Extending research to supplier involvement in the process industries is outlined by Panwar et al. (2015). The idea of extending the research studies beyond the scope of production is obviously stated and confirmed by many authors; Extending Lean in other areas such as product development and services areas and not only shop floors areas is reminded by Marodin et al. (2013). Extending Lean approach from the operation to the whole enterprise (Jasti et al. 2015). Expanding the research to the service industry is stated by Cherrafi et al. (2016) and Marshall (2015).

Regarding the different research interests, one can observe the possible growth of scientific interests to provide the community with new approaches, new frameworks, and new metrics in order to evaluate the system's performances. The scale of research is also switched from simple production scope to supply chain scope and from production/manufacturing lines to engineering and service departments.

Many new conceptual models, structures, and definitions of Lean implementation are still required. The aim of this thesis is to contribute in consolidating the research findings and interests.

I.8 Problematic and research assignments

I.8.1 The Use of Lean practices: what is the problem?

The industrial reality reveals that most activities in operations management are interdependent. When trying to assess a Lean practice impact on an independent activity, it becomes quite difficult to assess the effect of the considered individual Lean practice. Assessing exactly which Lean practice has resulted in what improvement becomes a very complicated task according to Bonavia et al. (2006) and Pool et al. (2011). Panwar et al. (2015) argued that further research is required to systematically explore the effect of Lean practices in "process industries", in particular. In fact, there is a shortage of studies that identify the appropriate benefits of each Lean practice/tool.

The Lean tools have to be adapted to the context. How to achieve this goal was clearly stated but not remarkably defined by Arunagiri et al. (2014). Likewise, Dora et al. (2015) emphasize taking into account sectors specificities when dealing with the implementation of Lean practices. We noticed that an increasing number of authors are subscribing to the idea of considering the context, the sector, and environmental situation while implementing Lean practices.

The latter elements influence the use and efficiency of the chosen Lean tools. Cua et al. (2001) stressed the importance of using JIT, TQM, and TPM simultaneously when implementing LM. TPM tools play significant role to prepare the right environment for efficient adoption of JIT and TQM techniques (McKone et al. 2001; Mackelprang et al. 2010). JIT and Kanban production was born in order to respond to market competitiveness (Jagdev et al. 1998). Four main interrelated practices, the TPM, TQM, JIT, and Human Resource Management (HRM) contribute significantly to the operational performance of the plants (Shah et al. 2003). It exists a positive relation between the 5S tool and some factors/contexts such as the product type, the plant size, the technology used, and others. In addition, 5S has a positive influence on some operational performance measures (Bayo- Moriones 2010).

Regarding these elements, we notice that using Lean techniques independently from the company's situation, enterprise's size, or the demand context may lead to disappointment. Many Lean techniques seem interdependent and correlated with specific situations. The research idea that we are developing in this thesis is raised from the aforementioned observation. The environmental context regarding the demand, the type of products, and the company may influence the choice of the Lean techniques to use.

For the moment, no correlation and no interrelations are neither obvious nor easy to guess. We would like through this thesis to initiate a structured and gradual analysis in order to clarify the Lean techniques and their relevancy to the surrounded context. In the literature, relevancy of contexts to some dedicated tools has been already studied. Kanban is well applied when the demand is stable, low number of references to deal, and the range of production are well defined with clear identified sequences and steps (Vlachos 2015). The context adaptation for tools is already considered but the tools adaptation to the context really less.

I.8.2 The relevance of Lean techniques: Browsing the literature

Browsing the literature, we emphasized previous studies that have shown that many actions can hinder the performance: using the wrong Lean practices, incorrect implementation of correct Lean practices, or wrong order of use of Lean practices. This misuse of Lean techniques induces Lean failures and incur losses for the firms (Abdulmalek et al. 2007). Smart et al. (2003) argue that dealing with contextual uncertainty and non-routine behaviors, such as unexpected changes to customer requirements, represents an interesting approach that brings closer the adoption of relevant tools to the environment.

Dora et al. (2015) identify Lean implementation approaches .While defining what they called “step3”, they remind the importance of preparation at the organizational level and the importance of making alignments with the sector-specific factors; a firm can choose the appropriate Lean practices to implement for high performance. For instance, in food processing sector, Small and Medium Enterprises (SME) avoid applying the pull system and JIT tools because of the uncertainty of demand variation. Likewise, Ohno (1988) has emphasized that Kanban can only work effectively if the flow is optimal. The lack of flexible and multiple-use equipment in resource-constrained companies, like SMEs, was found to negatively affect the implementation of cellular layouts. This correlation is perhaps intuitive, but there is a shortage in studies when it comes to the relevancy of Lean tools. According to researchers, there is a basic logical sequence in which these elements should be generally implemented. For instance, Shingo et al. (1989) have found that Single-Minute Exchange of Die (SMED) and Layout Improvements should be implemented before thinking about the Kanban and flow. A Kanban system when operational in an environment of fluctuating demand would be regarded as waste (Womack et al. 2003).

Similarly, Smalley (2004) has highlighted that Lean tools cannot be implemented randomly and that there is a necessity to implement some Lean tools before others. The stability improvements were claimed to be prior in consideration (manpower, machines, materials and methods) during Lean implementation. Standardized work and uninterrupted process flows are the key foundation stones of the TPS (Ohno 1988). It is important to consider that easy-to-use practices such as workplace organization, visual management (VM), and customer involvement should be given more prominence in the beginning stages of the implementation than the more advanced ones: line balancing, one-piece flow, pull and Kanban. It has been proven, especially in SMEs, that initial quick wins and success help firms in sustaining a quality initiative (Radnor et al. 2008).

This is an interesting notion for prioritizing Lean techniques. When implementing Lean techniques in a global Lean thinking approach, the techniques are chosen based on the managers’ experience and the intuitive thinking of the project managers. What the research provides with the existing literature examples is *a beginning of the "relevancy" structuration of Lean techniques to one context rather than another* and to one situation prior to another.

Shah et al. (2003) studied the influence of the plant size, the unionization status of the company, and the plant age context-factors on 22 Lean practices that are mostly used in Lean manufacturing systems. Recently, in the paper of Bortolotti et al. (2015), authors stated that it is not only the choice of Lean practice that influences Lean manufacturing measures but also, the situation and context; complexity of products, production typology strategy, and demand variability . Obviously, the correlation between the context and necessity of studying the relevancy of Lean practices imposes itself to the community of researchers.

I.8.3 Adapting the Lean techniques choice according to the context

Beside the success of Lean, there also exists examples of Lean failures. Some negative impacts on performance may appear due to the complexity of implementing Lean because of possible negative synergies between JIT tools and techniques (Mackelprang et al. 2010). Implementing JIT without relevantly considering a coherent long-term manufacturing strategy (Matsui 2007; Agarwal et al. 2013) is quickly becoming a trap. Bhasin (2012a) states the fundamental prerequisite to ensure the suitability of the techniques put into practice to fit the right circumstances in a manner that proceeds to support the organization's value chain (Bicheno 2008).

In the research of Angelis et al. (2011) and Laureani et al. (2012), authors argued that any organization willing to implement Lean should be careful to implementing the vital tools in priority.

- Cellular structures, since it is imperative that the requirements to produce a product(s) are grouped closely for efficiency (Lee 2007) .
- Kanban methodology needs to be fully embraced (Smalley et al. 2009).
- Kaizen, which focuses upon the constant quest of advances in quality, cost, delivery, and design.
- Single-piece flow systems to be adopted need to be geared towards adding value (Bartels 2005).

This needs to be combined with process mapping that indicates the product and information flows (Jones 2009). Fullerton et al. (2009) explain Lean manufacturing as a *"long, arduous process that can be both problematic and beneficial depending on differing contextual factors".* However, in their review of empirical studies on Lean implementations and their effects on performance, Camacho-Miñano et al. (2013) conclude that evidence examining how and whether contextual factors impact the relationship between Lean practices and financial performance is inconclusive.

The research possibilities remain open in this explorative field as we can see through the different mentioned elements.

Based on these findings, we would grasp the opportunity to build a theory on Lean tools relevancy to specific industrial contexts. These first elements provided by the literature consolidate the possible existing gap because of the inconsistent yet methodologies. The idea is to build up a structured and consistent methodology for improving the relevancy of the chosen Lean techniques.

I.9 Research Questions identified: RQ.1, RQ.2, RQ.3

All of the points that we developed in Chapter I lead to a set of research issues presented below in the form of scientific questions.

- **RQ.1 What are prior Lean techniques to apply?**

There exist a wide range of Lean tools to use during Lean implementation. However, some Lean tools, such as pull and SMED tools, are more frequently used. For some researchers, some techniques should be implemented prior to others in order to achieve a better performance. The stability improvements (manpower, machines, materials and methods) were claimed to be prior in consideration during Lean implementation. Some researchers consider that easy-to-use practices (visual management, customer involvement, workplace organization, etc.) should be taken into consideration at the early stages of the implementation, before implementing the advanced tools. Others consider that any company willing to implement Lean should implement vital tools in priority.

For the moment and regarding the state of the art in the panel of Lean techniques found in the literature, not all tools deserve to be applied. Among the wide range of techniques available in the hands of manager, a priority seems to be adequate approach. We can consider that this research question has been addressed by the literature to confirm the necessity of using partial Lean techniques according to the need. Even in unstructured and not deterministic way, we can consider that this research question is partially covered by the literature.

- **RQ.2 Is the economic context considered by managers for choosing Lean techniques?**

Managers and engineers are in continuous search for supported methodology and cross analysis for effective Lean use. Considering that Lean brings benefits despite the context is a trap. One of the major challenges that managers face is the difficulty to choose the real tools that best fit their company and lead towards better productivity and quality. Managers are not taking into consideration the context in which Lean tools should be applied. However, there exists a relation between the context and Lean techniques. This constitutes a hypothesis in this research. Obviously, this imprecise step is undetermined yet, but the first elements found in the literature consolidate this research path to identify and analyze the possibility of contexts influencing the choice of the prior Lean techniques to implement. Indeed, the scientific context is targeting to test various configurations revealing the beginning of context influence. Furthermore, using the right tools in a convenient context reflects the company’s profitable or poor implementation of Lean.

The remaining part of this thesis will answer this question in detail by providing a more structured approach based on a simulated model to support the managers and decision-makers in taking the right choices and priorities for Lean techniques implementation. The next developments will reinforce the idea of Lean techniques relevancy. We suggest dealing with that point in the next chapters to analyze in depth the possibility of answering this research question in detail.

- **RQ.3 Are the objectives directing to the right Lean techniques' choice?**

Until now, only the hypothesis of context influence is well argued in the literature. However, in our opinion, the context alone is not sufficient to choose a Lean technique rather than another. We can exclusively notice that one tool is more adapted than another without a structured global view analysis.

The choice of Lean technique in the global Lean thinking development is always directed by strategic objectives that the company has to draw on its roadmap. We believe that objectives are also valuable in helping to define the undertaken path of optimization and improvements. As the objectives are not yet claimed to be powerful and affecting the choice of tools and as the context already succeeded in this matter, we suggest answering this third research question in the following thesis development.

We argue the necessity to analyze how the context and the objective can direct the right Lean technique choice for sustaining the implementation.

I.10 Conclusion

This chapter has introduced the background of Lean manufacturing and Lean thinking in the literature. Different elements have been presented and analyzed; Lean implementation impact on performances, Key success factors, and main implementation hindrances and pitfalls.

Comments and observations of many authors push us to manifest the influence of Lean approaches on companies' performances.

Since customers became more demanding seeking for high quality products and taking into consideration companies' restricted budget and the increasing market competition, firms from all sectors, industries and services, are increasingly adopting Lean. However, implementing Lean at any level of the company is a complex task that requires a good understanding of its fundamentals. New measures and metrics should be developed for more precise and specific results leading to a greater outcome. Therefore, main factors should be considered regarding the management involvement, employee commitment, leadership, culture, and others. Moreover, Lean tools cannot be implemented randomly and should be adapted to the company's context.

Future development is needed in next chapter to expand new methods in helping with the integration of Lean by using the relevant Lean techniques according to contextual situation and defined objectives in the strategy of the company.

CHAPTER II. Research Methodology: Combining Production Contexts and Industrial Objectives for Lean Techniques Adoption

It is clear that the changing business environment is prompting decision-makers to take innovative approaches and look for new and competitive position. Lean implementation inspired from TPS is part of this. We are witnessing a change in the practices of some industrial companies with a voracious desire to apply the Lean approaches into their companies. Indeed, the margins generated by manufacturing operations tend to be reduced in favor of distribution design and after-sales service operations. Competition from countries with low labor costs threaten the competitive position of production companies (Baglin et al. 2004). These elements certainly push the application and appropriation of Lean approaches to improve the production flow and satisfy customers. Moreover, a reflection on the adaptability of the techniques and maintaining their sustainability in the enterprise is also necessary.

Today, a particular priority must be given to identify the factors affecting the adoption of Lean in companies (Achanga et al. 2006; Mann et al. 1995). There is a major research stream identifying facilitators or inhibitors when implementing Lean (Karlsson et al. 1996; Hines et al. 2008; Fullerton et al. 2009; Serrano Lasa et al. 2009; Bruun et al. 2004). However, few studies have analyzed the causal factors: why companies are adopting Lean production (Sohal et al. 1994; Kojima et al. 2004) and what preconditions are needed to manage this adoption (Achanga et al. 2006; Kochan et al. 1997). Identifying why and how companies adopt Lean production is therefore fundamental to ensuring the success of adoption. In fact, having a prior knowledge of these explanatory factors before starting Lean implementation is of upmost importance (Hines et al. 2004), as this could mean faster progress with fewer obstacles in the implementation process (Sohal et al. 1994) and fewer correction loops. For these reasons, the questions in this study focus on identifying the factors that influence the adoption of one Lean practice instead of another. The methodology developed in this chapter is based on two major issues for any company: The economic context in which it is embedded, and the industrial objectives defined in its strategic policy. The alignment of these two issues can be established through an adequate evaluation of the use of Lean practices in the production system.

In this chapter, we aim at answering the two research questions RQ2 and RQ3. Therefore, the technical and computing supporting developments will be evoked in Chapter III. Chapter II is dedicated to the development of the methodology built to sustain the relevancy of Lean techniques in various economic contexts subject to different objectives.

II.1 Lean production techniques to include in the study

Several Lean tools derived from the TPS are applied in manufacturing and praised in the literature. We note that researchers select Lean tools based on their needs, sensitivities, and strategies.

However, the choice of the Lean tool is rarely justified or defended. In what follows, we identify the different possibilities available nowadays for managers and decision-makers. Very little works studied the possibility of prioritizing between tools. Marodin et al. (2013) list around twenty researches that selected a set of tools for evaluation in the production workshops. It is interesting to note that there exists a variability and a disparity in the authors' choices regarding the number of practices deployed, ranging from four or five tools (Shah et al. 2007), to almost thirty tools (Cherrafi et al. 2016). The number of Lean techniques used does not indicate their success or their consistency. There is a tacit adaptability of methods to needs. Abdulmalek et al. (2007) proposed a Lean implementation approach for process industries but used only seven Lean tools. Thus, we prove that the number of tools is adapted to the needs of the researchers and their research interests. In addition, the number of tools derived from Lean is immense. It is very difficult to use all existing tools for the same research question.

Table II.1 Lean tools retained for this study

Main Practices	Definition	Authors
VM / 5S	▪ The aim of VM is to make a self- explaining, ordering and improving workplace. ▪ VM mainly focuses on visualizing information and displaying requirements and errors in order to have a good understanding of the work area and the associated processes. This will deliver an effective solution to improve communication and information flow in shop floor. ▪ The use of Andon Boards (illuminated displays) that provide information about the actual production status. ▪ VM tools are essential to communicate the requirements in order to improve production efficiency. ▪ 5S is a type of VM, it is a set of principles that improve the workplace environment which in turn improve the quality of life at work. 5S pillars are Sort, Set in Order, Shine, Standardize, and Sustain. - **Sort**: Clear the work area from unnecessary and unwanted objects in each station or production operation. - **Set in Order**: Organize, put objects in order, and label here each item must be stored in order to easily find and place the objects. - **Shine**: Maintain a Clean workplace and neat environment. - **Standardize**: Use visual cues (Signs, scoreboards, placards, etc.) in order to have consistency in the operational outcomes. - **Sustain**: Daily follow-up to maintain the above listed pillars. Achievements are non-durative without the sustainability pillar.	Cherrafi et al. (2016); Parry et al. (2006); Eaidgah et al. (2016); Bayo- Moriones (2010); Al- Araidah et al. (2010); Omogbai et al. (2017); EPA (2017); Fernando et al. (2007); Taggart et al. (2012); Chapman (2005); Grief (1995)
Pull / Kanban	▪ The pull system approach is to produce the exact quantities only on demand to reduce the work in progress, to eliminate potential wastes, and to reduce the floor area utilization. ▪ Pull system focuses on stopping the overproduction and increasing the flexibility of responsiveness to the market demand. ▪ Kanban is a LM tool that controls the levels of inventories in the production system on a JIT basis. When the inventory buffer reaches the maximum preset level, a signal is sent to the upstream workstation to stop the production.	Rahman et al. (2013); Arbulu et al. (2003); Cherrafi et al. (2016); Vinodh et al. (2011b)

	▪ A specific type of Kanban is the supplier Kanban when a signal is transmitted to the outside suppliers asking them for materials replenishment.	
SMED	▪ SMED stands for Single Minute Exchange of Dies. Its purpose is to reduce the setup/changeover time of the machine ▪ Setup Time is defined as the non-productive time needed for the machine to switch from the last processed good of the previous batch to the first good of the new batch that has to be processed ▪ SMED main goal is to achieve the setup/changeover operation in less than ten minutes. Even if this goal cannot be achieved, reduction is still a great improvement. ▪ The setup time reduction reduces the overproduction and the work in progress. ▪ There are two types of Setups: - The external setup where the setup is done during the machine run time - The internal setup where the setup is done when the machine is off only	Ulutas (2011); Kumar et al. (2012); Mali et al. (2012); Begam et al. (2013); Abraham et al. (2012); Karam et al. (2018); Coimbra et al. (2009); Gest et al. (1995); Cherrafi et al. (2016)
Poka Yoke	▪ Poka Yoke is a Japanese word that means “mistake-proofing”. It is a simple and inexpensive technique that prevents defective good in process from being passed to the next process. ▪ The main concept of this approach is to detect, eliminate, and correct errors at their current source before reaching the customer. ▪ Poka Yoke prevents abnormalities and defects by eliminating mistakes.	Deshmukh et al. (2010); Plonka (1997); Saurin et al. (2012)
Ucell	▪ Workstations are moved close to each other to minimize transport between them. ▪ More workstations or cells can be put in the same hall, which reduces the transportation time between workstations.	Bhasin (2015); Prakash et al. (2017); Chong et al. (2013)
Cross training	▪ Cross training aims to achieve multi-skilling for workers. ▪ It is essential and vital that operators become multi-skilled. This will increase work variety and thus, decrease work boredom. ▪ Cross training creates a balanced workload and a sense of responsibility between cross-trained operators.	Diego Fernando et al. (2007); Bamber et al. (2000); McDonald et al. (2009)

In Table II.1, we first displayed the various Lean tools selected for this dissertation and they are most commonly used in industry and literature. To bring the argumentation of that selection, we can say that since the objective is to simulate the contribution and the adaptability of Lean tools in different situations, it is necessary to choose the tools that can be modelled and simulated. These selected tools can be modelled and applied on the production line, time, quality, and flow. In order to run such simulations, one should have real existing parameters.

Some Lean tools like Kaizen or Gemba are related to the human aspect (identified as being soft Lean manufacturing tools in Chapter I, see section I.6). VSM, as another example, is a graphical illustration tool that highlights the information of the production flow and facilitates the communication with the top management. These tools are necessary and constitute an important part of Lean thinking and have a significant role in the Lean project management success.

However, it is difficult to formalize, model, and establish a configurable simulation for these Lean tools.

In Figure II.1, we list the choice of Lean tools through a clear separation between the tools considered as configurable and virtually reproducible, and the tools that are human related and cannot be simulated and reproduced on a virtual production line. In this thesis, we are working on the operational level that does not fall under neither the strategic nor the tactical level. The operational level demonstrates the interest of Lean tools that can be technically configured and simulated in production manufacturing systems.

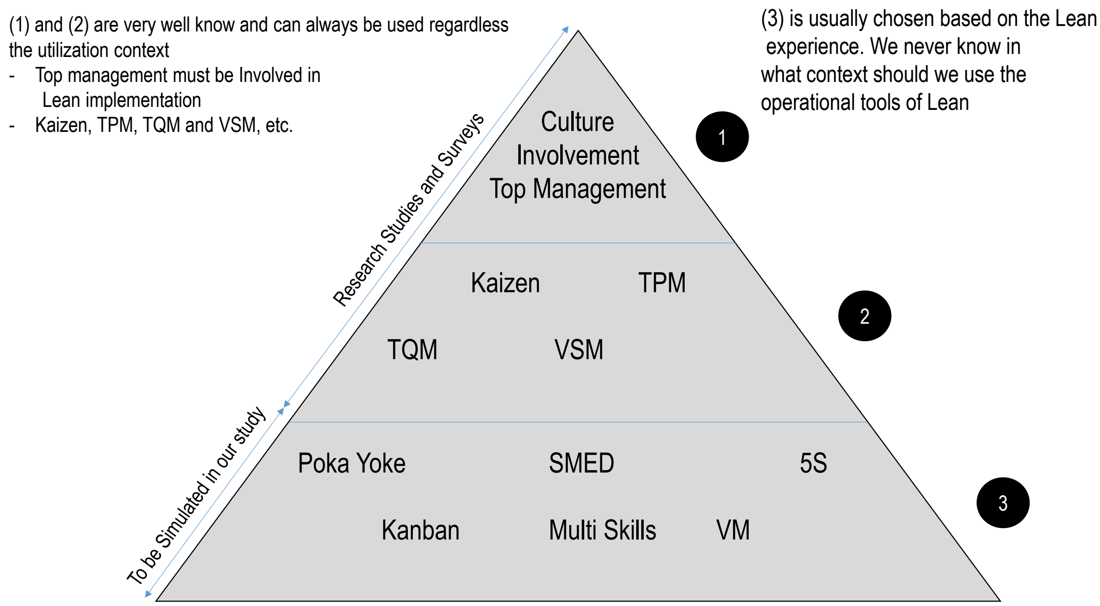


Figure II.1 Lean Tools to be simulated vs Lean Tools of human influence (Cannot be simulated in DES)

When the strategic level is mentioned in research, authors mainly talk about top management involvement (Jeyaraman et al. 2010; Kaye et al. 1999; Taner 2013; Lande et al. 2016; Antony et al. 2012; Alaskari et al.; Antony et al. 2002) and the importance of leadership (Achanga et al. 2006; Kaye et al. 1999; Kundu et al. 2012; Laureani et al. 2016; Antony et al. 2002). These two elements, among others, are particularly related to managers and group strategies. The encompassing tactical level (Kaizen, TQM, TPM and VSM) is also placed in the same box. These are imperative and required Lean tools; however, they cannot be developed and implemented in simulation. Strategic and tactical Lean elements can be considered essential in the management and completion of a Lean project but, from an engineering point of view, it is difficult to manipulate these concepts through technical representations and simulated modules.

We assume that the aforementioned Lean tools (strategic and tactical) assimilated to soft Lean manufacturing tools (discussed in Chapter I) are acquired and estimated as a prerequisite. We also support their importance and early use prior to operational elements. This point is supported by the following references.

Drohomeretski et al. (2014) claim that one of the starting tools for applying Lean is VSM. Singh et al. (2010) showed that with early application of VSM, it was possible to identify the various disruptions in the current systems and reveal the points of improvement; an interesting Work In Progress (WIP) decrease by nearly 89% and a reduction of 12.62% in the processing time were observed. Marodin et al. (2013) remind that VSM *"seems to be a fairly generalizable element of Lean production implementation. Besides being an implementation method by itself, the use of VSM is cited as a step for Lean production implementation by four other methods".*

In the paper of Chiarini (2011), the authors argued the necessity to carefully establish strategic Lean implementation with a combination of quality care ISO 9001 that specifies the requirements for QMS (Quality Management System) chosen in the strategic level. This analysis carried out a nine-years' study within European companies (using Lean and ISO 9001 certification). It shows that the company is able to increase its efficiency and standardize its Lean practices at early strategic level: total productive maintenance (TPM) and kaizen events were primary implemented.

To conduct a successful Lean approach, we cannot disregard the strategic and tactical elements. The steps are not "exclusively" an assembly of tools. However, it is certain that Lean includes essential operational tools to improve the production flow. Pepper et al. (2010) report that even if the VSM is considered as an important tool for identifying added and non-added value activities, the Lean is also based on a much larger set of tools (e.g. SMED, 5S, and others) that we should not ignore in fear of being qualified as using a Lean toolbox.

Nevertheless, even having guaranteed the strategic and tactical elements as mathematical inputs for our next development of hypotheses, the difficulty remains in making the appropriate choice of tools according to its environment. Trying to separate the operational elements will help provide greater visibility for decision-makers regarding the choice of technical operational tools that can be deployed in the production lines. Moreover, these tools can be modelled, reproduced, and simulated on a virtual environment.

II.2 Identification of industrials contexts

"Industrial context" is a very wide concept difficult to grasp. Perera et al. (2013) stated that developing a common definition that completely satisfies the demands of all domains equally, is impossible. There exists a variety of definitions for "context". The definition proposed by Dey (2001) and stated by Perera et al. (2013) and Alegre et al. (2016) describes context as being "any information that can be used to characterize the situation of an entity". An entity can be a person, place, or object. Bazire et al. (2005) define context as "a set of circumstances that frames an event or an object". Dey (2001) defines it as "any information that characterizes a situation related to the interaction between users and the surrounding environment".

Many classes of context can be included. Moreover, it is expected to perceive some research works that evoke only one industrial context (Mezgebe et al. 2018) where authors evoking the disturbed industrial context refer only to the variability of manufacturing process.

In this section, we will go through the concepts of the industrial context before defining the scope of the study and analyzing its elements.

In the paper of Rosenberger et al. (2018), the authors found that the context's definition of Dey (2001) is the best adequate one for the industrial domain after comparing different definitions. Nevertheless, they suggest two minor changes to improve its applicability. The notion of situation is first extended, "not only the situation of an entity but also the entity itself is included"; its characteristics are important for context awareness. Example of globally deployed information procurement system, operating in different countries with different languages, is provided. An information context would not include adapting user interface to the user's native language speaking if it was only externally considered. Therefore, the workers' language skills can be considered as "context" even though this element characterizes not only the situation of an entity but also the entity itself. Second, distinction is made between material objects (machine) and immaterial states (failure). For industrial applications, the immaterial states may increasingly become more important to consider. We subscribe to the following definition given by Rosenberger et al. (2018), "*Context is any information that can be used to characterize an entity, its condition, or its surrounding situation, if the information is considered relevant to the interaction"*.

The summary of the findings and subtle differences are shown in the Table II.2.

Table II.2 Industrial contexts as found in literature

Industrial Context	Content	Authors
○ *Primary context* ○ *Secondary context*	• It allows full capture of context around the entity • Primary context (time, identity, location, and activity) • Secondary (other events)	(Dey 2001)
○ *Set of circumstances*	• It considers event or object and surrounding circumstances	(Bazire et al. 2005)
○ *Active group* ○ *Passive group*	• Active group gathers all need context to identify the entity and its condition • Passive group gathers the remaining other contexts	(Chen et al. 2000)
○ *User (entity)* ○ *Environment* ○ *System* ○ *Information retrieval* ○ *Pattern recognition*	• Related to entity • Related to environment • Related to exploitation of system • Related to workers information • Related to past data	(Rosenberger et al. 2018)

II.2.1 Retained "Industrial Context" elements for the research

To define the industrial contexts of complex situations and implement production systems that can react to the contexts, decision makers have the possibility to build their systems upon the stated

contexts. Context classes don't have an equal importance in industrial applications. The company, measured as an "entity" itself, choose whether a context is strongly considered or not, based on its needs. Therefore, it is important not to ignore any context among the defined set of contexts as they might play a crucial role to figure out the relevant Lean tools accordingly.
In the current contribution, we consider the different specificities of the authors to build up our own industrial contexts related to manufacturing systems.

The entity and conditions evoked by Chen et al. (2000) are considered in our hypothesis. The manufacturing system driven by managerial decisions is undergoing different *market conditions* (uncertainty of the market and fluctuation of the order book). In the scope of our research, the entity represents all manufacturing systems, from the raw materials going through the assembly line until the final products. The entity itself, as claimed by Rosenberger et al. (2018), is also the source of certain context. The entity has its specific typology of production and has its possible circumstances (Bazire et al. 2005) of machine failures or workers' disturbances. To summarize, four industrial contexts are considered for the present dissertation.

- Industrial Context 1 noted *<ctx.1> Market fluctuation.*
- Industrial Context 2 noted *<ctx.2> Demand diversification.*
- Industrial Context 3 noted *<ctx.3> Uncertainty of Resources*.
- Industrial Context 4 noted *<ctx.4> Typology of production.*

Table II.3 Industrial contexts used in the thesis

Industrial contexts	Content	Linked to context classes of
<Ctx.1> Market Fluctuation	Market fluctuation is the possibility for the demand of certain product references to increase or decrease suddenly in an unexpected way. This context shows the possibility of changing the amplitude of demand.	(Dey 2001; Bazire et al. 2005; Chen et al. 2000; Rosenberger et al. 2018)
<Ctx.2> Demand Diversification	Demand Diversification is the necessity in some industrial contexts to widen the range of product portfolio. It represents the multiple product references to deal with.	(Dey 2001; Bazire et al. 2005; Chen et al. 2000; Rosenberger et al. 2018)
<Ctx.3> Uncertainty of resources	Uncertainty of resources is the symbol of a company that is often confronted to disruptions due to machines or workers. The non-reliability makes the manufacturing systems undergoing troubles	(Dey 2001; Bazire et al. 2005; Chen et al. 2000; Rosenberger et al. 2018)
<Ctx.4> Typology of production	Typology of production is a context where the company has to change the organization according to whether MTS or MTO strategy is adopted.	(Dey 2001; Bazire et al. 2005; Chen et al. 2000; Rosenberger et al. 2018)

Two of the context classes are external (Market fluctuation and Product diversification) and the other two context classes are internal (uncertainty of resources and typology).

- The four identified classes belong to the primary context (Dey 2001): because they are linked to *time notion* (market fluctuation is varying over the planning time horizon). The typology of production defines the *location notion*. Demand diversification define *identity notion.* The uncertainty of resources defines the *activity notion.*
- The four identified classes belong to the active group (Chen et al. 2000): because all of the four elements are required to define the entity and its condition according to the author's vision. All four defined contexts are required.
- The four identified classes entail material and immaterial states of context as claimed necessary by Rosenberger et al. (2018).

According to the literature review, we built the industrial contexts using at least one context class from the authors in Table II.3. Moreover, the industrial reality induces the decision makers to be careful and ready to face any context. According to our analysis, those four axes seem to be the most relevant with the industrial situation.

In the following sections, each of the retained industrial context will be modelled and contribute to the building of the combined approach of Lean tools relevancy.

II.2.2 Research scope

The purpose of this section is to present the scope of our work. It is necessary to address some important points in order to have a good understanding of the industrial system studied in this dissertation.

The scope of our study is as follows:

A decision-making center belongs to a part of the supply chain. A focal company (OEM: Original Equipment Manufacturer) is considered. We examine the case of a decision-maker facing a flow management within his internal units. The supply chain remains outside the scope of the study for two main reasons:

- The first reason is delimited by the objective of the thesis: Observe with a simulation model the behavior of a production system and its reaction with regard to the choice of Lean tools when confronted with various industrial contexts and aligned with different objectives.
- Potential analytical bias: The internal focus is substantially controllable at the scale of mathematical and simulation modelling. The objective of this work is to build a first approach supporting the hypothesis of the adaptability of Lean tools to industrial contexts and to decision makers' objectives within their production units. The contexts are multiple as well as the objectives. A scientific carefulness has led us to reduce the number of stimulated partners in the network in order to benefit from a visibility and an achievable potential of analysis. Otherwise, the combinatory of parameters and the crossing of

situations could lead to inextricable situations reporting NP-complete problems in mathematics.

A company manufacturing and selling valid products. No engineering/design phase are required. We assume that the products have been defined by the design service and that the range of production is pre-existing, and the components' composition is known (identified bill of materials). The method department has also defined the phase of industrialization and the means of production put at the service of the enterprise. Suppliers of product components are also identified and are reliable in their deliveries. No disruptions will be considered from the supplier's part for the scope of study.

A company processes under convergent production flow. The process of assembling a product starts from the receiving of raw materials to the production of the sub-parts and the assembly of the finished product. This research study focuses on manufacturing companies following the assembly in their production process, as it is often the case in the automotive and aeronautical industry (scope of the case study discussed in Chapter III and IV).

We exclude from our initial assumptions companies experiencing divergent flows (i.e. wood industry). In this context, companies have to build a disassembly process beforehand with constant hazards in raw materials availability. This configuration being very particular and scarce, we are renouncing this production scheme in order to concentrate our modelling and simulation efforts on an accepted and frequent configuration, which is the assembly industry in companies.

II.3 Industrial Objectives Identification

In the previous section, we defined the industrial contexts based on theoretical research. In this section, we will go through the "industrial objectives".

In manufacturing companies facing increased competition, decision-makers define a strategy for positioning their products in the market. They start from the choice of industrial systems supporting their production process to the market penetration strategy. Such decisions are based on an economic plan and a market penetration model, being the foundations of a strategic approach. Decisions falling in the strategic sphere cannot be modeled nor configured within the framework of this thesis, since they are subject to the subjectivity of the decision-makers. However, within an internal framework of the strategy, all decision-makers must be able to define the key objectives that their company is trying to achieve.

We are considering starting our study on a validated product belonging to a manufacturing company that has already established an economic model for its market and whose production system is defined beforehand.

Therefore, we are disregarding strategic and tactical decision-making stages that cannot be modeled or acted upon, especially that this study is scientific and based on the simulation of the production flow.

The specific question is: What are the potential objectives that a company aims to achieve when operating in a competitive environment?

From our point of view, an industrial objective is a kind of target for a company in an uncertain environment. Obviously, a company that has defined its targets aspires to achieve them as best as possible. The industrial objectives we are looking for, must remain within the scope of the operational level, which concerns the flows' management and the production's control from the purchase of raw materials to the delivery of finished products.

In order to achieve a consensual form of industrial objectives, we refer to the different approaches reviewed in the literature. The major finding we deduced is the challenge in defining a consensual set of determinant factors in a company's strategy.

The articles consulted fall within the "Operations Management", "Supply chain management" and "Production efficiency" domains. In the various articles consulted, after a careful study of what the objectives could represent as we imagine them, we found references that evoke the objectives in a different way. In the work of (Slack 1991), the author considers what he calls competitive priorities. A similarity of concepts can be seen with our desire to achieve industrial objectives. The "competitive" notion refers somewhat to the environmental factor to be considered, as the competition is imposed on the manufacturing company. The "priority" notion is an equivalent strategic parameter from a decision-maker point of view, and with the provided arguments, it ended up being considered an industrial objective.

In the list suggested by (Slack 1991), the quality is highlighted – offering products that meet project specifications. The reliability is the respect of delivery deadlines. The flexibility reflects the capacity to adapt operations whenever necessary and respond quickly whether it is due to changes in demand or needs of the production process. The speed – striving to achieve a shorter interval of time than the competitor since the start of the production. The cost – offering products at a lower cost compared to the competitors. Innovation – designing new products and launching more diverse products in faster development times than competitors.

Drohomeretski et al. (2014) highlight the notion of performance dimensions. This theory is also shared by the following authors (Okoshi et al. 2019). The various elements stated are in line with the "competitive priorities". Henao et al. (2018) identify three macro-performance categories: social, environmental, and operational performances; where the operational performance is similar and referred by other authors as performance dimensions. Khanchanapong et al. (2014) also refers to the concept of "operational performance". Bortolini et al. (2018) discuss four "performance perspectives" namely "responsiveness, system complexity, reliability, and quality". Although the labeling of the concepts differs between the authors, however, it is important to denote the common similarities.

In the Table II.4, we summarized the elements defining performance dimensions, which we call within the framework of this thesis "industrial objectives" anticipated and monitored by manufacturing companies.

Table II.4 Industrial objectives regarding the Performance dimensions in literature

Industrial objectives	(Wheelwright 1978)	(Leong et al. 1990)	(Slack 1991)	(Garvin 1993)	(Longoni et al. 2014)
Main insights	Efficiency Reliability Quality Flexibility Speed Cost	Quality Delivery Cost Innovation	Quality Reliability Flexibility Speed Cost Innovation	Quality Reliability Flexibility Speed Cost Innovation	Cost Delivery Flexibility Quality
Industrial objectives	(Drohomeretski et al. 2014)	(Bortolini et al. 2018)	(Khanchanapong et al. 2014)	(Henao et al. 2018)	(Okoshi et al. 2019)
Main insights	Speed Quality Flexibility Reliability Cost Innovation	Responsiveness Quality Complexity Availability	Quality Delivery Flexibility Cost	Cycle time/ Lead time Quality Inventory Quantity /Flexibility	Flexibility Speed Cost Innovativeness Dependability

Referring to the Table II.4, many common denominators stand out. Almost all authors systematically considered quality, cost, time, flexibility, and reliability. Reliability and quality were the major performance indicators.

Although some authors such as Wheelwright (1978), Slack (1991), Garvin (1993), and Drohomeretski et al. (2014) refer to reliability and quality as two separate concepts, other authors who consider reliability as an OTD (On-time delivery) factor, combine reliability and quality in the same category titled “quality”. Besides, Bortolini et al. (2018) bring together the concept of “reliability” to “quality”. The notion of innovation and innovativeness are similar. They are linked to the company’s ability to offer new products and extend the existing range of product portfolio. The notion of “dependability” proposed by Okoshi et al. (2019) refers to the “responsiveness” given by Bortolini et al. (2018). The term dependability is not adapted to our vision. It refers to a pejorative dimension rather than a target. We prefer the objective of “responsiveness” or “Reactivity”.

Longoni et al. (2014) and Khanchanapong et al. (2014) use the word “delivery” while other authors refer to “speed”. We prefer to preserve the word “delivery”. “Speed” under the notion of acceleration may be poorly perceived especially in scientific work related to Lean. Delivery symbolizes the time taken to proceed with the delivery of a finished product including the partial processes that come along with it: cycle time, process time, lead-time, etc.

All the proposals for consideration of performance concern the operational level with the exception of the approach of Henao et al. (2018) which involve two more strategic dimensions with the societal (well-being at work, polyvalence) and environmental considerations (energy emissions, etc.). We specify that these considerations are outside the scope of the present thesis. Drohomeretski et al. (2014) combined a list of performance dimensions assimilated to the industrial objectives in a table. Similar dimensions were listed in the summary of Okoshi et al. (2019). Accordingly, we draw inspiration from these tables to propose our synthesis and our vision, valid for this thesis.

Table II.5 Summary of industrial objectives adapted from (Okoshi et al. 2019)

Authors	**Requirements/ Target**	**Industrial Objectives**
(Wheelwright 1978; Bernroider et al. 2014; Chen et al. 2013; Slack 1991; Drohomeretski et al. 2014; Okoshi et al. 2019; Longoni et al. 2014; Bortolini et al. 2018; Franco-Santos et al. 2007)	• Do not make mistakes • Products in conformity with design specifications. • Manufacturer offers capability to the production process	*Quality* ***< Obj.1>***
(Nudurupati et al. 2011; Yusuf et al. 2014; Okoshi et al. 2019; Hong et al. 2011; Prajogo et al. 2012; Chae et al. 2013; Longoni et al. 2014; Henao et al. 2018)	• Keep delivery promises, increase service level • Correctly estimating the delivery dates • Able to meet the clients' deadlines • Clearly communicating dates to the client • Lead time should be lower than the competitors • Lead time: the total amount of time between the placing of an order and the receiving of the goods ordered	*Reactivity* ***< Obj.2 >***
(Tahir et al. 2010; Garrett Jr et al. 2015; Malhotra et al. 2014; Longoni et al. 2014; Okoshi et al. 2019; Henao et al. 2018; Khanchanapong et al. 2014)	• Adapt or reconfigure the production system/production process • Able to attend the changing demands • Able to reconfigure the operations due to changes • Manufacture system is able to change in the right pace.	*Flexibility* ***<Obj.3>***
(Demirbag et al. 2010; Franco-Santos et al. 2007; Ghattas et al. 2014; Longoni et al. 2014; Okoshi et al. 2019; Henao et al. 2018)	• Manufacturing the products at low cost • Being more efficient than the competitors • Negotiation of low-cost resources • Efficiently running the production process	*Cost* ***< Obj.4 >***
(Tan et al. 2003; Zhao et al. 2011; Yusuf et al. 2014; Drohomeretski et al. 2014; Okoshi et al. 2019)	• Design new products • Launch a more diversified collection of products in reduced product developing times • Reduce the Time To Market (TTM) comparing to competitors	*Innovativeness* ***< Out of scope >***

Regarding the various industrial objectives at the operational level, we can agree with the authors' idea of keeping four objectives as a major target to address in this study. Quality, Reactivity, Flexibility, and Cost objectives are considered in our study. The notion of "innovativeness" is considered to be outside the scope of the study. Indeed, we recall the hypotheses set out above, we are dealing with a company whose product portfolio is already validated, known, and marketable.

We exclude the development/design phase of the product. For this reason, the following table maintains innovativeness as an objective for companies to meet in order to acquire new market shares. It is simply specified for the framing and positioning of our study that the innovativeness will not appear in the list of objectives chosen for our thesis. We discard this phase of design to avoid combinatorics. In addition, the design phase is inherently uncertain with unpredictable outputs in the situation of fictitious simulation. Since the objective is not to study design parameters but production parameters, we are intentionally directing this study to the life cycle/production phase.

The listing and identification of the industrial objectives is the basis for the construction of our conceptual model in the coming section. We specify that at this stage, no prioritization of the importance of one objective over another is assumed. An interesting study was published in 2014 with this regard. Drohomeretski et al. (2014) test various hypothesis related to the operational management model of the Brazilian companies and their relationship to performance dimensions. 95 responses (over 178 sent) were analyzed.

Based on their findings, the reliability and quality appeared to be the most significant (74 and 70%), followed by speed (59%). For the 95 companies studied: *speed, quality and reliability,* are the most important representing a competitive advantage (>58%) of total sample. The performance dimensions: *flexibility, innovation and cost* are making the difference for more than 41% of companies considered as parameter winners. These competitive priorities are based on the perception of what customers consider as most important.

Table II.6 Results of expected priorities assimilated to industrial objectives (Drohomeretski et al. 2014)

Competitive priorities and competitive advantage	Number	% Average
Speed	52	59
Quality	62	70
Flexibility	38	42
Reliability	65	74
Cost	37	42
Innovation	38	43

While this study is not generalizable since it is subject to statistical and sampling bias, it highlights the possible prioritization of objectives, as decision-makers may perceive them. Nevertheless, in our study, the objectives are considered equally important except for the innovativeness, which is outside the scope of our study.

The identification of the industrial objectives allows determining the key elements to follow in the construction of the conceptual model. When discussing industrial objectives, it seems intuitive to also identify performance indicators related to those objectives to be achieved.

II.3.1 Performance Evaluation based on selected Objectives

In the field of industrial engineering in general and in operations management in particular, the evaluation of performance can be a priori or a posteriori (Frein 1998). A priori evaluation consists of evaluating the performance of a system that does not yet exist (future system) or an existing system, but on which structural and/or organizational changes are desired. After-performance evaluation is performed on real systems over a period of time with real performance measurements. Within the framework of this thesis, and in order to reach a conclusion, we followed the simulation approach. Then, we can follow a priori evaluation performance for the company in order to help the decision-maker change and/or enhance the organization of the system (structurally or organizationally). We went through this model because the decisions to choose Lean tools are not yet applied and because the elements resulting from simulation will not certainly be the ones that we would actually get. Therefore, we will be able to test a range of decision controls to assist the decision-maker in detecting the impact of each Lean technique decision on the company's behavior and industrial objectives.

The industrial objectives described above refer to the concept of performance since they are by nature operational performance objectives. In the context of research requiring a conceptual model and a translation into a simulation model, it is absolutely useful to have performance indicators attached to each identified key objective.

The model starts with situation modelling; afterward it will present contextualized inputs and then define performance indicators to monitor the results. The key objectives set out in the previous section guide the choice of performance indicators. We subscribe to the vision adopted in the thesis of (Villemont 2004). The author recalls that when the problem of modeling a supply chain is approached from a performance evaluation perspective, the indicators are often: the customer service rate (percentage of orders satisfied), the average time required to satisfy an order and the average level of stocks according to Rota-Frantz et al. (2001). This interesting notion of stock measurement is also used in the work of Panwar et al. (2015). A potential area for further research is evoked through a new group of key performance indicators to build up in Lean environment. Examples are given as follows, level of waste, employees' suggestions, inventory levels and frequency of shortages or backorders. Inventory assessment and goods in progress levels seems useful and necessary.

A choice must be made on the selection of indicators in order to accurately reflect the industrial objectives intended by the decision-maker.

<Obj.1> Quality, a quality rate calculated by identifying the number of non-defected parts over the number of total parts delivered.

<Obj.2> Reactivity, the main aim of this objective is to always meet the clients' deadlines. It is very important to have fast reactivity even if disruptions or production changes occur. Thus, the minimum lead-time and the highest production throughput are required to reach this objective.

<Obj.3> Flexibility, refers to the ability of a production system to fast and successfully adapt to changing conditions. This objective covers the system's ability to produce new types of products.

<Obj.4> Cost, in the absence of simulation model with authentic costing considering both labor costs and machine costs, we evaluate the financial aspect through the costs generated by the immobilization of stocks both in finished products and in buffer stock (WIP). This simplification is justified by the exclusion of costs related to labor, machinery, maintenance, reworking, etc. Considering that this cost assumption is equivalent to previous studies and scenarios. In addition, the estimation of the variable part of the cost, which is the cost of the WIPs, is a legitimate simplification and does not hinder a bias on the results of costs that are relative.

II.4 Proposed methodology combining Industrial "Objectives" and "Contexts"

"Previous studies showed that using wrong practices, incorrect use of correct practices, or wrong order of use of practices lead to Lean failures and incur losses for the firms implementing Lean" (Abdulmalek et al. 2007). Dora et al. (2015) reinforce the importance of the sector's specificity and the context of the company. The authors highlight the necessity for firms to choose the appropriate Lean practices for implementation; Lean practices choice should be based on the planning stages and the firms' specific needs in order to achieve higher operational performances. Dora et al. (2015) noticed that specific food processing SMEs have difficulty in implementing JIT because of the uncertain demand fluctuation context. According to different researches, there is a sequence in which these Lean practices should be implemented. Shingo et al. (1989) have suggested that SMED and layout improvements should be implemented prior to the Kanban and flow improvements. Smalley (2004) has showed that stability improvements (manpower, machines, materials and methods) need to be the first techniques implemented in Lean.

In the research of Netland (2016), the authors evoked that caring about the impact of Lean in industries is widely studied in the literature. One should not worry about the "impact" of Lean but should be concerned with "how to properly introduce Lean". The issue raised with the "practice choice" is consistent with the recommendations of Netland (2016). Thus, the research question of primary interest is no longer whether Lean is beneficial to the industry or not, the question is how to successfully implement it (Netland et al. 2014; Liker 2004; Rother 2009). In many cases, companies suffer from Lean integration failures (Bortolotti et al. 2015; Camagu 2010; R. Jadhav et al. 2014). Decision-makers can use researches as a roadmap to help them clarifying the contribution of tools gradually in their contextualized configuration. Bortolotti et al. (2015) remind of the difficulty of implementing Lean. While Lean integration projects are becoming more frequent, the number of companies failing to achieve a better performance is gradually increasing. Academics in the field of OM discuss the different reasons behind these failures called Lean management complexity in Lander et al. (2007). In fact, companies are hardly working to successfully implementing Lean management in a complex system of inter-related sociotechnical practices.

II.4.1 Modeling: Representativeness and Experimentation

Models are intelligible, artificial and symbolic representations of the situations in which we intervene (Le Moigne 1990). The representativeness of reality by models (for a dynamic system case) is an interesting approach from which we can extract scientific knowledge representing the system's reality (Fishwick 1997). The purpose of building a model is to understand and study the system represented by this model. The model's reusability helps in analyzing and developing the system under study. The modelling approach covers the steps and the intelligible process leading to a representative model of the studied system.

We subscribe to the vision of dual function: representation and simulation set out in the work of Tremblay et al. (2003). The function of representation is an abstraction and the function of using the model for the construction of new knowledge is the experimentation expected via simulation. The function of a model representation is sometimes in itself a goal to explain the knowledge of the studied system. Nevertheless, in some situations, the construction of knowledge is not attainable with the representation only, making the representative model work becomes a necessity. Here comes the simulation part that allows reproducing the functioning and the behavior of the real system in an identified context and in relation to a previously described problem.

The construction of a model does not in any way remove its "simplifying" and "artificial" appearance and characteristics. The model can be explanatory or predictive (Balin 2007). Model requirements validation depends on whether the model is predictive or explanatory. Indeed, for explanatory models built for simulation purposes, the strict comparison between simulation results and actual/real data can be complicated; indeed, the multi-variable nature and the consideration of qualitative factors make the comparison difficult (Le Fur 1994; Ferber 1997). Rather, it is a question of seeking a "consistency" of the models tested with real situations, and not a quantitative "adequacy" (Balin 2007). The simulation model remains above all a model to help understanding a real existing scenario. In addition, simulation model can be considered as a tool for reflection allowing a good and better understanding of the system.

In Figure II.2, based on the research model established by Taggart (2009), we have proposed our research approach pursued in this manuscript. The first step establishes the theoretical foundation; literature positioning, the scope of research, and the problem statement. Next step is the research design selection. The methodology proposed in this research is based on a combination of two elements of studies introduced previously: the industrial contexts and the objectives of decision-makers. A case study was selected to conduct this research.

The implementation part involves the technical development of a co-simulation framework that will be used as decision support. Managers can use a developed platform to virtually analyze the Lean tools contribution on their manufacturing system.

An aeronautical case of a real company based in the north of France was chosen because it corresponds to our hypotheses stated in the scope of research. Furthermore, we have enough interesting data to develop the production system virtualization (nomenclature, processing time, operating range, product type, type of components, etc.). We will detail the data input and the development process later in Chapter III.

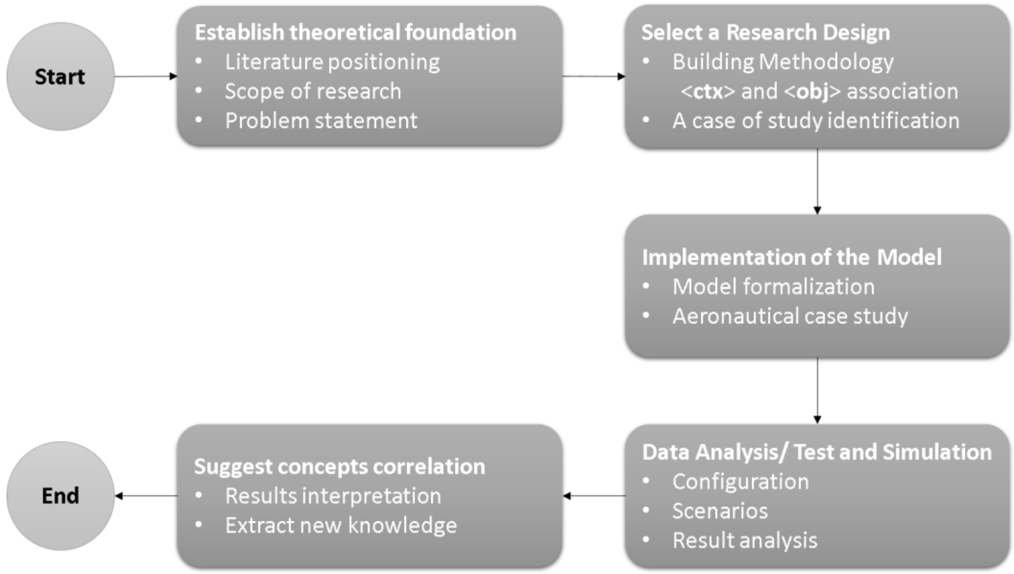


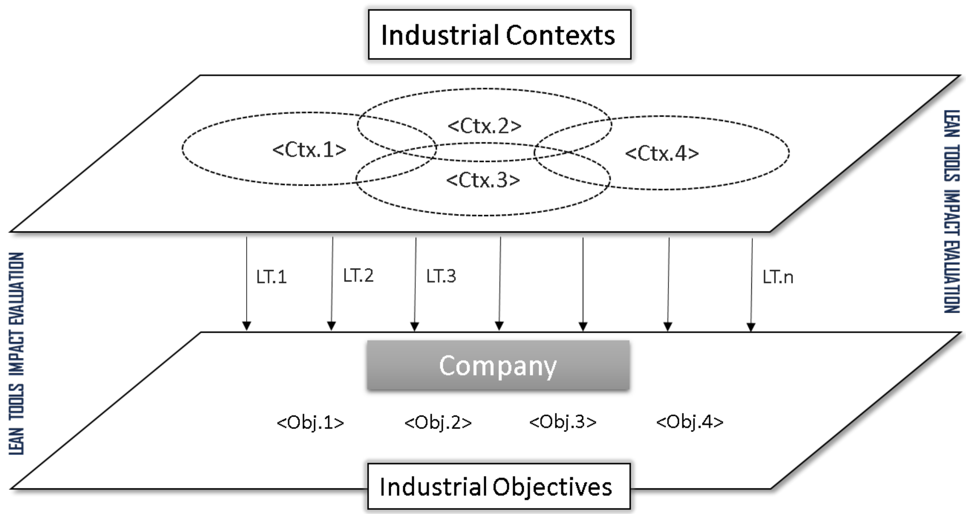
Figure II.2 Research approach in OM

II.4.2 Methodology: Conceptual Model and Formalization

The direction to include contextualized Lean tools implies the involvement of decision-makers in the decision-making system through simulation. In order to model and configure production systems targeting various goals, the consideration of the "industrial context" in a modelling and simulation approach is necessary. This is justified by the fact that the enterprise amends its economic framework according to market demand changes, demand diversification circumstances, uncertainty in human and technical resources, and typology of production. The company must have knowledge regarding the different contexts it intends to experiment, in line with the objectives predefined by the decision-makers. The objectives listed in the literature review turn out to be dissimilar in nature. The work converges on the four objectives studied above in order to organize the production system. The possibility of using various modular Lean operational techniques configurable to respond to different industrial contexts is intended. We propose a linkage-based modelling approach (see Figure II.3) between (i) analysis of industrial contexts (ii) alignment to key objectives. The approach developed represents a vision of interaction involving different parameters in order to gradually evaluate the performance of Lean tools responses to given configurations. As shown in Figure II.3, a surrounding level represents the upper layer including the various industrial contexts selected. The lower layer of Figure II.3 represents the physical system on which the different Lean techniques will be loaded then simulated to check their respective impact on the industrial objectives.

To summarize, leadership, Lean culture, and management involvement should be considered and effectively conducted in order to have a successful Lean conversion in the enterprise. When trying to integrate the Lean thinking in any sector (manufacturing, service, healthcare, etc.), the dominated idea in leading successful conversion is "the cultural mind-set" to put in place rather than "tool box" (Westphal et al. 1997; Vlachos et al. 2015; Dobryzowski 2016; Hopp et al. 2018). Moreover, leadership becomes a prevailing factor and consistent driving force that can shape an adequate problem-solving environment (Graban 2009; Longenecker et al. 2014; Womack et al. 1996; Delli et al. 2010; Longoni et al. 2013; Xie et al. 2019).

Overcoming doubt about the validity of these strategic aspects in the conduct of Lean (Gemba, Kaizen, leadership, culture, management involvement and operators) allows us to reach a more detailed analysis at the operational level. Disregarding these strategic and operational considerations will undoubtedly lead to the trap of the toolbox we are not conscious of. However, the individualized impact of each tool in a given study configuration consents a better understanding of the contribution and influence of the operational tool.



Ctx.1 - Market fluctuation	Obj.1 - Quality	LT_i| i=1.b
Ctx.2 - Diversification of Demand	Obj.2 - Reactivity	Set of Lean tools/Operational level to be tested
Ctx.3 - Uncertainty of resources	Obj.3 - Flexibility	
Ctx.4 - Typology of production	Obj.4 - Cost	LT strategic and LT Tactical: Positive initial conditions

Figure II.3 Combining industrial contexts and industrial objectives for Lean tool evaluation

The steps of a simulation-based decision model can be summarized as: conceptual modelling of the research approach, coding model, implementation, and experimentation (Robinson 2006). In this chapter, we will detail the foundations of conceptual modelling. The other points related to architecture, languages and coding will be discussed in the two remaining chapters, because special vigilance is necessary in order to take appropriate control measures, as well as validation and verification; so that the modelled system adequately represents the behavior of the proposed system.

The research method suggests combining in a single conceptual model: the industrial contexts and the objectives, in order to define a set of cross-situations worth testing scenario sets. As shown in Figure II.4, the conceptual model catalyzes four tubes (crossing situations). For each defined tube, an algorithm will be formalized to define the intervention steps in order to pass the data on to the production system that will be subjected to different Lean Tools (LT). The conceptual catalyst model is built around mathematical formulas and a simulation platform. The mathematical formulas of industrial data are intended to synthesize and represent the situations studied in a concise and precise manner. The simulation platform is a necessary basis for the development of hypotheses and the testing of various scenarios. This part will not be detailed in this chapter because it has been the subject of a detailed and complex study in order to imitate the behavior of a production system in real situations (see Chapter III).

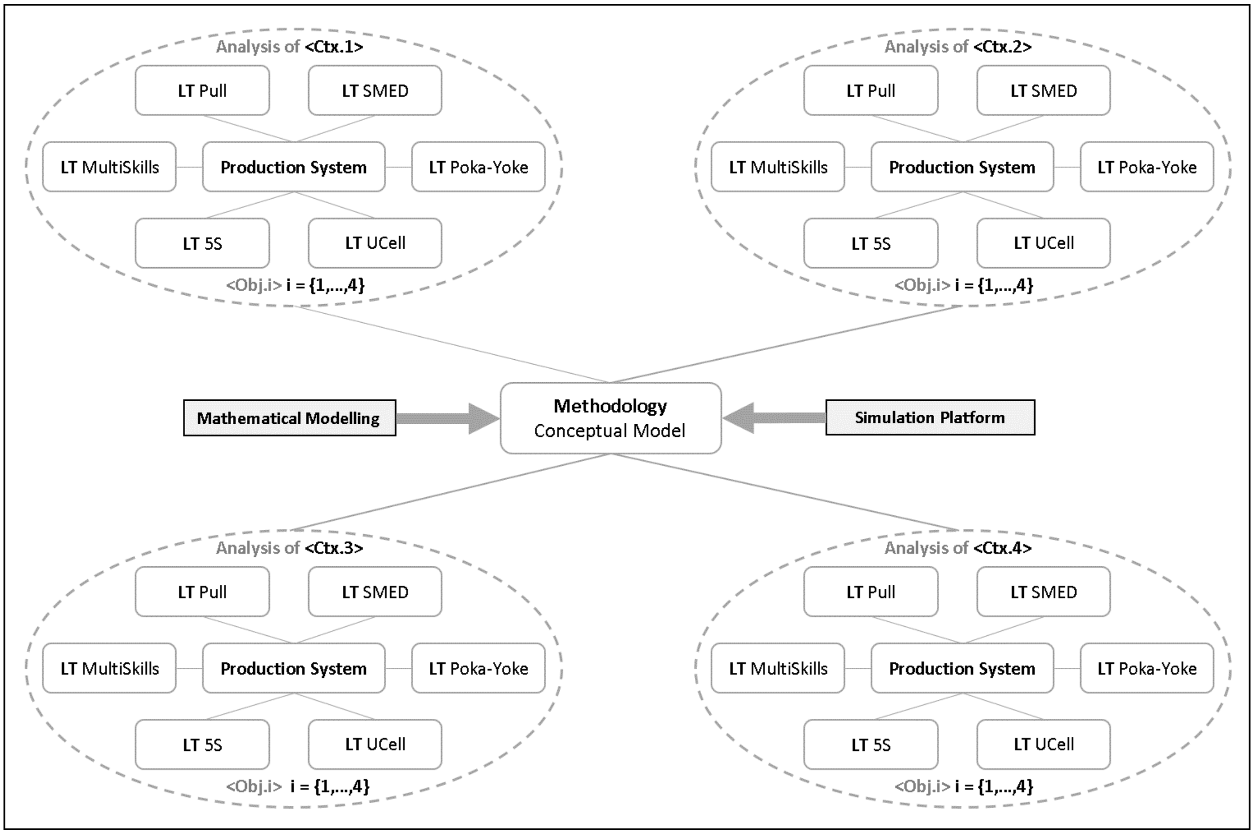


Figure II.4 Research methodology combining formal modelling and simulation model

Model of <Ctx.1>: Market fluctuation

Considering fluctuation and uncertainty in the market, this assumption represents the situation that can be faced by an enterprise having a change in its order book, upward or downward fluctuation. An upward fluctuation causes nervousness in the production systems and a need for company's adaptation in order to review its production schedule. A downward fluctuation causes a reorganization of the production system. However, at this stage, we cannot yet define the company's reactions to this context. The first step is therefore to define the representative variables in order to reduce the description's complexity of the targeted industrial situations of this study.

S_p is considered as the set of products produced by the company. Each reference is denoted by XRF_i where $i = 1, \dots, n$.

$S_p = \{XRF_i | \; i = 1 \dots n\}$

Moreover, as we are implementing the production system in a disturbed context, it is imperative to introduce the time (t), highlighting the period of time where the market fluctuation arises. The planning horizon over which demand is projected is time limited. This limit is noted (m).

(m)*: Last period of the planning horizon*

(t)*: Defined as the period of time over the global planning horizon.*

To obtain a visibility on the order book over the entire product portfolio, we will use the variable $D_{XRF_{it}}$.

$D_{XRF_{it}}$: *Demand of the product* XRF_i *at a period of time* (t)

When market is undergoing a fluctuation, we add the symbol “ ˆ ”.

$\widehat{D}_{XRF_{it}}$: *Demand fluctuation*

We represent by (α) the percentage of increase or decrease.

Fluctuation of the market at time t is relative to the initial value given in $D_{XRF_{it}}$ with an increase or decrease of ($\pm\, \alpha_{i(t)}$%).

The concise representation of the market fluctuation can be written as per the following:

$$\forall\, i = 1 \dots n\,; \exists\, t \in \{0 \dots m\}\,/$$

$$\widehat{D}_{XRF_{i(t)}} = D_{XRF_{i0}} \pm\, \alpha_{i(t)}\%$$

Model of <Ctx.2>: Diversification of Demand

In the context of demand’s diversification, diversity representation will be also formulated. In order to achieve this, it is necessary to define all customers that might order various products. Each customer can order different quantities of various references.

XRF_i*: Product Reference i*

Set of products: $S_p = \{XRF_i \mid i = 1 \dots n\}$

(m)*: Last period of the planning horizon*

(t)*: Period of time over the planning horizon*

(k)*: Maximum number of clients per company,*

C_u*: Client profile*

A set of clients is defined by: $S_C = \{C_u \mid u = 1 \dots k\}$

$D^u_{XRF_i}$ *: Demand of product* XRF_i *by client* (C_u)

$\overline{D_{C_u}}$*: Variety of demand required by client* (C_u)

f*: Number of varieties required* $(f \leq n)$.

The number of varieties required by the company will always remain less or equal to the total number of actual products ranges given by the company. We remind that in the hypothesis of the thesis no product is under development. We deal with the existing product references already validated and suggested for delivery to the various customers.

The concise wording of the diversification of demand may be as follows:

$$\forall\, u = 1 \dots k;\ \forall\, i = 1 \dots n;\ \exists\, t \in \{0, \dots, m\}\ /$$

$$\overline{D_{C_{u_t}}} = \sum_{i=1}^{f \leq n} D^{u}_{XRF_{it}}$$

Model of <Ctx.3>: Uncertainty of resources

In the context of unreliability of resources, the enterprise is faced with a situation where it will not meet its production target. In response to this situation, the company needs to adopt new strategies. The variables used to represent this context are defined in the following section. As we are working at the operational level, the resources concern the workstation machines of the production/assembly line responsible for the progress of the flow. These resources also include the operators in charge of operating the production/assembly line. The model must be able to represent the occurrence of a machine malfunction or an operator-related malfunction (absence, accident, etc.) at a given period (t). The set of machines is represented by S_M and the operators (human resource) by S_H.

Set of Machines: $S_M = \{M_p | p = 1 \dots U\}$

U*: Maximum number of machines*

Set of operators: $S_H = \{H_v | v = 1 \dots V\}$

V*: Maximum number of operators*

The machine malfunction represents an inability to run at a stable rate or to produce the expected quantity. An operator-related malfunction denotes a blockage at the workstation caused by the unavailability of the operator. The mathematical modelling does not go into the causes' details of the aforementioned malfunctions, but it will use a parameter to symbolize the technically configurable disturbance (the downtime machine or operator unavailability). Disturbance is symbolized with a binary representation:

- (1) to represent the existence of the production system's disturbance.
- (0) for the lack of it.

$\lambda_{vt};\ \lambda_{vt} \in \{0,1\}$*: Operator* H_v *disturbance event (error or absence)*

$\theta_{pt};\ \theta_{pt} \in \{0,1\}$*: Machine* M_p *disturbance event (failure, unavailability, or defect)*

Q_{ti}*: Quantity of products of type (i) produced at period (t)*

Q_{pti}*: Quantity of products of type (i) produced at period (t) by the machine (*M_p*)*

Q_{vti}*: Quantity of products of type (i) produced at period (t) by the operator (*H_v*)*

μ_{pti}*: Percentage of damage on the production system caused by machine (*M_p*) at the period of time (t) impacting product type (i)*

ω_{vti}*: Percentage of damage on the production system caused by human (*H_v*) at time period (t) impacting product type (i)*

To introduce the context of uncertainty of resources, a period of time (*t*) must show the possibility to undergo disturbances in two ways (exclusively and cumulatively). Both cases can co-exist where machine and human dysfunctions appear in the production system. It means that the quantity produced in a machine or by a worker may be reduced by a percentage of (μ) or (ω) respectively.

Therefore, the result of the following formula shows the produced quantities after a reduction due to human or/and machine malfunction.

$$\forall\, i = 1, \dots, n\,;\ \forall\, t \in \{0, \dots, m\};\ \ \lambda_{vt} \in \{0,1\} \wedge\ \ \theta_{pt} \in \{0,1\};\ p = 1 \dots U$$

$$Q_{ti} = \min_{t}\left[\sum_{p} Q_{pti}\left(1 - \theta_{pt} \cdot \mu_{pti}\right), \sum_{v} Q_{vti}\left(1 -\ \lambda_{vt} \cdot \omega_{vti}\right)\right]$$

The aforementioned formula takes into consideration all possible scenarios.

- The disturbance occurs on one or more machines without any human disturbance, therefore $\theta_{pt} = 1$. That means that the machine M_p incurred a disturbance at the period (*t*) and that affected the production rate by a percentage of μ_{pti}.
- The disturbance affects one or more operators without a machine disturbance, therefore $\lambda_{vt} = 1$, means that the disturbance affected the operator H_v at the period (*t*) and the production rate was by a percentage of ω_{vti}.
- Disturbance is caused by both operational resources; one or more machines and one or more operators. At this case, $\theta_{pt} = 1$ and $\lambda_{vt} = 1$.

The best-case scenario would be disturbance free. As this is not always the case in real life, one should focus on maintaining both μ_{pti} and ω_{vti} close to zero.

Model of <Ctx.4>: Typologies of production

The typology of production entails the physical organization of the company according to the type of products that the company is producing. As for the well-known SCOR model (Supply Chain Operation Reference model), four strategies assimilated to typology of production co-exist: Make to Stok (MTS), Assemble to Order (ATO), Make to Order (MTO), and Engineer to Order (ETO). In each production strategy, the decoupling point has different positions. Different manufacturing situations such as MTS, ATO, MTO, and ETO relate to different positions of the Customer Order Decoupling Point (CODP), also called order penetration point (Olhager 2010). The CODP is defined as the point in the value chain where the product is linked to a specific customer order. The CODP divides the forecast-driven (upstream of the CODP) material flow from the customer order-driven (downstream the CODP) flow. The CODP is the last point at which inventory is held. According to (Hoekstra et al. 1992), the CODP is important because it separates order-driven activities (pull) from forecast-driven activities (push). It coincides with the last major stock point in the goods flow. The CODP can also optimize the upstream activities independently from irregularities of the market demand. In addition, it identifies two main concerns for industrial decision makers: if the upstream is toward the CODP, the risk of stock build-up is significant; however, if the downstream is toward the CODP, the risk of missed orders becomes dominant.

The implied four situations describe the ability of manufacturing operations to operate in different typologies.

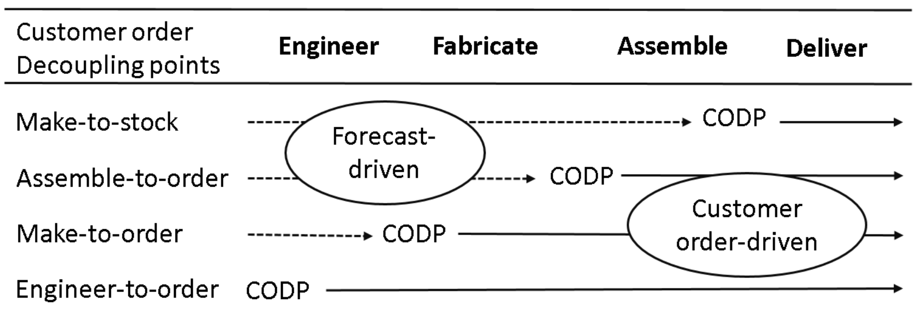


Figure II.5 Different Customer Order Decoupling Points (based on Sharman (1984))

The ETO is out of the scope of study. The typology ATO is the situation where semi-finished products are prepared before the CODP. At this time, the company will be waiting for customer orders to pursue the production systems. To simplify the modelling without reducing the validity of the analytical parameters, we assimilate ATO to MTO. In this situation, we are able to identify two main typologies: MTS and MTO. The company will organize its production based on these two contexts.

Context of an MTS typology

The configuration of the MTS typology is characterized by a set of points that lead the company to adopt it. Generally, the MTS is suitable for companies that are subject to strong competition and that produce non-technologically complex products easily imitated by competitors. It is also common to adopt the MTS when products have a long-term expiry date that allows safe storage, even if it will have additional storage costs.

(e): index of the steps along the production flow belongs to the Sourcing S, fabrication F, assembly A, and delivery D.

$e \in \{S, F, A, D\}$

XRF_i*: Product Reference i*

Set of Products: $S_p = \{XRF_i \mid i = 1, \dots, n\}$

(m)*: Last period of the planning horizon*

(t)*: Period of time over the planning horizon*

$D_{XRF_{ite}}$*: Demand of final product* XRF_i *known at time period* (t) *at step (e) of the flow.*

Fr_{ite}*: Set of the values attributed to demand* $D_{XRF_{ite}}$ *using the forecasting tools of the company at time period* (t) *at step (e) of the flow.*

$$\forall\ i = 1 \dots n\ , \forall\ t = 0 \dots m\ , \forall\ e \in \{S, F, A, D\}:$$

$$D_{XRF_{ite}} = Fr_{ite}$$

The company shall organize its production system and initiate the scheduling of manufacturing orders in accordance with the demand forecasts based on the market study.

Context of an MTO typology

The MTO is adopted in a particular context; when the company is not confronted with a strong competition. It is also utilized when the company is able to produce and deliver the products in a short period of time, noting the importance of the reliability of the production process to produce customer orders on time.

(e): index of the steps along the production flow belongs to the Sourcing S, fabrication F, assembly A, and delivery D.

$e \in \{S, F, A, D\}$

XRF_i*: Product Reference i*

Set of Products: $S_p = \{XRF_i \mid i = 1, \dots, n\}$

(m)*: Last period of the planning horizon*

(t)*: Period of time over the planning horizon*

In this context, the company has some of the demand forecasts in order to prepare the raw materials by an estimated assessment. The stage covered by the forecasts is the stage (e=S) Source. From this point, the company will base its production system on the firm orders it will obtain from customers.

In this context, the company started with requesting the raw materials based on the forecast study. This stage, sourcing stage S, is the only stage where the demand is based on the forecast. After this stage, all requests are strictly based on the order demand of the clients.

$D_{XRF_{ite}}$*: Demand of final product* XRF_i *known at time period* (t) *at step (e) of the flow.*

Or_{ite}*: Set of orders for the product* XRF_i *made by the clients at time period* (t) *at step* $e \in \{F, A, D\}$ *of the flow.*

Fr_{its}*: Set of the values attributed to demand* $D_{XRF_{ite}}$ *using the forecasting tools of the company at time period* (t) *at step* $e = S$ *of the flow.*

In the following formula, the CODP appeared due to the separation made between the forecast driven activities and the order-driven activities.

$$\forall \ i = 1 \dots n \ , \forall \ t = 0 \dots m \ ,$$

$$\text{for e} = \mathrm{S}: \mathrm{D}_{\mathrm{XRF}_{\mathrm{its}}} = \mathrm{Fr}_{\mathrm{its}}; \ \text{for e} \in \{\mathrm{F}, \mathrm{A}, \mathrm{D}\}: \mathrm{D}_{\mathrm{XRF}_{\mathrm{ite}}} = \mathrm{Or}_{\mathrm{ite}}$$

At this stage, the different contexts influencing the choices of the operational Lean tools were presented and modelled. These models are a representation of complex situations that must be considered and configured later on in the developed simulation platform. The conceptual model of Figure II.6 gradually represents the adapted methodology. After defining the objectives, we enter the inputs to the developed graphical system interface; this interface will allow users to run the co-simulation framework in order to test the LT efficiency implemented to their production system.

We assume that strategic and tactical Lean elements mentioned before do not affect our simulation process and are considered successfully met at simulation time (t=0). The focus of our study is on the operational level which is the source of Lean implementation ambiguity.

Multiple scenarios arise from the combination of different situations. At each stage, analysis is conducted in order to choose the most reliable tool(s) that can be adapted for each of the four studied contexts.

$LP_k = \{P_k \,|k = 1 \dots p\}$*: set of Lean tools tested*

$LP_a = \{P_a \,|a = 1 \dots q\}$*: set of adapted tools such as* $LP_a \subset LP_k$

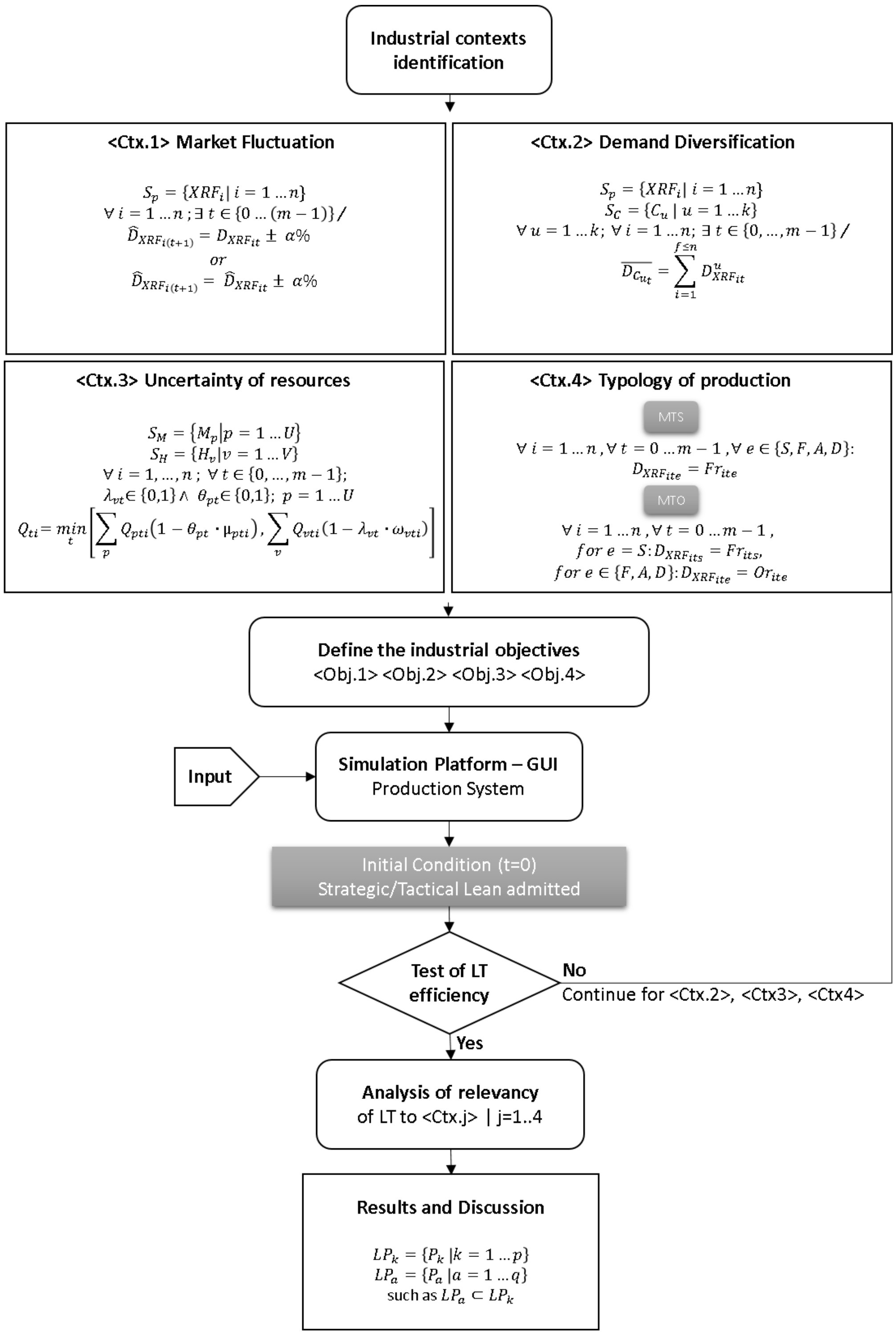


Figure II.6 Conceptual Model

Two key factors emerge. The first factor concerns the first objective that is quality and the second factor clarifies the specificity of innovativeness objective. Indeed, quality is a target objective for any business regardless of the economic context in which it operates. An enterprise that is faced with a market fluctuation and willing to meet the desired increase in quantity will not deviate from the quality requirements (Lyonnet 2010) if its initial strategy was to maintain a high level of quality. In the case of a company facing a diversification of demand with a multitude of customers requiring various volumes of various products, it can even be assumed that for the company wishing to work in small batches to hire orders on the market, the challenge will be to diversify references without losing quality (Bazire et al. 2005). When an enterprise is faced with an uncertain environment beyond the threat of a machine failure (decrease of the quality rate in case of technical failure) there is also the risk of human failures reducing the quality of the output products. The decision-makers in this case are competing for efforts to address these uncertainties and their impact on maintaining a satisfactory quality rate. Productions following the MTS model ensure stock availability in a highly competitive environment and in low-tech products (Olhager 2010). The quality of MTS products resulting from batch productions are randomly tested on one sample product. On the other hand, the quality of the MTO products and the specificity of the production triggered by the firm order are not based on forecast; rather the production requires a grounding of the production system. The production time and the quality standards must be absolutely respected during the manufacturing process (see the first dark grey box in the quality objective's row, Figure II.7)

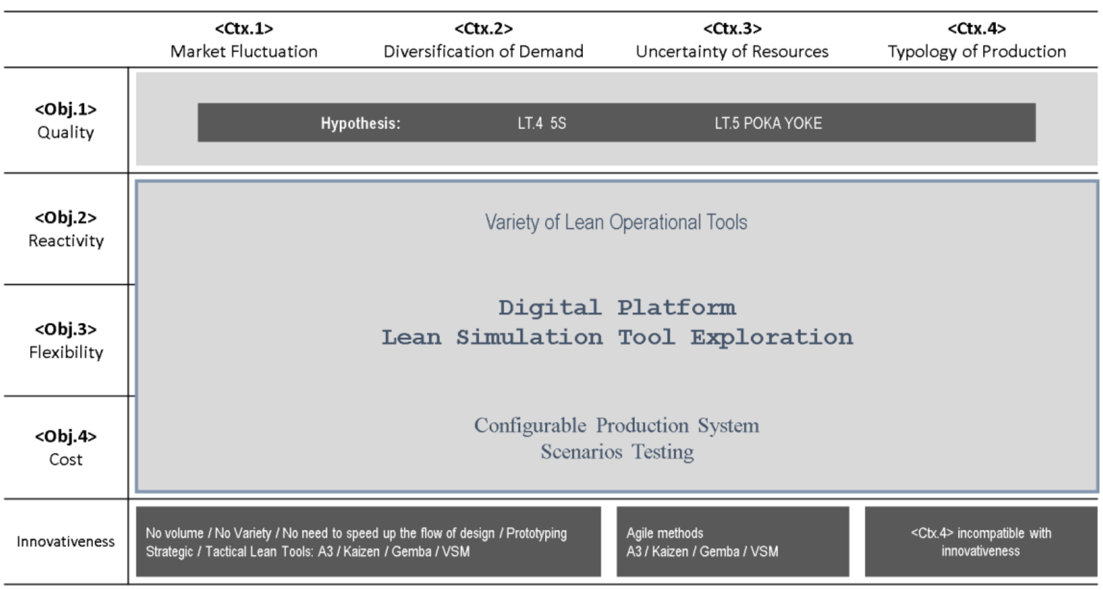


Figure II.7 Possible combination of analysis

The cross-sectional nature of quality in all industrial contexts leads to conclude that all operational Lean tools that can influence quality improvement could be appropriate for all contexts studied. Verification of this assumption will be carried out in Chapter IV where the results of the study are detailed.

In addition, referring to the literature, the "innovativeness" (last row in Figure II.7) objective pursued by some companies is not included in the conceptual model. In fact, the industrial contexts treated, especially "market fluctuation and diversification of demand", might lead to a variation in quantity and a diversity of references. They therefore do not intersect with the aspect of innovativeness (in which there is no fluctuation or variety when the product is in the prototyping/design stage). Since there is no notion of acceleration of innovation flows, operational Lean tools will not be studied, but the assumption of favorable initial conditions such as the use of Lean techniques (A3, Kaizen, Gemba, VSM, etc.) at a strategic and operational level by the actual teams is still valid.

In the context of uncertainty about technical and human resources, with an objective of innovativeness, the operational Lean tools do not contribute explicitly to the process of creating innovative value. Strategic and tactical tools (referred to soft tools in Chapter I) followed by the implementing team can be useful in routine and day-to-day management. The tools selected from a simulation perspective will not be tested under this objective, knowing that this notion is often managed by agile methods in the industry. For the "typology of production" context, this context naturally deals with the "production" phase in a product life cycle. There is an incompatibility between this context and the objective of innovativeness that by nature falls under the "conception/design" phase. The simulation of operational Lean tools will not be carried out in the cross "typology of production" and the "innovativeness" objective.

All remaining cross-configurations (in Figure II.7) can undergo the variety of operational Lean tools that can be simulated on the digital platform designed to allow a live exploration of the adaptation of Lean tools to the system of production. A reconfiguration will make it possible to write and expose the framework to different situations in order to decide and analyze the suitability of some Lean tools to given contexts rather than others. Chapter IV is dedicated to the results in order to identify the correlations found.

II.5 Research Questions identified: RQ.4

- **RQ.4 How to combine Objectives and Contexts to relevantly choose the Lean techniques?**

In this chapter, we studied the selected industrial contexts and objectives. The selected contexts for our research are as per the following,

- *<Ctx.1> Market fluctuation*, in this context, we consider the uncertainty and fluctuation in the order book of the company. It could be a market demand increase or decrease fluctuation. This fluctuation is represented by $\widehat{\mathrm{D}}_{\mathrm{XRF}_{\mathrm{i(t+1)}}}$.
- *<Ctx.2> Diversification of demand*, in this context, we consider the fact that customers might order various products. Each customer can order different quantities of various references. The diversification context is represented by $\overline{D_{C_u}}$.
- *<Ctx.3> Uncertainty of resources*, this context shows the produced quantities after a reduction due to human or/and machine malfunction. It is represented by Q_{ti}.

- *<Ctx.4> Typologies of production*, in this context both typologies are tested, the MTO and MTS typologies and are represented by $D_{XRF_{ite}}$ in the mathematical formulas.

All the aforementioned contexts are tested through the developed co-simulation framework that will run all chosen Lean tools simultaneously in parallel. The simulation results will allow us to check each Lean tool relevancy in each of the contexts stated above. Lean tools relevancy is checked based on four selected industrial objectives, *<Obj.1> Quality*, *<Obj.2> Reactivity*, *<Obj.3> Flexibility*, and *<Obj.4> Cost*.

The detailed configuration of the framework and the chosen simulation model are discussed in Chapter III. The results of Lean tools relevancy are examined in Chapter IV.

II.6 Conclusion

This chapter illustrates the various industrial contexts used in our study in relation to the literature. In one hand, the industrial objectives were described from an extensive literature review in order to define practitioners and researchers' priorities. We kept four objectives in tight link with the findings of the literature review. We defined a delimitation section of the research scope in order to outline the contours of our research work. On the other hand, this study raised the question that - companies wishing to introduce Lean tend to start implementing the Lean approach in a haphazard way, through experience, through foresight: Is this approach adequate? We conclude that there is a possible adaptation of the operational Lean techniques/tools to the context of the company. Therefore, the choice should not be made randomly but by contextualizing the correspondence of the tools with the defined contexts and the target objectives of the companies. This hypothesis was already validated in the research of (Lyonnet 2010) where the author was questioning the concept of offering new flexible and adaptable tools to Process and Manufacturing Engineering (PME) in the Rhones Alpes region; considering that an environment can influence the choice of tools.

For the moment, the correlation between the objectives remains diffuse and fuzzy. The aim of this thesis is to develop through the suggested conceptual model, a possibility of a simulation model enabling the identification of possible correlations and inter-influences.

The conceptual model discussed in the methodology section of this research study interferes with different industrial contexts, combined with decision-makers' objectives. At the intersection of these situations, the contributions of the various operational Lean tools can be studied. We proposed a modelling of the four different industrial contexts that strongly affect the demand variables, the order, and the production quantities. Once the models are realized, it will be easier to manipulate the action variables on the simulation model, which will be presented in the next chapter.

It is also necessary to resituate the Lean approaches throughout Lean thinking. It's obviously not a toolbox, hence, all the tools known by strategic and tactical decisions and resulting from the company's direction strategy are considered pre-requisites for competing efficiently.

In the inverse hypothesis of disregarding strategic Lean techniques, there always exists a doubt in non-validity of our tested tools at the operational level. To avoid the pitfalls, we assume that in initial condition (t=0) Lean techniques (VSM, KAIZEN, A3, Problem Solving, etc.) are considered used, adopted and encouraged by committed leadership. The hypothesis resolved in this study suggests that even in these surroundings conditions ultra-favorable to the deployment of Lean, there exist differences observed at the operational level. Certainly, a category representing a set of Lean tools will be more suitable for some contexts and not others.

Chapters III and IV will aim to provide answers to these questions. The simulation platform will allow us to test and explore the hypothesis based on a chosen aeronautical case study.

CHAPTER III. Co-Simulation Framework:
Case Study, Design, Architecture, and Development

Reproducing a physical process or a system is most often costly, time consuming, and disruptive. To avoid that, simulation serves as a good solution. In addition, simulation is employed in the course of analyzing systems' operations and behavior prior to the build process. Thus, landing the ability to engineers to optimize the system design, reduce errors, and decrease design mistakes. Several companies and organizations utilize simulation and virtual environments for entertainment and training purposes. Across different sectors, Simulation has become one of the best ways to try, explore, analyze and optimize systems structure, behavior and performance prior to the implementation process. Simulation is necessary to deal with real-world uncertainties, variations, and complexities (Ören et al. 1979; Detty et al. 2000).

Highlighting the different applications of simulation reveals the advantages of this process. The different applications can be divided into two main groups or categories (White et al. 2009). The first category is named "man-in-the-loop"; this category's purpose is mainly to train professionals; it can also serve for entertainment purposes. Man-in-the-loop simulations benefit several safety critical professions such as medicine, aviation and many others. These simulations help professionals in learning to operate in the real world through an exposure to a simulated workplace. The second group covers the examination and development of objects, tools and processes for analytical and performance optimization purposes. In this category, changes are applied to the entities and processes in order to search for the best model behavior or performance.

III.1 Simulation Core Concept

As per Figure III.1, the simulation development process starts by defining the problem and the system, formulating the conceptual model, designing the initial experiment, collecting and preparing the data, and by translating, verifying, and validating the model (Kelton et al. 2007). The next phases will be to run the experiments, analyze, and interpret the results, and finally, document the output results (White et al. 2009).

First, problems in the system must be identified. In this step, the end objective, performance measures, and period of the study shall be examined as well. Then, data collection and analysis should be performed. Input variables need to be identified along their probability distributions (Poisson, exponential, etc.). Afterwards, a model has to be developed using appropriate software tools. A conceptual model portraying the flow of entities is translated to a software compatible form. This model is verified through changing parameters while checking the corresponding output. After developing the model, a validation step is crucial to ensure completeness and consistency (Validation methods are detailed in the section below). It is also useful for improving the overall confidence in the developed model. The validation step consists of a comparison between real world outputs and simulation performance results under well-known settings. Subsequently, experimental conditions have to be established. In this stage, initial conditions are studied. Other factors are also examined such as determining whether the system is stationary or non-stationary, choosing the run length, the number of runs and many more. The final steps consist of running the simulation and analyzing the results accordingly. Future recommendations can be made upon a thorough interpretation of the simulation output and results.

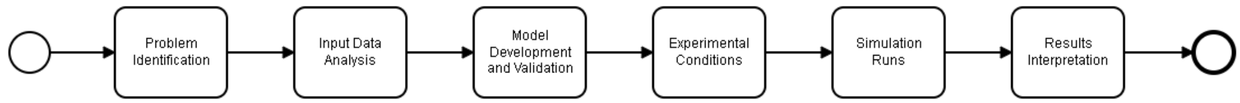


Figure III.1 Simulation Core Concept

III.1.1 Validation and verification of simulation models

Simulation models are more and more used to solve problems and facilitate decision-making. The designers of these models and the decision makers using the results obtained, are all interested in the validity of these results. The approaches of model verification and validation address this concern. Model verification is often defined as "a methodology to ensure that the model and its implementation are correct". Regarding the Validation of models, it is generally defined in the literature as "the proof that a model developed in its field of applicability, has a satisfactory range of accuracy consistent with the intended application of that model" (Foures 2015).

Validation and testing techniques are commonly encountered in the literature. A combination of these techniques is generally used by designers to verify and validate the submodels and the overall model. Some of the most used techniques include the following (Sargent 2010):

- *Animation:* the operational behavior of the model is displayed graphically over time. For example, the parts' movements in a factory during a simulation are represented graphically.
- *Comparison with other models*: Multiple results obtained by launching several simulations are compared with other models that are already valid.
- *Degenerated tests:* The evolution of the behavior of the model is tested by an appropriate selection of the values related to the input parameters and the configuration parameters. For instance, does the number of SimEntities in the server's queue continue to increase when the arrival time in the EntityGenerator is greater than the service rate?
- *Face validity:* Meetings with specialists in the field provide a return on the validity of the model. For example, is the logic of the conceptual model correct, and are the input-output relationships of the model reasonable for them?
- *Historical data validation:* If there is historical data (for example, data collected specifically on the system for test construction), some of the data is used to build the system and the remaining data is used to determine (test) if the model behaves almost in the same way for each test. This test is often performed by piloting the simulation model with distribution samples.

In our study, we used the Discrete Event Simulation (DES) to develop the actual model of a French aeronautical company and the Lean tools applied to this model. In order to exchange data and messages between all developed DESs, as well as sending/receiving data input/output during simulation runs from/to an external application, we used the HLA standard to develop the Distributed Simulation (DS) part.

This chapter is divided into two main parts, the first part is the models' development based on a DES system. The second part is the DS part, which focuses on data collaboration, time management, and synchronization between the developed models and an external developed Java application that provides a GUI (Graphical User Interface) to enter simulation-related data and draw graphs that illustrate the Lean tools' performances during the simulation run.

III.2 Discrete Event Simulation (DES)

A simulation system is considered discrete if during its observation, it is possible to identify periods of time during which the system does not change its state. These time periods are highly dependent on the details chosen for the study, and thus on the system's characteristics that were initially identified. In DES, the simulated system changes state or value at discrete points of time, and the simulation moves from one state to another upon an event occurrence (Fujimoto 1990). This technique is widely used by industries and research centers to design, validate, and optimize their organizations.

III.2.1 History and Evolution

In the past fifty years, the advancements in simulation software and the computing field have helped DES to become one of the most widespread modelling and simulation techniques. A brief overview of its history will be provided in order to understand its current and future stages. The history of DES can be divided into four main periods: The pioneering period, the period of innovation, the revolution period and the evolution period (Robinson 2005).

The pioneering period dates back to the late 1950's and continues to the 1960's. During this period, pioneers developed simulations with the help of the first-generation computers which appeared in the 1950's. These simulations were designed and developed using machine code. Moreover, the 1960's witnessed drastic improvements in the field of discrete event simulations. This is due to the development of programming languages and the increased reliability and power that computers could offer during that time. In addition, many simulation software were developed in the 1960's such as GPSS and SIMSCRIPT (Goldsman et al. 2010). The period of innovation starts in the 1970s. As its name implies, this period embodies persistent improvement and innovation. Simulation software continued to progress along the advancements in the computing field. Numerous new programming languages appeared (i.e. SLAM, GPSS-H, etc.)(Brunner et al. 1991). In addition, microcomputers were first introduced in the late 1970's. During that time, everything was being prepared for the 'revolution' to occur in the 1980's. During the revolution period, microcomputers became more commonly available in organizations due to IBM's introduction to the market. Furthermore, the period saw the entry of VIS software, with SEE-WHY being the first to be developed in 1979(Bell et al. 1987). Meanwhile, microcomputers and VIS packages continued to grow until their boom the late 1980's. Powerful microcomputers became accessible by most organizations and many VIS packages appeared, for instance, HOCUS, SIMAN/CINEMA and GENETIK (Bell et al. 1987). Several organizations in the manufacturing sector started adopting DES as a decision-aiding tool. The last period represents the evolution of DES from the early 1990's to the present. In the early 1990's, the world witnessed the success of the Personal Computer (PC), the World Wide Web and the Windows operating system. These technologies helped the field of DES and enabled models to be executed at high-speed rates (Hollocks 2006). The evolution of DES touched many areas such as software integration, visual interactive modelling and simulation optimization. A tremendous advancement was made through simulation's integration in the World Wide Web and distributed computing. This advancement resulted in the birth of DS (Robinson 2005).

III.2.2 DES in production and manufacturing domains

The simulation in manufacturing and supply chain fields became a very widespread scientific approach since early 2000's (Jain et al. 2002) because of the ability to reproduce a virtual system that simulates the real production system (Long 2014). In addition to a "What If" analysis of different scenarios that observes and understands the Supply Operations (Chatfield et al. 2006; Zhao et al. 2000) and forecasts the impact of alternative configurations (Tan et al. 2011).

The DES, in particular, is one of the preferred research topics nowadays (Yoo et al. 2010) for its ability to simulate production system and supply chain behaviors (Zengin 2011; Zengin et al. 2013). DES was often considered as a dynamic tool that allows the visualization and quantification of technological and operational changes in processes (Julie Yazici 2005). DES is suitable for leading analysis of the dynamics of discrete processes such as manufacturing systems (Ingemansson et al. 2004) and the possibility to run different scenarios in a short period of time (Banks 1998). DES is an effective tool for process improvement (Barnes et al. 1998). It is a method to simulate real system or process and it is nowadays used in different environments such as manufacturing plants, queuing systems, distribution systems, inventory and delivery systems, health-care, transportation networks, communication networks, and many others (Fishman 2013).

Jeon et al. (2016) remind that DES for Production Planning and Control problems is a frequently used tool that represents more than 45% of the simulation models in the studied sample. Further studies have made the attempt of combined methods as DES and Agent technology for studying complex supply network (Alavi-Moghaddam et al. 2012) to be able to integrate micro-behaviors of individuals and macro system to guide the managers in their decision-making process. In a complex production environment with a complex demand evolution, many authors use DES to quantify the effect of VSM implementation on Lean performance measures (Abdulmalek et al. 2007; Detty et al. 2000).

In some cases, DES alone is not an effective solution. The simulation system must be disassembled into subsystems or nodes in order to be parallelized or distributed on a multiprocessing environment for performance enhancements (Misra 1986). In other cases, a collection of interacting simulations is needed to form a more complex system that offers additional functionalities to the existing ones (Falcone et al. 2018). There are also scenarios where users need to compare many different DESs, and this cannot be run sequentially and needs to be also parallelized or distributed on a network of processors (Possik et al. 2018). For all the aforementioned scenarios, time management and synchronization mechanisms are necessary to avoid timing discrepancies and to ensure precise event interconnections and data communication between subsystems or simulations.

III.3 Distributed Simulation (DS)

Discrete event simulations were mainly limited by the power of the machine they were executed on. The emergence of complex models has resulted in the integration of distributed technologies to the simulation field. In fact, DS is a simulation where its execution occurs on multiple processes connected through a network. These different DSs are part of a comprehensive simulation that can be seen as one simulator (Chaudron 2012).

III.3.1 History and Evolution

In the 1983, SIMNET project, supported by the DARPA (Defense Advanced Research Projects Agency), has started. SIMNET was the first DS System for Virtual Reality applications and simulations. SIMNET project was achieved by BBN (Bolt, Beranek, and Newman), that was in charge of the data exchange and the DS system, and by Perceptronics that was in charge of the training studies. The main aim of SIMNET was to help military units to be organized and fight in teams. This system was used by the US military for military trainings, because training with real equipment was very dangerous and expensive. The key idea was the communication of multiple simulators over the network, where each simulator was autonomous, having its own display, controls, and resources. The interactions and messages exchange among the simulators is on P2P (Peer to Peer) basis without a central system.

In 1993, this project gave birth to the DIS (Distributed Interactive Simulation) protocol. The main aim behind the development of the DIS protocol was to improve and extend SIMNET's functionalities. The IEEE standard of DIS is still available nowadays under the name IEEE 1278 (Miller et al. 1995). In fact, the DIS standard is a communication protocol. It consists of sending and receiving messages called PDUs between different simulation objects. PDUs make the interaction between entities possible during simulation. The DIS protocol outlines 27 PDUs that are responsible of data exchange between simulation objects (Chaudron 2012). DIS succeeded but has different lacks of reuse and extensibility. Therefore, DARPA has planned to develop a new architecture called HLA.

When HLA was first developed, the standard HLA US DoD (Department of Defense) 1.3 was created. In the year 2000, it was adopted by IEEE and named HLA IEEE 1516. Then, it was modified and updated in 2010 to encompass improvements; this last version is known as HLA Evolved. The HLA protocol is a standard that helps in the development of DSs. HLA operates through the creation of a simulation that is composed of different simulation components. These components are called "*federates*". A federation consists of federates, a run-time infrastructure (RTI), and a Federation Object Model (FOM). The HLA standard defines ten rules to ensure a successful HLA simulation. The first five rules encompass the functionality of federations while the last five consist of the functionality of the federates. The aforementioned points and their mechanisms will be discussed later in this chapter.

Functional Mock-up Interface (FMI) is a standard that supports model exchange and co-simulation of dynamic models based on Extensible Markup Language (XML) files and compiled C code. This standard was developed during a project named MODELISAR. It is now developed and managed as a Modelica Association Project (Sievert 2016). FMI 1.0, published in 2010, is the first version of the Functional Mock-up Interface. This version was followed by FMI 2.0 issued in 2014. FMI was developed to improve the simulation models' exchange between the suppliers and the original equipment manufacturers. It is now supported by more than hundred simulation tools used mostly used in automotive industries (Neema et al. 2014).

III.3.2 Why HLA?

To exchange data between models and in order to develop the co-simulation process between all running simulation models, we used the HLA standard in our study due to some key limitations in the FMI standard. FMI does not have the time management and synchronization features that already exist in HLA standard (Bouanan et al. 2018). In FMI, programmers have to develop a master algorithm to orchestrate the Co-Simulation's steps. The developed algorithm then, controls the synchronization and the exchange of data between the simulation models (Neema et al. 2014). FMI does not have HLA mechanisms that enables the interaction with external heterogeneous DS components. Using the objects/interactions, publish/subscribe, and time management mechanisms of HLA, such interactions become feasible. In addition, timestamp events are not supported by FMI; consequently, it becomes hard and difficult to run event-driven simulations. Furthermore, FMI is dependent from the master unit and run as one black box entity, which is not the case in HLA standard (Garro et al. 2015).

III.4 Modeling and Simulation Framework Architecture

Based on the HLA standard, we have developed a Co-Simulation framework that simulates the Actual model of an enterprise in parallel and simultaneously with the same model having Lean tools applied on. We will describe the framework's architecture of Figure III.2 in the following part.

A GUI platform is developed using JavaFx. On this platform, the user can choose the Lean tools to load then specifies the models' inputs: the market demand for each type of product needed, the setup time and processing time of each machine, the travel time between machines, the planned/unplanned down time of each machine, the defects rate, and the data related to the Lean Tools configuration. User can also start/pause/stop the simulations or change the simulation speed factor on this platform.

All data are sent or received as objects/attributes or interactions/parameters. It exists a common FOM XML file that lists all shared objects/attributes and interactions/parameters. Input data are filled in an external Java application able to interact and collaborate data with other simulation systems. This application is referred as "Master" federate in our study. The Master federate, like all the other federates, should be connected to the RTI of the federation in order to send or receive data. Based on the publish mechanism of HLA, the Master sends the input data to the connected federates and based on the HLA subscribe mechanism, it receives the output data from all connected Lean tools federates. We used the Java library of the Pitch pRTI (Technologies) in order to use the HLA mechanisms in our Master federate.

Simulation models' federates are designed in JaamSim (King et al. 2013), a Java based DES software. This software is used in this research instead of other simulators, because of its transparency, reliability, capability, and most importantly because it is an open-source software and can be configured to interact with third-party applications. Jaamsim by default is a black box simulator; users should pause the simulation or wait for the whole simulation run to change an input data. In addition, Jaamsim is not developed to connect to external systems and environments.

As it is an open source application, we were able to access the Java code of Jaamsim and add/change some functionalities. We created our own version of Jaamsim that makes it an HLA compatible DES software. Now, Jaamsim can interact, collaborate, and exchange data with external simulations. In our research, these functionalities were essential in order to run all the Lean tools in parallel, change their input data respectively and check the responsiveness of each tool based on the graphs and results. During the simulation run, the output data of running simulations are published to the RTI and received by the external application, which is subscribed to these output data. The Master Federate in turn will draw these data in a real time appealing graphical presentation. By varying the economic contexts during the simulation run, one can easily compare between all simulations' outputs and choose the best model that fit the organization production and financial targets.

Simulations can run on a network of processes, on different machines and different operating systems. Moreover, heterogeneous data are exchanged, processed, and synchronized between different simulations, without interpretation.

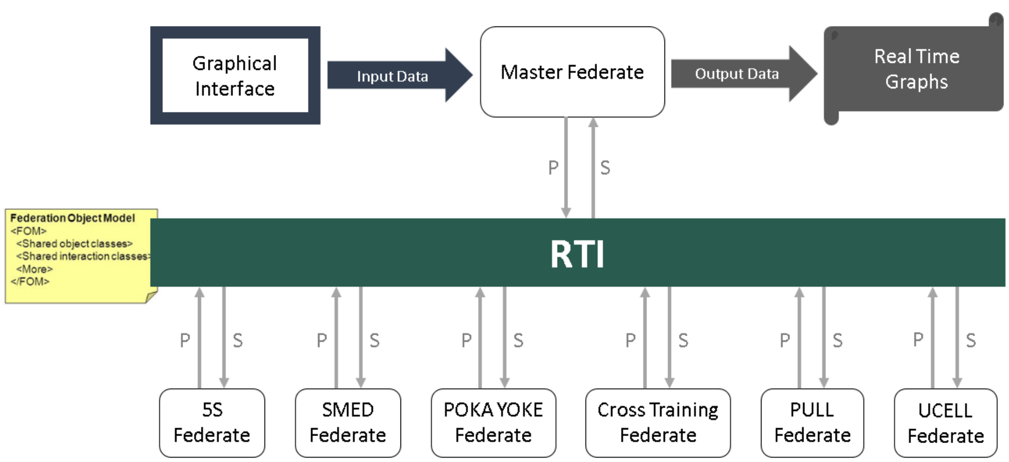


Figure III.2 Framework Architecture

We will study the case where multiple DESs run simultaneously in parallel. This work is part of a project developed to test the behavior of Lean tools and techniques during context changes. Lean Manufacturing is a systematic method that uses multiple tools and techniques in order to eliminate wastes from the manufacturing processes, improve inventory, quality, and customer satisfaction (Amrani et al. 2018). The goal of this project is to guide the companies willing to implement Lean

Manufacturing in their industries to choose the right Lean tools that suit their production processes and economic contexts.

III.5 Simulation Models Development

III.5.1 Case Study

To undertake different scenario rounds to check the adaptability of various Lean techniques in different contexts, it is essential to build a case study representing an industrial system with its inputs and outputs. The studied case study is extracted from previous collaboration of the production engineering research team (Amrani 2017). This case was the basis of Lean management implementation found in internal report dealing with Aerocomp. The choice of this case study instead of another one, is justified by the availability of the data, the closeness to the real field issues, and the possibility to split the process into different steps on the workshops in such a way to get as close as possible, a case related to discrete event simulation. Indeed, the willing is to reach discrete production line to fit with the hypothesis a discrete event simulation and the possibility to simulate at each moment (t) various disruptions and variations. The production flow is qualified as being discrete because of non-continuous flow, each part is produced at a period (t).

Choosing the aeronautic industry is related to previous works in team. This choice is justified by the tendency of team’s research, but the findings are not exclusively dedicated to aeronautic. The production process was chosen to be convergent as it is mostly the case in industry. Divergent production flow as “wood industry” has the specificity to start from common raw material piece and divergent flows creates different final products. The direction taken in this manuscript is revealing the most common tendency. The convergent process starts from different raw materials and components to assemble and produce.

Likewise, the product is pretty simple to understand, it doesn’t require technical background and accessible for any reader from other sectors. No electronic, nor mechanical and computing data are required for understanding the range. The different steps represented in different workshops are well defined with assigned duration.

Since some of the information is confidential, the company is referred to as “AeroComp” in this study. In this paragraph, we will give a clear identification of the product description, product portfolio, workshops, and range of production of Aerocomp that will be studied further later on. The product designed and manufactured by AeroComp is an aeronautic fastener composed of a metallic cylinder part over which bearings are added on the right and left sides. Gears are then welded and screwed into the back of the metallic cylinder. The metallic cylinder has a specific length and diameter provided by the client in the specification sheet. See Figure III.3.

Based on the order book of the company, raw materials are sent to the cutting shop where the metallic cylinder is cut to the exact dimensions specified by the client.

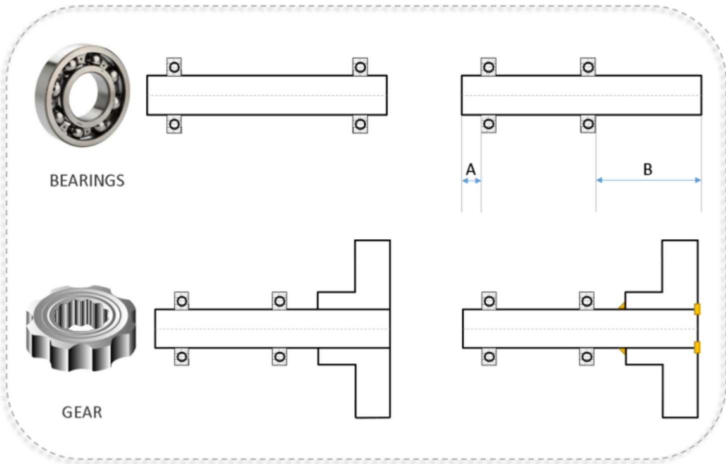


Figure III.3 AeroComp Product

Goods in process are then sent to the treatment shop where a layer of Zinc is added to the product. The product is then sent to the assembly shop where four workstations (noted WS) exist to make the semi-finished axis, add the bearings, and then fix the gears. It is finally sent to the machining shop where two workstations exist to place the pins and send the final aeronautic fastener to the warehouse for delivery. As per Figure III.4, one operator with particular skill is needed on each machine to efficiently complete the job and operate at capacity. We will describe the detailed production process in the modeling part of this chapter.

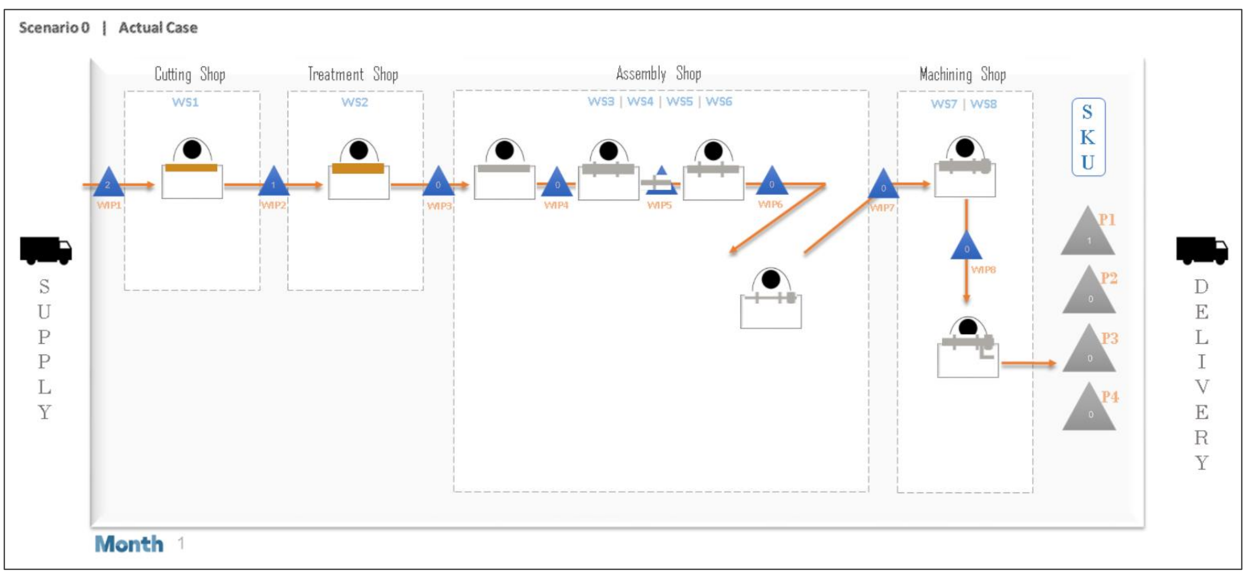


Figure III.4 Actual case model of Aerocomp

Each machine in the production line has a Processing Time and a Setup Time. Processing time is considered as the period each machine takes to complete a prescribed job or procedure. Setup/Changeover Time is defined as the period needed for the machine to switch from the last processed good of the previous batch to the first good of the new batch that has to be processed (Gest et al. 1995; Coimbra et al. 2009).

When WorkStation 1 (WS1) finishes the processing, the product is sent to WS2 and so on.

In order to switch to a new batch to be processed, each WS needs a Setup/Changeover time to prepare the machine for the new batch processing. In this study, we define $\Delta CO_{M_p XRF_i}$ as being the changeover time needed for machine "M_p" to switch to a new product reference "XRF_i".

AeroComp has a catalog of 12 different references of finished products as per Table III.1. It exists only four different diameters (12 mm, 24 mm, 32mm, and 41 mm) manufactured by AeroComp. For each diameter, length can vary regarding the client's order. There is no specific or standard length. So, $\Delta CO_{M_p XRF_i} = \Delta COD_{M_p XRF_i} + \Delta COL$, where $\Delta COD_{M_p XRF_i}$is considered as the "Diameter" changeover time needed to for machine $M_p$to switch from reference to another, and ΔCOL is the "Length" changeover time needed to for machine $M_p$to switch from reference to another. $\Delta COD_{M_p XRF_i}$ is defined for each machine M_p and each reference XRF_i.

As there is no specific or standard length, ΔCOL is calculated based on a triangular probability distribution.

Table III.1 Product Portfolio – twelve references based on four diameters and different lengths

	Ø = 12 mm	**Ø = 24 mm**	**Ø = 32 mm**	**Ø = 41 mm**
References	TX-70 TX-90	AX-80	LD-30	AX-100
	TX-80 TX-100 TX-110	BF-43	LD-40 LD-50	AX-102

$$\Delta CO_{M_p XRF_i} = \Delta COD_{M_p XRF_i} + \Delta COL$$

Travel time between machines, planned and unplanned downtimes, Work In Progress (WIP), and other details are discussed later in this chapter.

III.5.2 Models' Development

The project was developed based on version 2018-05 of Jaamsim. It can be installed on Windows or Unix operating systems. During our research studies, we added new Java modules for Jaamsim in order to convert it to an HLA compatible simulator. In this chapter, we will be working on the modified version of Jaamsim that will allow us to interconnect all existing Jaamsim components and connect them to an external application in order to collaborate and exchange data.

Once Jaamsim runs, a graphical interface will appear as per Figure III.5. User can use this graphical interface to add entities and create the simulation model or write/edit a configuration file (.cfg) in which all entities/objects can be added and configured. Some users might prefer the graphical interface to drag and drop their entities and configure them on the GUI (Graphical User Interface). Others, especially programmers, would find it faster and easier to create/edit the configuration file (.cfg).

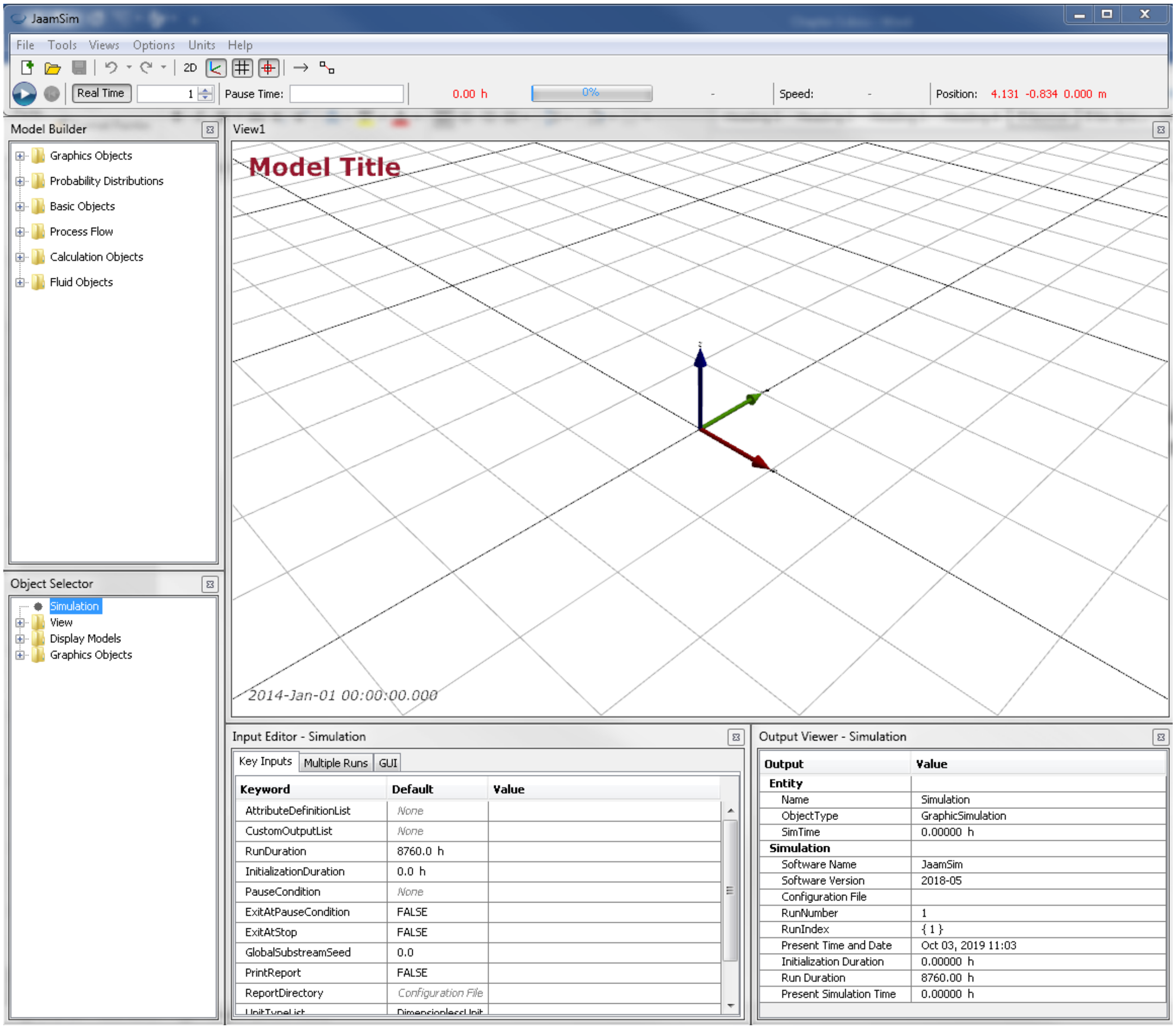


Figure III.5 Jaamsim Interface

Jaamsim provides built-in objects for building and simulating models. Users with Java programming knowledge can add new objects and edit the existing built-in objects. Jaamsim model can be launched automatically from the terminal or command line using:

```
java -jar JaamSimexecutable.jar ConfigurationFile.cfg -tags
```

Tags can be as per the following:

- "b" or "batch" to start the simulation directly and exit right after the completed simulation run.
- "m" or "minimize" to minimize the GUI during the simulation run. This will allow the simulation to run much faster as visualizations are not required.
- "h" or "headless" to run the simulation without a GUI. This tag can be used to run the simulation on a server without graphics.

The GUI of Jaamsim is divided into six main components. The first component is the control panel window that offers multiple simulation control features. The second, is the model builder in which the user can find different objects and choose the entities needed to build the simulation model. In the model builder palette, user can choose different graphic objects, probability distributions (uniform, triangular, exponential, gamma, etc.), basic objects (TimeSeries, ExpressionLogger, FileToVector, etc.), objects related to process flow (EntityGenerator, Server, Queue, Resource, Branch, etc.), calculation objects (Controller, Polynomial, Integrator, etc.), and fluid objects. See Figure III.6.

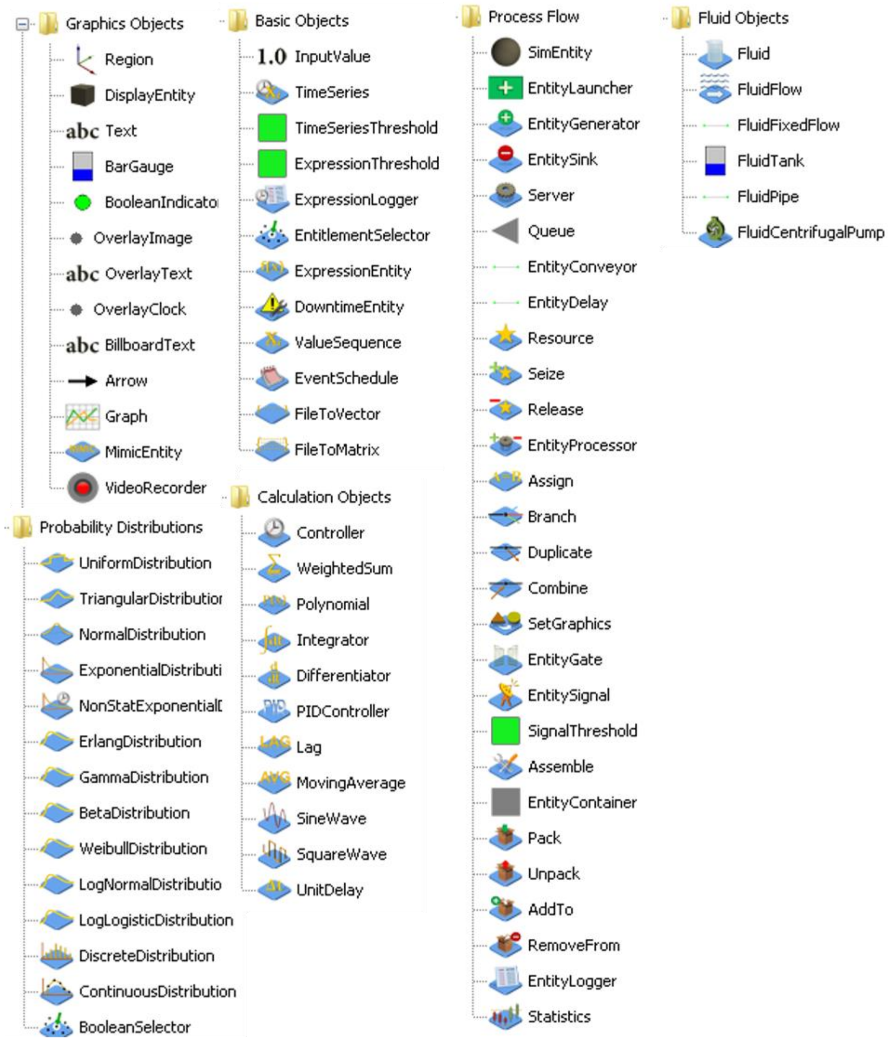


Figure III.6 Jaamsim Model Builder

The third component is the object selector in which user can find all the objects inserted in the model. The fourth one is the Input Editor where the user can edit a selected object. The fifth window is the output editor that displays all the output related to a selected object. The last window is the view window that shows the graphical representation of the simulation model.

In the section below, We will discuss the configuration of the main objects used during our models' development process.

III.5.2.1 The basic objects

a) FileToVector

FileToVector object reads a one dimensional array from a text file. Records in this file should be delimited by tabs or spaces. We used this object to create the "ProcessingTime", "TravelTime", and "NumberOfWorkers" FileToVector's objects that points to their respective text files.

Table III.2 shows the ProcessingTime vector. It is a one dimensional array that has eight elements. it represents the processing time of the respective eight workstations of our model. In FileToVector's object, we use `[FileToVector].Value(i)` to get the value of the *i* index of the array. For instance, if we need the processing time's value of WorkStation1 (WS1), we use `[ProcessingTime].Value(1)` which is equal to 2.8 minutes.

Table III.2 Processing Time Vector

2.8[min]	2.1[min]	3[min]	2.9[min]	1.8[min]	2.5[min]	3.6[min]	3.3[min]

We also used the FileToVector object to insert the TravelTime and NumberOfWorkers data inputs. The travel time is also in minutes and represents the required time to go from a machine to another. The Number of Workers is represented by dimension less unit integers.

In our model, the number of workers are represented as follows,

`NumberOfWorkers = { 1  1  4  2 }` where, `NumberOfWorkers.Value(1)` is the number of workers in the Cutting shop, `NumberOfWorkers.Value(2)` is the number of workers in the Treatment shop, `NumberOfWorkers.Value(3)` is the number of workers in the Assembly shop, and `NumberOfWorkers.Value(4)` is the number of workers in the Machining. See Figure III.4**.**

b) FileToMatrix

FileToMatrix object reads two dimensional array data from a text file. Records in this file should be delimited by tabs or spaces. We used this object to create the "Demand", "SetupTime", "PlannedDownTime", and "Unplanned DownTime" FileToMatrixs' objects that points to their respective text files.

Table III.3 shows an example of the yearly Demand matrix of the company. It is a two dimensional array that has 252 elements, 12 rows and 21 columns. The 12 rows represent the months of the year (Jan, Feb, Mar, etc.). The 1st column represents the working hours/month of the French industry. The hours are calculated based on a calendar chart of the French time working hours excluding Saturdays, Sundays, holidays, and all closed days of the year. The 2nd column represents the product type 1, the 3rd represents product type 2, and so on, till the 21st column that represents product type 20.

`[Demand].Value(x)(y)` is a specific record in the matrix, where x and y are integers. "x" represents the row number in the matrix and "y" represents the column number. `[Demand].Value(2)(3)`, for instance, will give the value 413.

In Table III.3, we can see an example of the yearly order book for only four types of products.

Table III.3 Order Demand Matrix

168.6666667[h]	350	463	437	386	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
153.3333333[h]	400	413	403	386	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
168.6666667[h]	389	453	445	376	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
153.3333333[h]	339	453	434	350	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
145.6666667[h]	375	485	445	329	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
161[h]	386	434	437	386	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
168.6666667[h]	397	429	425	363	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
168.6666667[h]	399	458	413	388	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
153.3333333[h]	408	480	435	385	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
176.3333333[h]	417	488	465	394	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
161[h]	395	515	456	400	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
153.3333333[h]	384	485	448	382	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

We also used the FileToMatrix object to define the SetupTime. Setup time is required to change the settings, for the machine to be able to operate a different line of product; if consecutive products have the same reference, no setup time is required. `[SetupTime].Value(x)(y)` is used to represent a specific record in the matrix, where x and y are integers. "x" represents the row number in the matrix and "y" represents the column number. The row number define the Workstation number; for instance, row1 represents WS1. The column define the product type, column1 means product type 1 or XRF_1. For example, in Table III.4, `[SetupTime].Value(2)(4)` is the setup time required for WS2 to change its settings and switch to start producing product reference 4 (XRF_4).

Table III.4 Setup Time per Workstation/Product Reference

2[min]	3[min]	2[min]	3[min]
3[min]	3[min]	3[min]	3[min]
1[min]	3[min]	2[min]	3[min]
2[min]	3[min]	2[min]	3[min]
2[min]	3[min]	3[min]	3[min]
3[min]	3[min]	3[min]	3[min]
2[min]	3[min]	3[min]	3[min]
2[min]	3[min]	3[min]	3[min]

In Table III.5, we can see an example of the PlannedDownTime configuration based on the FileToMatrix object. A planned downtime is a period required to implement machine upgrades, planned system maintenance, machine cleaning, and others. During this period, the system or machine cannot be active. The first row of this matrix represents the interval of time between a planned downtime and another. The second row represents the downtime period. The first column represents the first machine in the production line (WS1), the second represents WS2, and so on, till WS8.

Table III.5 Planned Down Time

168.66[h]	1932[h]	1932[h]	1932[h]	1932[h]	483[h]	1932[h]	1932[h]
50[min]	0[min]	0[min]	0[min]	0[min]	135[min]	0[min]	0[min]

The unplanned downtime has the same configuration procedure as the planned down time using FileToMatrix. Unplanned downtime is any software/hardware error, operator error, or unforeseen event that stops the machine and makes it unavailable.

c) Downtime Entity

Downtime Entity object has "Interval" and "Duration" inputs. We created 8 PlannedDownTime objects for the 8 exisiting workstations. As per Figure III.7, in the "Interval" field of WS1PlannedDowntime entity, we put the value of the interval time between planned downtimes for machine WS1 which is `[PlannedDowntime].Value(1)(1)` equals to 168.66 hours. As we stated before, PlannedDownTime values are accessed through the FileToMatrix object. The "Duration" input of the aforementiond planned downtime is `[PlannedDowntime].Value(2)(1)` equals to 50 mins.

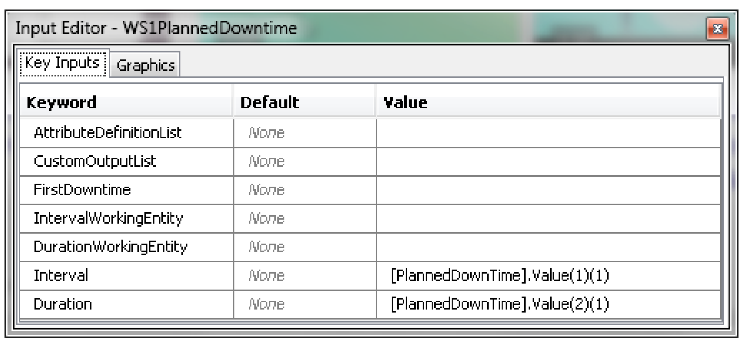


Figure III.7 Planned Down Time Object of WS1

Using the Downtime Entity, we also created 8 UnplannedDowntime objects for the 8 exisiting workstations. When planned or unplanned downtimes occur, the "WorkingState" output of the object switches from TRUE to FALSE.

d) ExpressionThreshold

ExpressionThreshold object has an "OpenCondition" and a "CloseCondition" inputs.

For the OpenCondition input, the algorithm of Algorithm III.1 is applied.

Algorithm III.1 ExpressionThreshold Open Condition

```
if OpenCondition value == 0 then
       do [ExpressionThreshold].Open = FLASE
//the [ExpressionThreshold].State will be "Close"
else
       do [ExpressionThreshold].Open = TRUE
//the [ExpressionThreshold].State will be "Open"
```

Regarding CloseCondition input; if it is not specified, it will be the opposite of the OpenCondition value. If it is specified, it will have the return of Algorithm III.2.

Algorithm III.2 ExpressionThreshold Close Condition

```
if CloseCondition value == 0 then
       do [ExpressionThreshold].Open = TRUE
//the ExpressionThreshold State will be "Open"
else
       do [ExpressionThreshold].Open = FALSE
//the ExpressionThreshold State will be "Close"
```

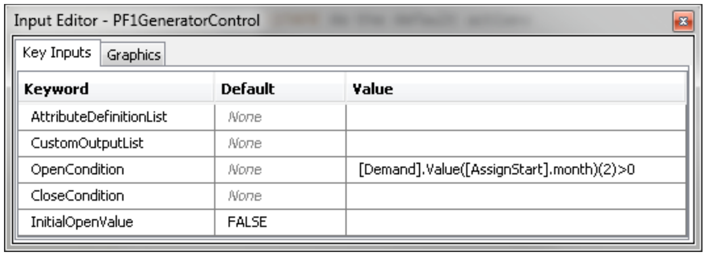
The ExpressionThreshold has also an InitialOpenValue that can be TRUE or FALSE (See Algorithm III.3).

Algorithm III.3 ExpressionThreshold Initial Open Value

```
if InitialOpenValue == TRUE then
      do [ExpressionThreshold].Open = TRUE
//the ExpressionThreshold State will be "Open"
else
      do [ExpressionThreshold].Open = TRUE
//the ExpressionThreshold State will be "Open"
```

In our model we used this object to control the EntityGenerator. If the ExpressionThreshold is in "Open" state the EntityGenerator will allow the rawmaterial generation. Otherwise, it will block the generation of new SimEntities.

In the actual model, we used 20 ExpressionThresholds to control 20 EntityGenerators. Each EntityGenerator generates a Type or Reference of production. Our Simulation model can produce 1 to 20 different references. In Figure III.8, we show the ExpressionThreshold "PF1GeneratorControl" that will control the raw material's generation of product type 1. This expressionthreshold is applied to the EntityGenerator "PF1Generator".


Input Editor - PF1GeneratorControl

Key Inputs | Graphics

Keyword	Default	Value
AttributeDefinitionList	None	
CustomOutputList	None	
OpenCondition	None	[Demand].Value([AssignStart].month)(2)>0
CloseCondition	None	
InitialOpenValue	FALSE	

Figure III.8 The OpenCondition of the PF1GeneratorControl ExpressionThreshold

As we stated before, demand is a FileToMatrix object that points to a matrix file containing a 2 dimensional array. As per Figure III.8, the OpenCondition of PF1GeneratorControl is:

`[Demand].Value([AssignStart].month)(2)>0`

This means that the "OpenCondition" of PF1GeneratorControl is based on the monthly demand of the first type of product to produce. If the value was zero, PF1GeneratorControl will close its state. Thus, PF1Generator will be blocked and unable to generate product type 1. See the pseudo code of Algorithm III.4.

Algorithm III.4 PF1GeneratorControl OpenCondition

```
if [Demand].Value of the 1st type of Product on
month [AssignStart].month == 0 then
      do [PF1GeneratorControl].Open = FALSE
//the ExpressionThreshold State will be "Close"
else
      do [PF1GeneratorControl].Open = TRUE
//the ExpressionThreshold State will be "Open"
```

Consequently, the OpenCondition of PF2GeneratorControl is:

`[Demand].Value([AssignStart].month)(3)>0`

We use the same development procedure for all existing product types.

III.5.2.2 The process flow objects

a) SimEntity

SimEntity object serves as the prototype that will be processed in the objects of the model. we used this entity to create twenty product types. As Aerocomp produces only four different diameters or references, only four "SimEntity" entities are needed. However, we created 20 entities of this object to test the diversification of demand industrial context noted <ctx.2> in the previous chapter.

Each of the aforementioned entities have three attributes, the "Type", the "leadTime", and the "defective" attributes. In Figure III.9, The AttributeDefinitionList of Product Type 1 noted (PF1) is, `{Type 1}{leadTime 0 h}{defective 0}`. For PF2, the AttributeDefinitionList is `{Type 2}{leadTime 0 h}{defective 0}`, for PF3, the AttributeDefinitionList is `{Type 3}{leadTime 0 h}{defective 0}`, etc.

The first attribute is “Type”. It is an integer varying from 1 to 20 in our case. “leadTime” attribute is calculated in hours (h) and initialized to “0 h”. “defective” attribute is an integer initialized to “0”. Defective attribute can be “0”, which means “not defective”, or “1”, which means “defective”.

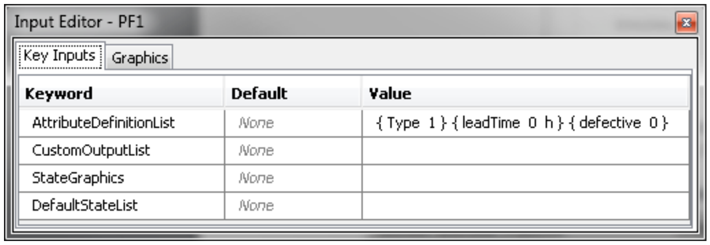


Figure III.9 SimEntity of Product Type 1 "PF1"

b) EntityGenerator

EntityGenerator object generates copies of the prototype SimEntity. In our model, we created 20 EntityGenerators to make copies of the 20 exisiting prototypes or SimEntities. For the first EntityGenerator object called “PF1Generator”, the assigned ProtoypeEnity is “PF1”. It represents product type 1. “PF2Generator” has PF2 as prototype entity, etc.

The “NextComponent” input accepts the entity name of the next object to which the SimEntity will be passed. As per Figure III.10, the Nextcomponent of PF1Generator is the AssignStart object entity. We will discuss the AssignStart object in the Assign paragraph of the process flow section. The “InterArrivalTime” input is the time between generated Entities. This input accepts a number with TimeUnit type. The InterArrivalTime of PF1 is calculated as per the following,

```
[Demand].Value([AssignStart].month)(1)/[Demand].Value([AssignStart].month)(2)
```

where [Demand] is a FileToMatrix object discussed in the previous paragraphs.

In the above formula, `[Demand].Value([AssignStart].month)(1)` points to the first column of [Demand] matrix that represents the working hours/month.

`[Demand].Value([AssignStart].month)(2)` points to the second column. It represents the demand quantity of PF1/month.

We have divided each month’s working hours over its corresponding market demand quantity of PF1. In this way, each EntityGenerator generates the monthly quantity required of a specific predefined product type. For instance, in the case of product type 1, the InterArrivaltime value changes based on the month’s working hours and the demand quantity of PF1 for the corresponding month. It is important to note that this InterArrivalTime value is only for PF1.

For PF2, the InterArrivalTime is as follows,

```
[Demand].Value([AssignStart].month)(1)/[Demand].Value([AssignStart].month)(3)
```

We use the same procedure to calculate the InterArrivalTime of all existing products, ranging from PF1 to PF20.

Input Editor - PF1Generator

Key Inputs | Thresholds | Maintenance | Graphics

Keyword	Default	Value
AttributeDefinitionList	None	
CustomOutputList	None	
StateGraphics	None	
NextComponent	None	AssignStart
FirstArrivalTime	0.0 h	
InterArrivalTime	2.77777777777...	[Demand].Value([AssignStart].month)(1)/[Demand].Value([AssignStart].month)(2)
EntitiesPerArrival	1.0	
PrototypeEntity	None	PF1
BaseName	Generator Name	
MaxNumber	Infinity	

Figure III.10 The EntityGenerator "PF1Generator" that generates the raw materials of PF1

In the "Thresholds" tab of PF1Generator, we call the "PF1GeneratorControl" created previously in order to control the flow of PF1Generator that generate raw materials of Type 1. We put the "PF1GeneratorControl" expression threshold in the "OperatingThresholdList" input. See Figure III.11.

Input Editor - PF1Generator

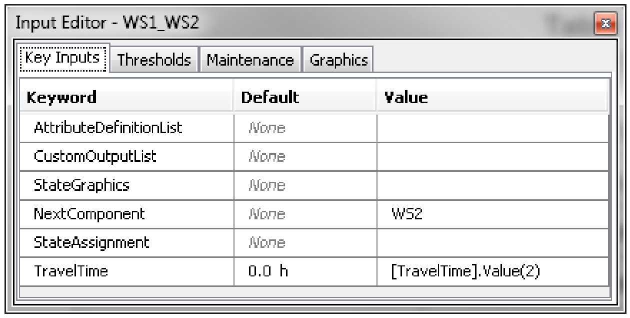
Key Inputs | Thresholds | Maintenance | Graphics

Keyword	Default	Value
ImmediateThresholdList	None	
ImmediateReleaseThresho...	None	
OperatingThresholdList	None	PF1GeneratorControl

Figure III.11 Threshold List of PF1Generator

c) EntityConveyor

EntityConveyor object is used to transport an entity on a specified path at a specific speed. As per Figure III.12, the travel time needed to go from an entity to another is defined in "TraveTime" input. As we discussed before, the "TravelTime" EntityVector has all the values of the travel times needed between entities. The travel time defined in Figure III.12 is named [WS1_WS2]. It represents the time needed to go from machine WS1 to machine WS2. The value of this travel time is `[TravelTime].Value(2)`. We use the same procedure to define the travel time of [WS2_WS3], [WS3_WS4], etc. The "NextComponent" input of [WS1_WS2] EntityConveyor is WS2 object. The "NextComponent" of [WS2_WS3] conveyor is WS3 object, etc.



Keyword	Default	Value
AttributeDefinitionList	None	
CustomOutputList	None	
StateGraphics	None	
NextComponent	None	WS2
StateAssignment	None	
TravelTime	0.0 h	[TravelTime].Value(2)

Figure III.12 Travel Time between machines "WS1" and "WS2"

d) EntityProcessor

EntityProcessor object processes the incoming SimEntity then sends it to the next object. Entities waiting to be processed are sent to the Queue object. In Figure III.14, we show the configuration of “WS1” EntityProcessor. The following inputs of WS1 processor are used, “StateGraphics”, “NextComponent”, “WaitQueue”, “ResourceList”, and “ServiceTime”.

The StateGraphics input is used to change the EntityProcessor graphics based on the State of this Entity. The Sates used in our model are as per the following, `{ Idle NoWorker } { Working Worker } { Maintenance Maintenance } { Breakdown Breakdown }`. If the WS1 EnityProcessor state is “Idle”, the graphical representation of WS1 will be as per Figure III.13 (a), “Working” State is represented by Figure III.13 (b), “Maintenance” State is represented by Figure III.13 (c), and “Breakdown” State is represented by Figure III.13 (d).

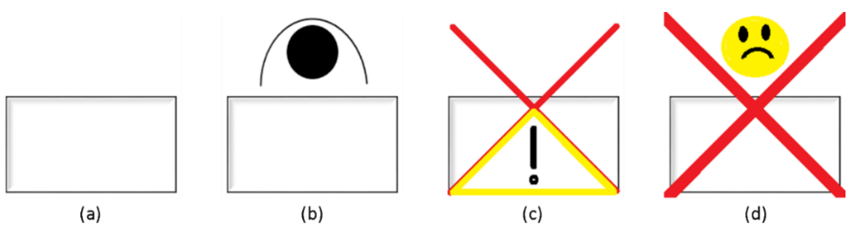
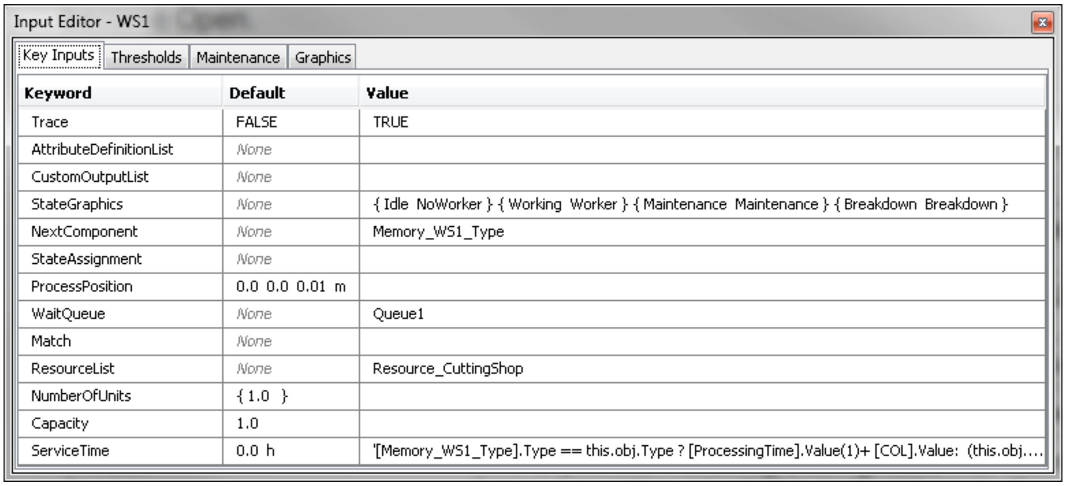


Figure III.13 EntityProcessor State Graphics

NextComponent of WS1 object is [Memory_WS1_Type] object. This object is an “Assign” object that memorizes the type of the previous processed entity. By memorizing the processed entity, we can know if the new entity to be proceessed has to wait for the setup time of the machine or it can be directly processed. The WaitQueue input informs the entity to store the waiting SimEntities in the assigned Queue object, “Queue1” in this case. The “ResourceList” input accepts a list of resources from which units can be seized.



Input Editor - WS1

Key Inputs | Thresholds | Maintenance | Graphics

Keyword	Default	Value
Trace	FALSE	TRUE
AttributeDefinitionList	None	
CustomOutputList	None	
StateGraphics	None	{ Idle NoWorker } { Working Worker } { Maintenance Maintenance } { Breakdown Breakdown }
NextComponent	None	Memory_WS1_Type
StateAssignment	None	
ProcessPosition	0.0 0.0 0.01 m	
WaitQueue	None	Queue1
Match	None	
ResourceList	None	Resource_CuttingShop
NumberOfUnits	{ 1.0 }	
Capacity	1.0	
ServiceTime	0.0 h	'[Memory_WS1_Type].Type == this.obj.Type ? [ProcessingTime].Value(1)+ [COL].Value: (this.obj....

Figure III.14 The EntityProcessor "WS1"

In Figure III.14, the ResourceList input is “Resource_CuttingShop”, it contains the number of operators needed in the cutting shop. In our model, we have only one operator in the cutting shop. So, only one operator can be seized from the ResourceList “Resource_CuttingShop”.

The ServiceTime is considered as the time needed to process an entity. It accepts a number with TimeUnit type. The service time of the EntityProcessor WS1 is calculated as per the pseudo-code of Algorithm III.5.

Algorithm III.5 ServiceTime of the EntityProcessor "WS1"

```
int x = this.obj.Type;

if this.obj.Type == [Memory_WS1_Type].Type then
   do ServiceTime = [ProcessingTime].Value(1)+ [COL].Value
else
   do SerivceTime = [SetupTime].Value(1)(x)  + [ProcessingTime].Value(1)+ [COL].Value
```

`this.obj.Type` gives the type of the current SimEntity that will be processed in WS1.

`[Memory_WS1_Type].Type` refers to the previous type of the SimEntity processed in WS1. `[COL].Value` is the value of time required to setup the machine in order to produce a specific Length. As we stated before, there are no predefined products Lengths. So, we consider that the machine needs a setup time of `[COL].Value` for each SimEntity.

As for the product diameter (type), we modeled 20 different diameters. We consider that product types are based on their diameters and not their lengths. For instance, `[SetupTime].Value(1)(x)` is considered as the time needed to prepare the machine to produce type x of product that have a defined diameter x'.

e) Resource

Resource is an object that holds a pool of identical units. These units can be seized and released. In the model we developed, the units represents the operators at each shop. A shop can have multiple machines or workstations. As per Figure III.4, we have one operator (unit) in the cutting shop, one operator in the treatment shop, four operators in the assembly shop, and two operators in the mahcining shop. The “Capacity” in the resource object is by default 1. As per Figure III.15, the “Capacity” of workers or operators in the cutting shop refers to the first element of the EntityVector [NumberOfWorkers].

We created four Resource entities, the “Resource_CuttingShop”, the “Resource_TreatmentShop”, the “Resource_AssemblyShop”, and the “Resource_MachiningShop”. Each of these resources’ capacity is assigned in the EntityVector [NumberOfWorkers].

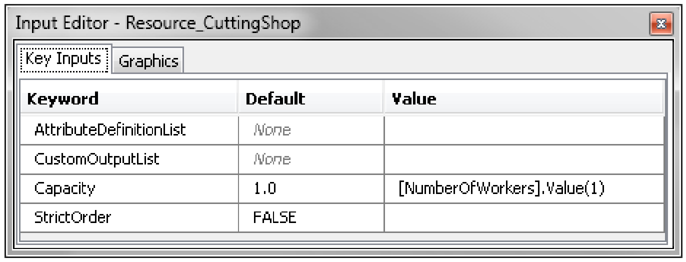


Figure III.15 Capacity of the Resource "Resource_CuttingShop"

f) Queue

Queue is an object that stores the received SimEntities and hold them until they are needed. In our model, each workstation has its own Queue. One can check the queue length of each queue by calling the QueueLength function. For instance, Queue1 has `[Queue1].QueueLength` SimEntities waiting in its Queue.

g) Assign

Assign is an object that makes one or more assignements to the attributes. This is the only place where objects and entities’ attributes can be modified. We create different “Assign”. We will list some important “Assign” objects created. The others can be found in Appendix B.

“AssignStart” and “AssignEnd” objects are created to calculate the lead time of each SimEntity produced, as well as the mean value of lead time for all the products produced. “AssignStart” is also used to tag some SimEntities as “Defective” based on a boolean selector probability and to determine the “Month” during the simulation run.

An “AssignDefective” object is also created to count the number of defective products and send them back to the “AssignStart” object in case of defective.

Different Assign objects were also created to memorize the type of the previous product produced on each machine. [Memory_WS1_Type] object will have the type of the previous product produced on machine WS1, [Memory_WS2_Type] object will have the type of the previous product produced on machine WS2, etc.

In order to calculate the production lead time, we assigned some attributes in “AttributeDefinitionList” of AssignStart and AssignEnd objects.

In the AssignStart AttributeDefinitionList we add,

```
{ this.obj.leadTime=[Simulation].SimTime }
```

In the above line of code, we assign the simulation time to the attribute "leadTime" of the SimEntity that accessed the AssignStart object. It is important to note that the AssignStart object is accessed by raw materials before starting the production process.

When the SimEntity finishes the production process, it passes by the AssignEnd object where the product's lead time is calculated. In the AssignEnd AttributeDefinitionList we add,

```
{ this.obj.leadTime=[Simulation].SimTime-this.obj.leadTime }
```

In the above line of code, the value of leadTime assigned to the SimEntity in the AssignStart object is subtracted from the current simulation time. This will give the exact leadTime of each SimEntity produced from the moment the order is received until the delivery of the finished product.

We also used this object to calculate the total WIP during the simulation run, which is equal to the sum of all items in the workstations' queues and the sum of all goods under process, as per the following,

```
{ this.WIP=([Queue1].QueueLength+…+[Queue8].QueueLength)+[WS1].WorkingState
+…+[WS8].WorkingState }
```

Additional Assign objects can be found in Appendix B.

h) Branch

Branch object is used to branch the received SimEntity to a selected destination based on "Choice" input values. We will show the two main Branches created in our model. The first branch object is "Branch_To_Defective". "Choice" in Figure III.16 is used to choose the NextComponent branch. If choice is "1", SimEntity will be branched to the "Branch_To_Sink" object where the SimEntity will be sent to the respective "EntitySink" that represents the end of its production process. If choice is "2", SimEntity will be directed to the "AssignDefective" object discussed in the previous paragraph.

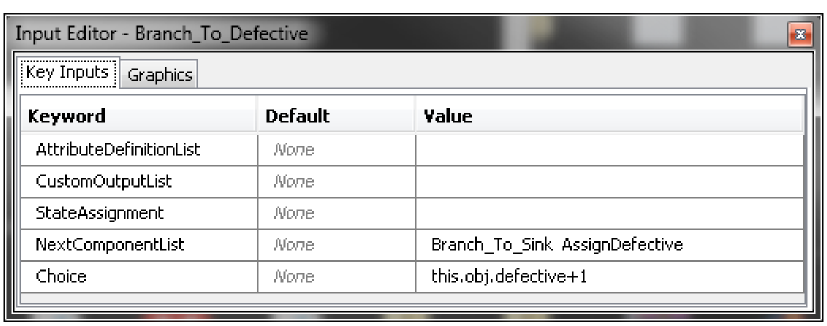


Figure III.16 The Branch Oject "Branch_To_Defective"

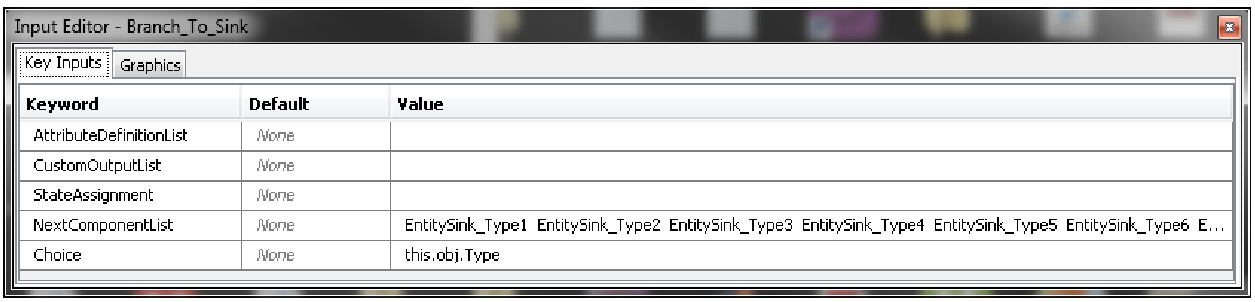
"this.obj.defective" will return 0 or 1. If the value is 0, it means that the product is not defective. If the value is 1, it means that the product is defective. "defective" is a boolean attribute given to each SimEntity. See Algorithm III.6 for more detailed and clear explanation.

Algorithm III.6 Branch_To_Defective pseudo-code

```
if this.obj.defective + 1 == 1
//it means that this.obj.defecive is 0 → NOT DEFECTIVE
then
do Branch to [Branch_To_Sink] object
else if this.obj.defective + 1 == 2
//it means that this.obj.defecive is 1 → DEFECTIVE
then
do Branch to [AssignDefective] object
```

In Figure III.17, the [Branch_To_Sink] object will branch the SimEntity to its respective EntitySink Type. If the SimEnity or the product was of Type 1, it is then sent to "EntitySink_Type1". If the SimEnity or the product was of Type 2, it is sent to "EntitySink_Type2", etc.

The EntitySink objects are accessed at the last production stage of the SimEntity.



Input Editor - Branch_To_Sink

Key Inputs | Graphics

Keyword	Default	Value
AttributeDefinitionList	None	
CustomOutputList	None	
StateAssignment	None	
NextComponentList	None	EntitySink_Type1 EntitySink_Type2 EntitySink_Type3 EntitySink_Type4 EntitySink_Type5 EntitySink_Type6 E...
Choice	None	this.obj.Type

Figure III.17 Branch_To_Sink Object

i) SetGraphics

SetGraphics is an object used to modify the appearance of a specific entity. We created different SetGraphics object in our models to simplify and clarify the simulation process of our chosen model. All created SetGraphics objects are shown in Apendix A.

j) EntitySink

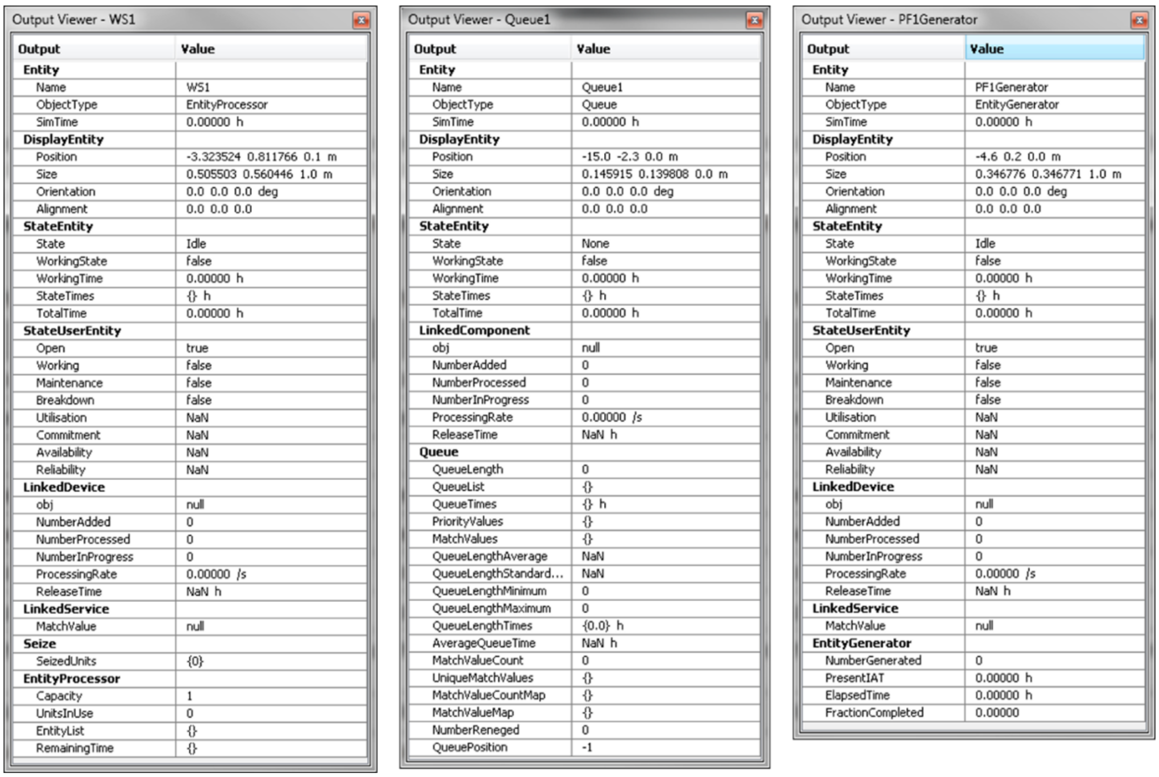
is an object that destroys the received entity. In our model, we use the EntitySink objects at the end of the product production process. We created 20 EntitySink objects. Each product Type is sent to its respective EntitySink. As per the "Branch" paragraph example, each product, based on its type, is branched to its EntitySink to destroy it. This way, we will be able to have the production rate of each type by calling the function "NumberProcessed". For instance, in order to know the number processed of products of type 1, we can use `[EntitySink_Type1].NumberProcessed`.

II.5.2.3 Output Viewer

The Output Viewer tool in JaamSim is an interesting tool to display the selected object's outputs. Those Outputs are also used in the model's configuration. As per Figure III.18, we show three objects' outputs; the output of the EntityProcessor "WS1", the output of the Queue entity

"Queue1", and the output of the EntityGenerator "PF1Generator". Some of the outputs are common in most of the objects, such as "Name", "ObjectType", "SimTime", "NumberAdded", "NumberProcessed", etc. Others are object related functions such as "QueueLength", "QueueList", "NumberGenerated", etc. In order to call these outputs functions, the developer should use the following structure, `[Entity].Output`. For instance, `[WS1].NumberProcessed` will give the number of SimEntities processed by the machine WS1, `[Queue1].QueueLength` will give the number of Entities the Queue 1.

The output of these functions can be of different types; TimeUnit, DistanceUnit, SpeedUnit, DimensionLessUnit, etc. Some of these outputs are used during HLA implementation in order to manage the simulation, exchange data, and collaborate the HLA Objects/Interactions between all parallel running simulations and the external Java application (Master Federate).



Output Viewer - WS1

Output	Value
Entity	
Name	WS1
ObjectType	EntityProcessor
SimTime	0.00000 h
DisplayEntity	
Position	-3.323524 0.811766 0.1 m
Size	0.505503 0.560446 1.0 m
Orientation	0.0 0.0 0.0 deg
Alignment	0.0 0.0 0.0
StateEntity	
State	Idle
WorkingState	false
WorkingTime	0.00000 h
StateTimes	{} h
TotalTime	0.00000 h
StateUserEntity	
Open	true
Working	false
Maintenance	false
Breakdown	false
Utilisation	NaN
Commitment	NaN
Availability	NaN
Reliability	NaN
LinkedDevice	
obj	null
NumberAdded	0
NumberProcessed	0
NumberInProgress	0
ProcessingRate	0.00000 /s
ReleaseTime	NaN h
LinkedService	
MatchValue	null
Seize	
SeizedUnits	{0}
EntityProcessor	
Capacity	1
UnitsInUse	0
EntityList	{}
RemainingTime	{}

Output Viewer - Queue1

Output	Value
Entity	
Name	Queue1
ObjectType	Queue
SimTime	0.00000 h
DisplayEntity	
Position	-15.0 -2.3 0.0 m
Size	0.145915 0.139808 0.0 m
Orientation	0.0 0.0 0.0 deg
Alignment	0.0 0.0 0.0
StateEntity	
State	None
WorkingState	false
WorkingTime	0.00000 h
StateTimes	{} h
TotalTime	0.00000 h
LinkedComponent	
obj	null
NumberAdded	0
NumberProcessed	0
NumberInProgress	0
ProcessingRate	0.00000 /s
ReleaseTime	NaN h
Queue	
QueueLength	0
QueueList	{}
QueueTimes	{} h
PriorityValues	{}
MatchValues	{}
QueueLengthAverage	NaN
QueueLengthStandard...	NaN
QueueLengthMinimum	0
QueueLengthMaximum	0
QueueLengthTimes	{0.0} h
AverageQueueTime	NaN h
MatchValueCount	0
UniqueMatchValues	{}
MatchValueCountMap	{}
MatchValueMap	{}
NumberReneged	0
QueuePosition	-1

Output Viewer - PF1Generator

Output	Value
Entity	
Name	PF1Generator
ObjectType	EntityGenerator
SimTime	0.00000 h
DisplayEntity	
Position	-4.6 0.2 0.0 m
Size	0.346776 0.346771 1.0 m
Orientation	0.0 0.0 0.0 deg
Alignment	0.0 0.0 0.0
StateEntity	
State	Idle
WorkingState	false
WorkingTime	0.00000 h
StateTimes	{} h
TotalTime	0.00000 h
StateUserEntity	
Open	true
Working	false
Maintenance	false
Breakdown	false
Utilisation	NaN
Commitment	NaN
Availability	NaN
Reliability	NaN
LinkedDevice	
obj	null
NumberAdded	0
NumberProcessed	0
NumberInProgress	0
ProcessingRate	0.00000 /s
ReleaseTime	NaN h
LinkedService	
MatchValue	null
EntityGenerator	
NumberGenerated	0
PresentIAT	0.00000 h
ElapsedTime	0.00000 h
FractionCompleted	0.00000

Figure III.18 Output Viewer

III.5.2.4 Lean Models' development

The Actual Model (Scenario 0) represents the actual case of the aeronautic fasteners industry configured as a Lean Free Scenario. In Scenario 0, twenty EntityGenerators are created to generate twenty different types of Raw Materials. The number of product types and the yearly order book of the company are filled in the "Demand" matrix. In the production process of Scenario 0, raw materials are sent to the WS1 machine in the Cutting Shop. WS1 has a Setup Time and a Processing Time defined previously. After being processed, goods are sent to WS2, and so on, until the delivery process. Entity Conveyors are used to specify the travel time between machines. "WIP1" is the Work In Progress of the First Machine, WIP2 is the Work In Progress of the second Machine, and so on. The same model structure is used for the remaining six models that reproduces the same company having one of the Lean tools applied to its production processes.

The Pull production method strives to minimize and eliminate overproduction. In Pull scenario's configuration, each machine sends a signal to the upstream one when its WIP exceeds a predefined number of units to stop sending products in process. In order to generate this signal, we have created multiple ExpressionThreshold objects to control the flow of production on each machine. "PULLonWS1" is an ExpressionThreshold that controls the raw materials flow, "PULLonWS2" is an ExpressionThreshold that controls the flow coming from the machine WS1, etc.

For instance, the open condition of the ExpressionThreshold "PULLonWS1" is `[Queue1].QueueLength<[Pull].Value(1)`. It means that the PULLonWS1 will have its state "Open" only if the entities in Queue1 are less than the value of the first element of the vector [PULL]; otherwise, the PULLonWS1 will have its state "Close". We add the PULLonWS1 ExpressionThreshold to the "OperatingThresholdList" of all EntityGenerators; PF1Generator, PF2Generator,…, PF20Generator. We add the PULLonWS2 ExpressionThreshold to the "OperatingThresholdList" of WS1 EntityProcessor, PULLonWS3 ExpressionThreshold is added to the "OperatingThresholdList" of WS2 EntityProcessor, etc. In this way, the flow is controled on each machine using ExpressionThreshold objects.

In SMED tool, Setup Time of each machine is reduced. SMED goals are to minimize/eliminate the waste resulting from lack of material, to ensure tools and machine cleanness and to organize the workshop place associated with setup/changeover processes. On each workstation, we reduce the service time by a certain percentage. This percentage represents the setup time reduction that can be attained by each machine or workstation. The setup time reduction is a percentage of reduction represented by `[SetupTimeReduction].Value(x)` where x is the index of the SetupTimeReduction Vector. For instance, WS1 is affected if x is equal to 1, WS2 is affected if x is equal to 2, etc. We implement this reduction in the ServiceTime input of each workstation. Thus, in SMED, the setup time calculated previously is now multiplied by `{1-[SetupTimeReduction].Value(x)}`.

5S tool aims to make a self- explaining, ordering and improving workplace. It is a set of principles that improve the workplace environment which in turn improve the quality and the production efficiency. On each workstation, we reduce the service time by a certain percentage. This percentage represents the production time and defects reduction that can be attained by each machine or workstation. The processing/defects reduction is a percentage of reduction represented by `[ProcessingAndDefectsReduction].Value(x)` where x is the index of the ProcessingAndDefectsReduction Vector. For instance, WS1 is affected if x is equal to 1, WS2 is affected if x is equal to 2, etc. We implemented this reduction in the ServiceTime input of each workstation. Thus, in 5s, the processing time of each machine is multiplied by `{1-[ProcessingAndDefectsReduction].Value(x)}`.

Cross training aims to achieve multi-skilling for workers. This increases the work variety and creates a balanced workload between the operators. To develop the multi-skilling model, we created a new Resource object called [Resource_CrossTraining]. The capacity of the

Resource_CrossTraining is filled in the EntityVector [NumberOfWorkers]. In each “ResourceList” of the EntityProcessors (WS1, WS2, etc.), we import the Resource_CrossTraining object. In this model, we consider that all workers are multi-skilled and are ready to operate on any machine and cover operator’s absences. For instance, in the model of Figure III.4, if an operator is absence, the eight machines will remain operating, controlled by seven workers.

Ucell focuses on the flow of the product. Machines are placed close to each other in order to minimize the transport time between them. In the model development of Ucell, the travel time between workstations is reduced by a certain percentage. This percentage represents the travel time reduction that can be attained on each EntityConveyor. The travel time reduction is a percentage of reduction represented by `[TravelTimeReduction].Value(x)` where x is the index of the TravelTimeReduction Vector. For instance, if x is equal to 1, the reduction affects the travel time between the raw material generation and the first workstation (WS1), if x is equal to 2, the reduction affects the travel time between WS1 and the second workstation (WS2), etc. We implemented this reduction in each of the EntityConveyor objects created. The travel time of each conveyor is now multiplied by `{1-[TravelTimeReduction].Value(x)}`.

Poka Yoke means “mistake-proofing”. This tool is a simple tool that prevents defective good in process from being delivered to the next process. The main concept of this approach is to detect, eliminate, and correct errors at their current source before reaching the customer. We created an EntityVector named [DefectsReduction] in which we assign the percentage of defects reduction that can be attained by the company. In the Probability Distributions, we already mentioned the BooleanSelector object in which we assign the TrueProbabilty of the Selector returning “TRUE”. To change the existing TrueProbability input, we substract the defects reduction percentage from the existing value of the defective probability. The configuration files of these models can be found in Appendix B.

III.6 Co-Simulation Framework Development

In this section, we will describe the development process of the Co-Simulation framework using a BPMN/HLA-based methodology for collaborative Distributed DES (Possik et al. 2019). We used BPMN to clear up the proposed methodology and simplify the understanding of the integration and collaboration between discrete event simulators. The main goal of BPMN is to provide a detailed visual representation of the complete sequence of business activities and information flows and deliver a standard notation easily readable by non-expert users. We will explain briefly the basic BPMN representation in the section below.

III.6.1 BPMN Standard

BPMN is a business process-modeling standard that offers a graphical notation based on a flowcharting technique. BPMN represents the end-to-end flow of a process. The Business Process Management Initiative (BPMI) developed the Business Process Modeling standard. In 2005, this group merged with the OMG (Object Management Group). In 2011, OMG released the BPMN 2.0 version release and changed the name of the method to Business Process Model and Notation. This Business Process Modeling standard became more detailed by using a richer set of symbols and notations for business process diagrams. Its purpose is to increase efficiency. It is the enterprise equivalent of the Unified Modeling Language (UML) used in software design.

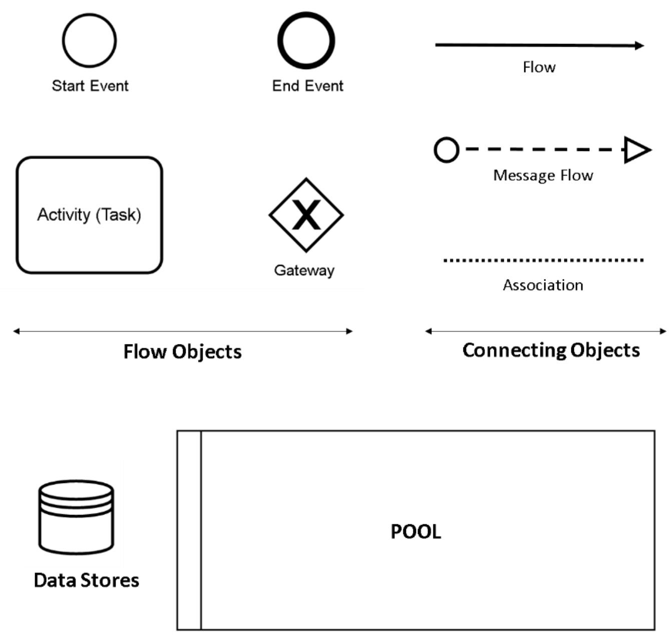


Figure III.19 Basic BPMN elements used

We will explain briefly in this part the basic BPMN elements that we used in this chapter to clarify the development process and the implementation stages. Flow Objects are the main graphical elements, which are used to define the behavior of a process. They are Events, Activities, and Gateways. An Event represents the concept of something that happens. It can represent the start or the end of a process. An event is displayed as a circle. An activity represents a portion of work or step to be done during the process. It is represented as a rounded-corner rectangle. A gateway represents the behavior of the process flow to specify its convergence and divergence. Using gateways, we can express different branching types in the execution flow (i.e., merge, join, fork, decisions, etc.). A gateway is represented with a diamond shape. Connecting Objects connect Flow Objects together or to other information such as data stores. Connecting objects control the sequence of activities and the overall flow of the process. The types of connecting objects are Sequence Flows, Message Flows, and Associations. A Pool is a Swim lane object used to organize different activities; it is represented by a big rectangle, which contains multiple Flow Objects, Connecting Objects, and Artifacts.

III.6.2 HLA Standard

HLA is an architecture for interoperation and reuse of interacting simulations. We use the HLA Evolved Standard to develop a collaborative distributed DES. The HLA standard describes a set of services and rules for distributed simulations' implementation; its approach promotes interoperability and reusability. However, it does not promote any programming or modelling languages to describe the choreography between federates i.e. the way the federates will intercommunicate during the simulation execution (Zacharewicz et al. 2008). In HLA, the system is considered a federation, a federation is a collection of federates, federates are interconnected through a Run-Time Infrastructure (RTI). The RTI role is to ensure a smooth run of the simulation (Youssef et al. 2017; Zacharewicz 2006). Figure III.20 describes the overall architecture of an HLA simulation. In the example we worked on, Jaamsim models are federates that interconnect and connect to other external DESs.

The HLA standard defines:

- Ten architectural rules describing the responsibilities of the entire federation. One of the rules specifies that all data exchanges between federates must go through the RTI.
- A federate interface specification delineating the set of services provided by the RTI. These services are required to manage federates during simulation execution.
- A FOM that describes the shared objects and interactions used to exchange data.

HLA also supports optional services for time management, allowing the coordination of event exchanges between the existing federates. Time management is responsible for the mechanism of regulating the progression of each federate on the federation time axis. Each federate has a logical time. The RTI guarantees the time synchronization of the federates by consistently advancing the logical times of each federate. The logical time is equivalent to the simulation time in the classical literature of DES.

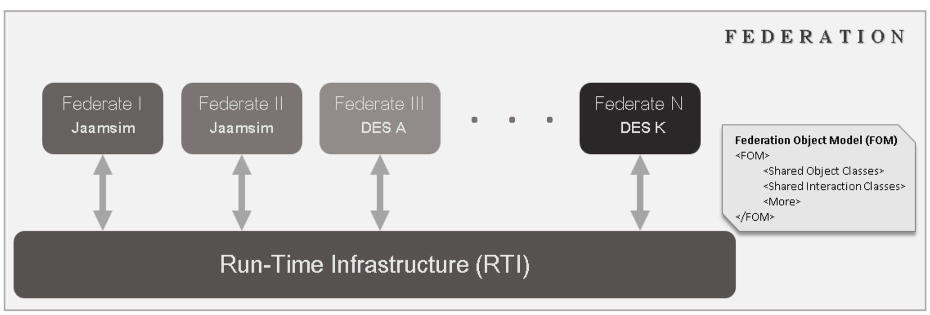


Figure III.20 Global Orchestration

The technical part of the methodology is also discussed in the present work. In this section, we will discuss the Java implementation of this methodology; the methods used to create, join, or destroy an HLA federation, the publish/subscribe mechanism, interactions/parameters communication, objects/attributes communication, time management and synchronization, DES (Jaamsim) configuration.

III.6.3 Material and methods

In this part, we will explain the development steps to connect all existing federates; the Master federate (external application), the actual case, the 5S, the Pull, the Ucell, the Cross training, the SMED, and the Poka Yoke federates. The aim of this DS is to determine how these scenarios react to changes in attributes and parameters, and compute the best behavior scenarios. These federates are linked via the RTI constituting a federation. These federation elements use a common FOM, an XML file that defines the objects/attributes and the interactions/parameters of the federation. We used the Java library of Pitch pRTI platform (Technologies) to develop the following part.

III.6.3.1 Federation related Services

First, a federation should be created. As per Figure III.21, when the Master platform starts the simulation, it creates the Federation by calling the RTI Ambassador; this HLA service creates the Federation using a unique Federation name and links it to its corresponding FOM XML file. The method in *RTIAmbassador* class used to create the federation execution is *RTIAmbassador.createFederationExecution("Federation Name", xmlFOMfile)*. After creating the federation, the Master, as a federate, joins the federation using the *RTIAmbassador.joinFederationExecution("Federate Name", "Federate Type", "Federation name to join")* method. Next, the Master launches the other DES scenarios that also join the created federation using the same method and parameters.

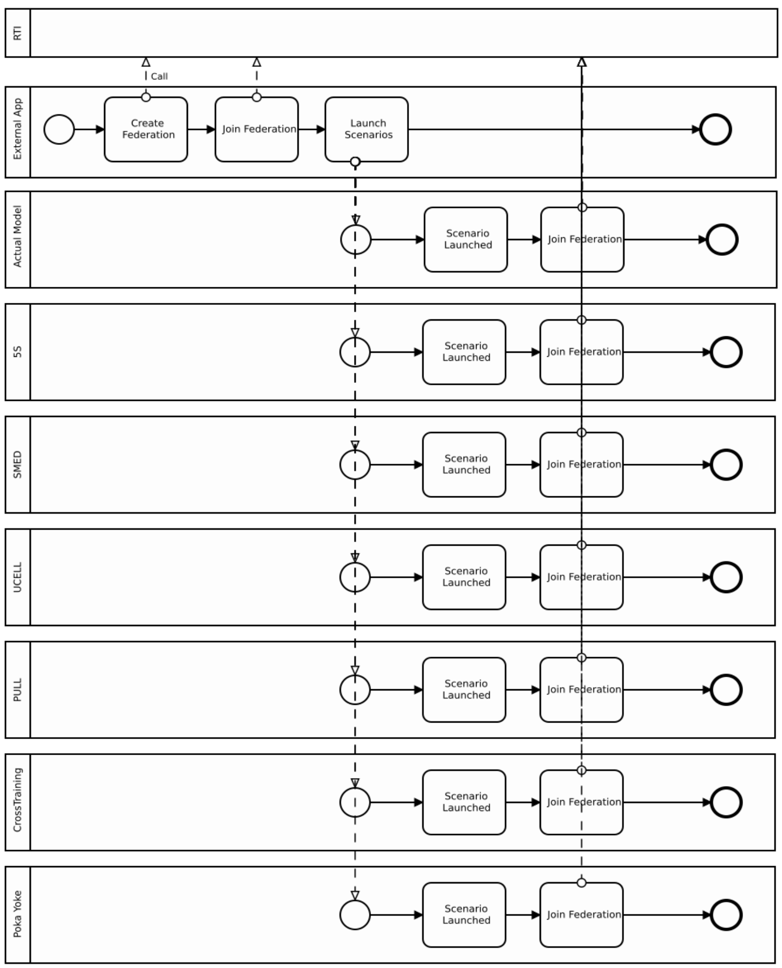


Figure III.21 Create/Join Federation

In the Figure below, we can see all federates joining the federation "HLA_Lean". We have 8 federates, the Master Federate that should be launched first in order to create the federation, the Actual model federate that represents the Actual state of the enterprise, and all the implemented

Lean tools that will be applied to the enterprise production system (5S, SMED, PULL, UCELL. POKA YOKE, and Cross Training).

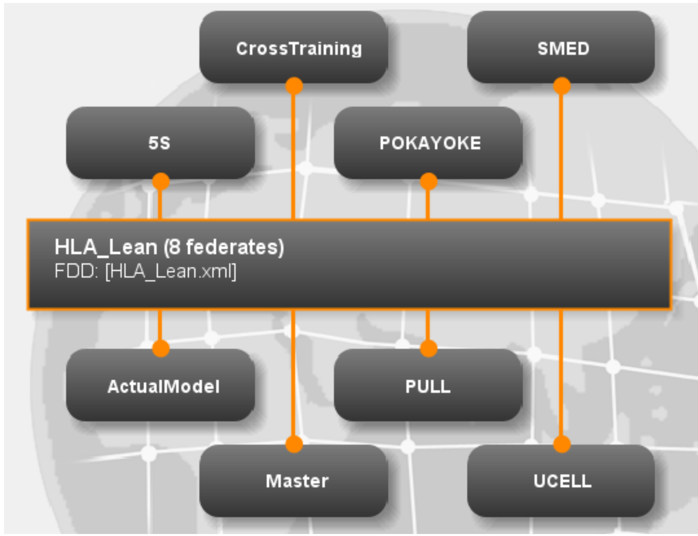


Figure III.22 Federates joining the Federation HLA_Lean

In the models we developed, the object class created is "Scenario". As per Figure III.23, the Scenario Object Class has the following attributes: Name, SimTime, RunDuration, Material Buffer, SKU, WIP, NumberOfDefects, NumberOfFinalProducts, LeadTime, SetupTime, ProcessingTime, etc. "Name" represents the Scenario Name, "SimTime" is the Simulation Time of the DES during the run time, "RunDuration" determines in years the duration of the simulation scenario, "MaterialBuffer" represents the number of raw materials waiting for the production process, etc. Each of these attributes has a specific type defined as per the figure below.

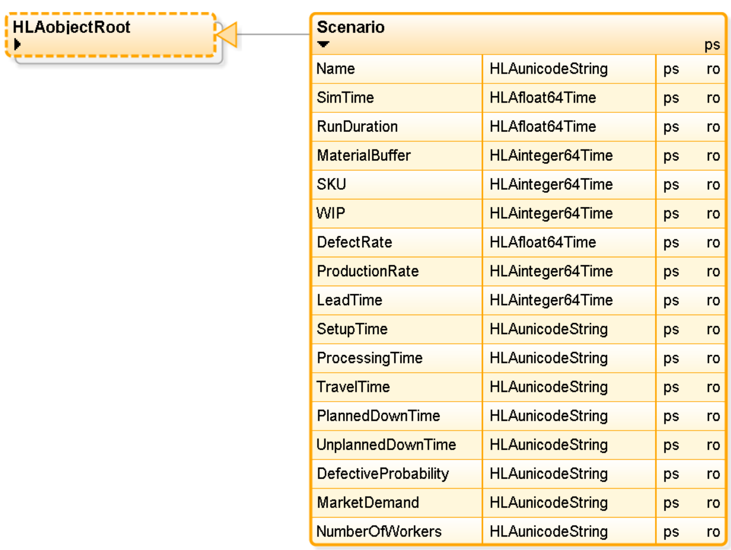


Figure III.23 Object Class of the FOM

We subsequently created nine interactions: Scenario Load, Scenario Loaded, Scenario Error, SMED Interaction, Poka Yoke Interation, Simulation Control, etc. See Figure III.24. For each interaction, we have one or more parameters listed as noted in Figure III.24. The objects/attributes and the interactions/parameters sharing mechanisms (Publish “p”, Subscribe “s”, Publish/Subscribe “ps”) are also listed in Figure III.24.

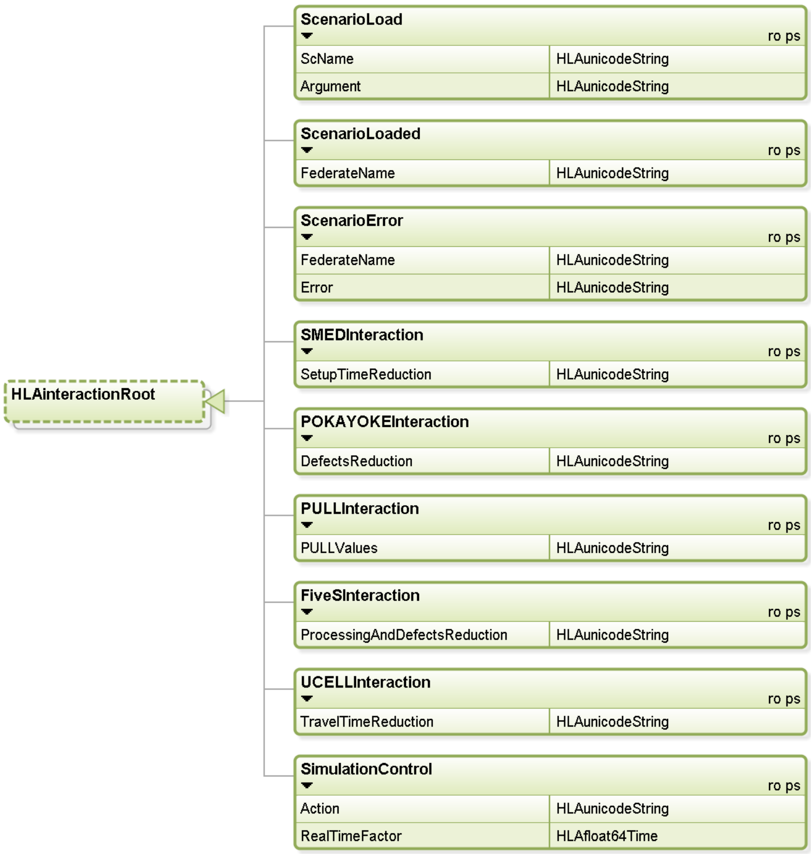


Figure III.24 Interaction Class of the FOM

III.6.3.2 Declaration Management Services

The BPMN model of Figure III.25 illustrates the steps used to select the Publish/Subscribe interests of the Object Classes. Each object should first get the handle for the actual object class in order to be published.

The method used for this service is *RTIAmbassador.getObjectClassHandle(“Object Class”)*. In our example, “Object Class” is “Scenario”. The next step involves creating an Attribute Handle Set using the method *create()* in the *AttributeHandleSetFactory* class. Next, one should get the Attribute Handle using *RTIAmbassador.getAttributeClassHandle(“Object Class Handle”, “Attribute”)* method. One of the attributes could be “Name” that exists in the Object Class “Scenario”. Next, the Attribute Handle Set should be added using the method *add()* in the *AttributeHandleSet* class.

The last step in the declaration part is to Publish/Subscribe the AttributeHandleSet of the Object Class using *RTIAmbassador.publishObjectClassAttributes(“Object Class Handle”, “Attribute Handle Set”)* and *RTIAmbassador.subscribeObjectClassAttributes(“Object Class Handle”,* *"Attribute Handle Set")* methods. After the publish Object Class Attribute, a callback from the RTI accesses the *startRegistrationForObjectClass("ObjectClass Handle")* method.

Registering the Publish/Subscribe for the interaction classes is more straightforward. First, one should get the Interaction Class Handle using the method:

RTIAmbassador.getInteractionClassHandle("Interaction Class") then get the Parameter Handle using *RTIAmbassador.getParameterHandle("Interaction Class Handle", "Parameter")* method.

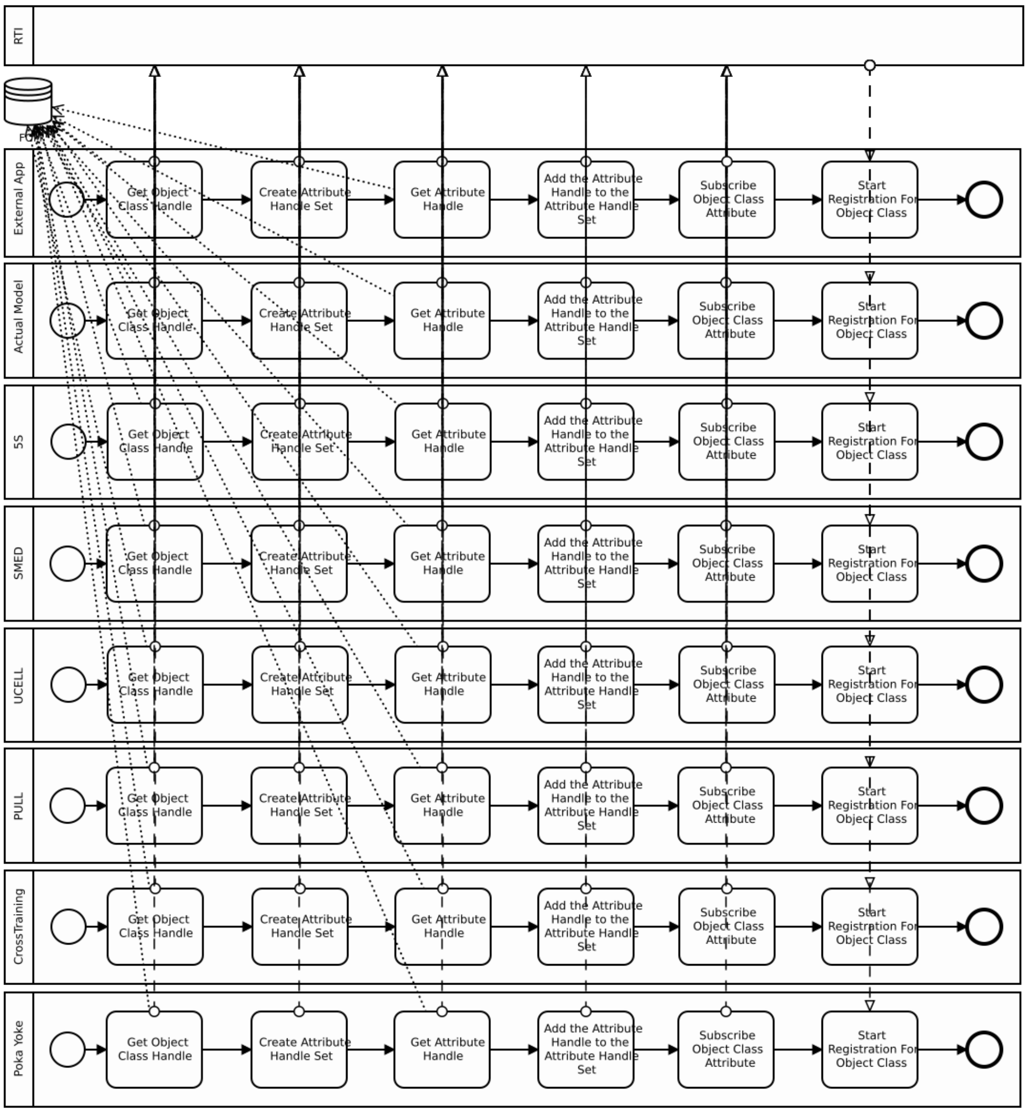


Figure III.25 Object/Attribute Declaration

III.6.3.3 Object Management

Figure III.27 shows the required services to register/discover object instances. *RTIAmbassador.registerObjectInstance("Object Class Handle", "the Object Name")* is required to register the object instance. After the registration process, a callback is sent to the other existing federates, accessing the method *discoverObjectInstance("Object Instance Handle", "Object*

Class Handle", "the Object Name"). *turnUpdatesOnForObectInstance()* callback method is accessed in the federate that registered the object instance.

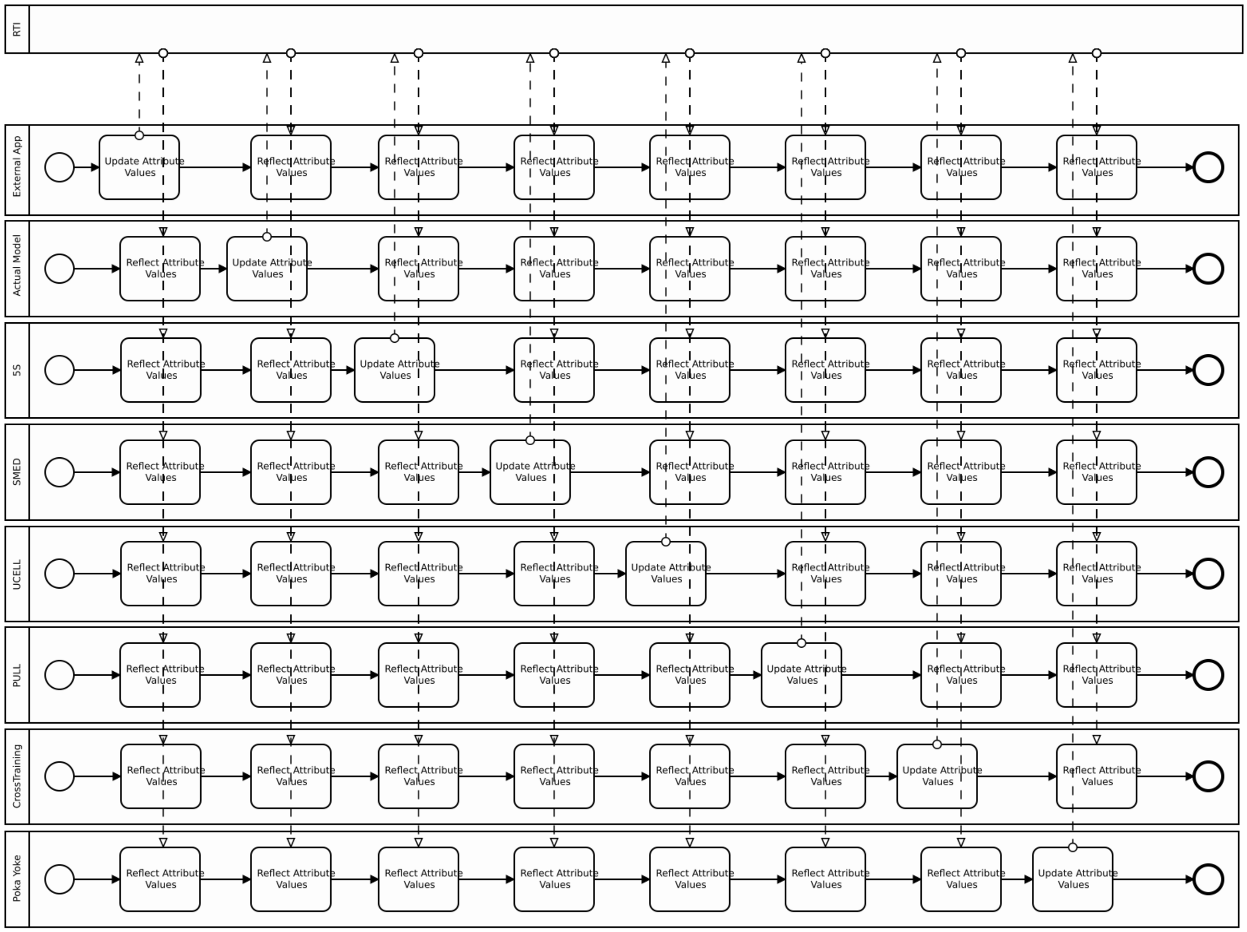


Figure III.26 Objects/Attributes update

The method *updateAttributeValues()* of the RTIAmbassador class is used to update the attributes related to the registered object instance. After the attribute update, *ReflectAttributeValues()* callback method is accessed in the other existing federates as per Figure III.26. As for the interactions, the same concept is used with the *sendInteraction()* method and *receiveInteraction()* callback method.

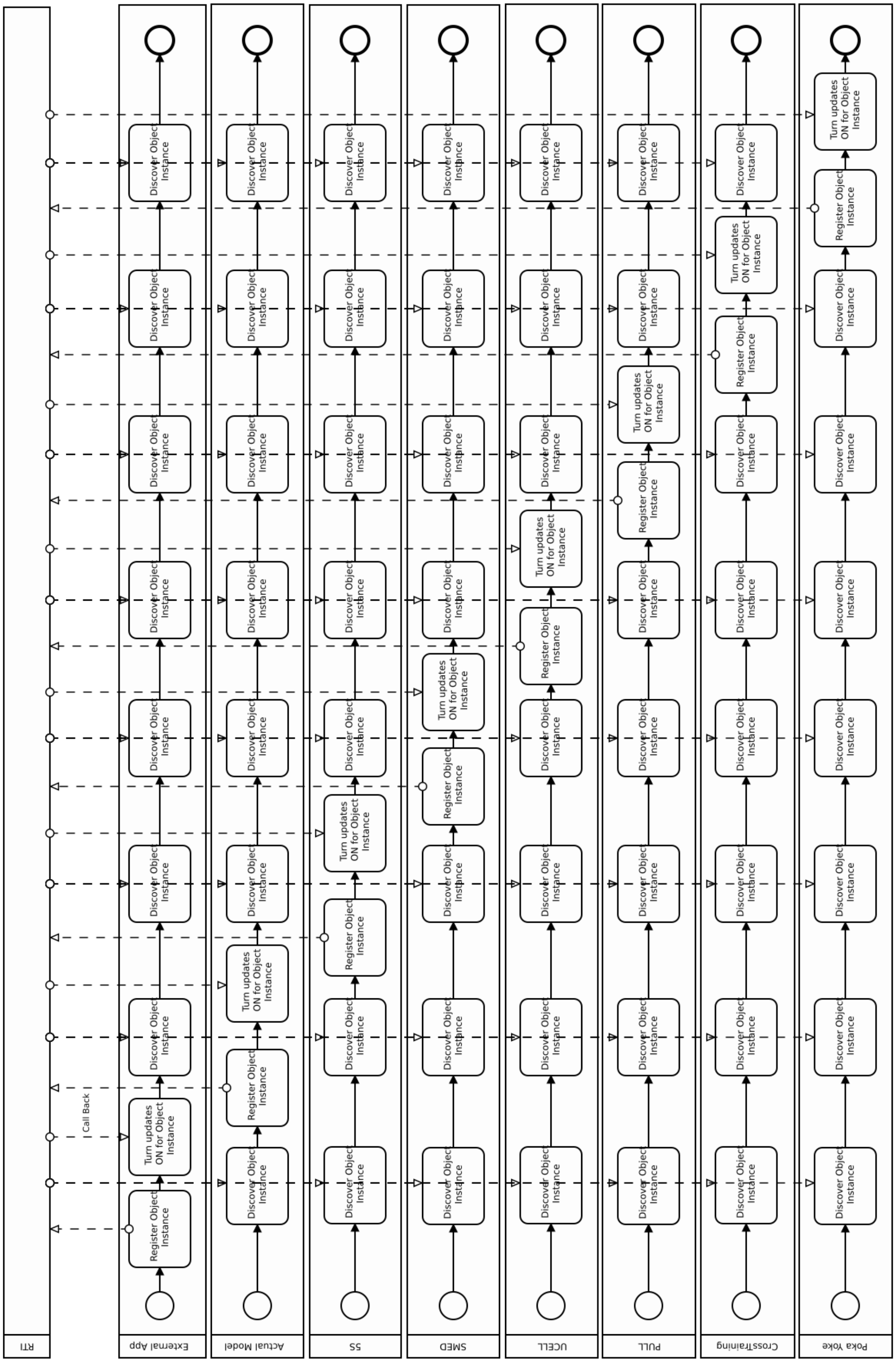


Figure III.27 Object Instance Registration

III.6.3.4 Time Management

In HLA, time management mechanisms are responsible for controlling the advancement of the federates along the time axis of the federation. The insight of “current time” might differ among the federates joining the federation. Time advances are coordinated to the services of Object Management so that federates will get their information in a precise and ordered manner. Federates can be assigned as regulating, constrained, or regulation/constrained. A regulating federate can control the logical time progress of constrained federates. By default, the RTI does not manage the time between federates. Thus, the time regulating and time constrained services are initially disabled.

To enable the time management services, a federate requests to be a time regulating federate using the method *EnableTimeRegulation(),* or to be time constrained using the method *EnableTimeConstrained()*. A federate could be time regulating/constrained at the same time. When these two methods are used, the Federate Ambassador calls back the *TimeRegulationEnabled()* and *TimeConstrainedEnabled()* methods. In our study, all DESs have the time regulating/constrained enabled in order to have them all running in parallel at almost the same simulation time. As per the external application, we have both time constrained and time regulation disabled.

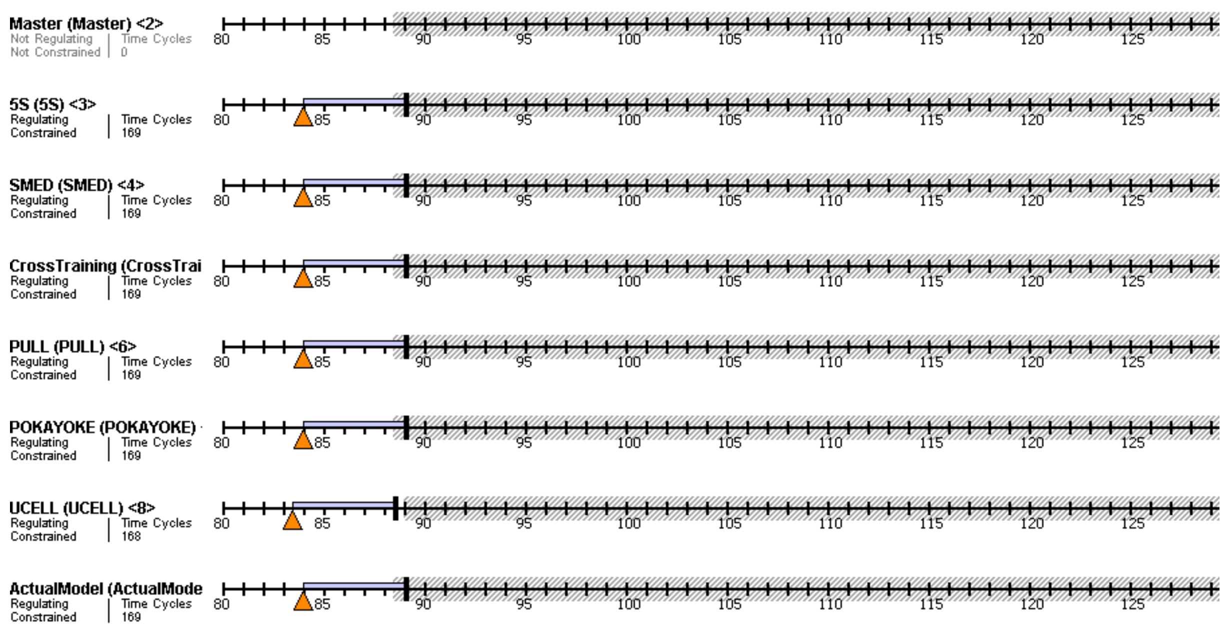


Figure III.28 Time Management of connected Federates

Time advancement can be requested through different time advancement services, event-based, time-step, and optimistic. We used the event-based time advancement service as we are working on Event-based federates. The goal of this service is to process all events in Time Stamp Order (TSO). In event-based federates, the method *nextEventRequest()* is invoked to request a logical time advancement.

Each federate declares a positive value for the Lookahead. The lookahead being the time delay that cannot be exceeded between simulations, it is essential to allow the processing of concurrent events having different time stamps. The larger the lookahead value, the longer it takes for messages to reach the other federates. With a zero lookahead, messages should reach the other federates instantly (Zacharewicz et al. 2006). These HLA services are used to avoid out of order messages delivery. The TSO sent by any federate should have its time stamp greater or equal than the current time of the federate plus its lookahead. When the federate asks for time advancement to send new events using the *nextEventRequest()* method, the RTI ensures that it will not deliver any message with a Time Stamped Order less than the lookahead time and the federate actual time combined.

In Figure III.28, the orange triangle represents the current simulation time of each federate. The purple bar represents the lookahead value of each federate. The Lookahead used for each federate during the co-simulation process is “5”. Our goal is to run the DESs in parallel in order to compare the results of each Lean tool when the same context change is imposed to these simulations. Therefore, each federate is getting its logical time from its DES simulator. The logical time of our federates is calculated in hours. A DES simulator might have different simulation time if it is running slower or faster than other simulators. Each of these simulators can automatically be paused then started multiple times to keep the parallelism between simulations. So, no federate’s logical time can exceed the minimum existing logical time plus its lookahead. As per Figure III.28, all simulations are running at almost the same logical time where the lookahead is never exceeded.

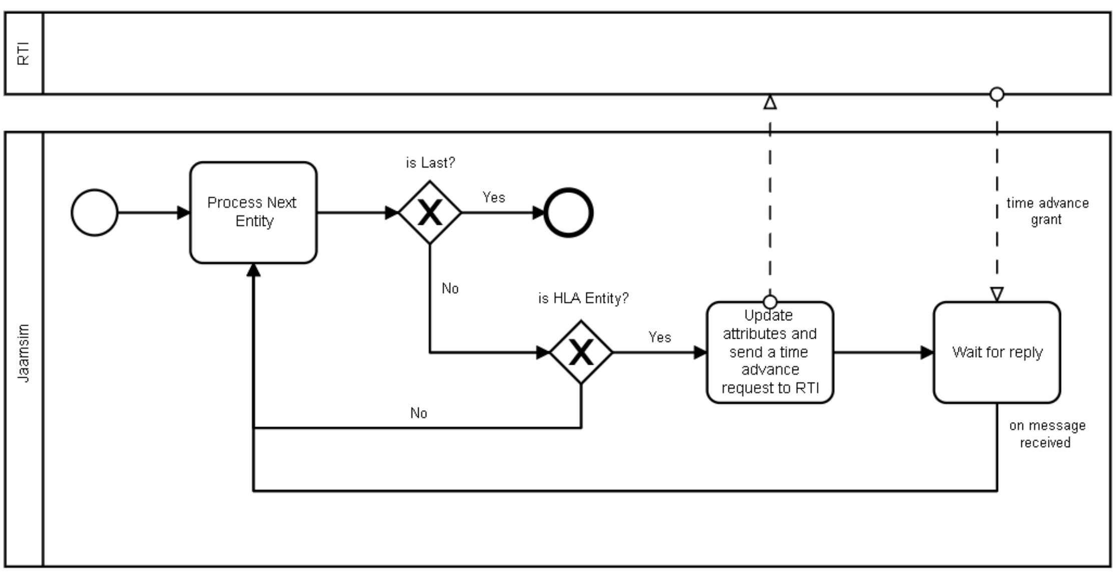


Figure III.29 Federate flowchart to connect to RTI

The DES Federates in this research are developed using Jaamsim. Jaamsim is not designed for communications to external systems and not fitted for DS; it is viewed as a black box simulator. As it is an open source software, we were able to change its Java code in order to make it HLA compatible.

In addition to the Federate java class created in the package *com.jaamsim.ui* of Jaamsim, we added a function to all Jaamsim objects. This function reads all attributes assigned to Jaamsim objects; if an entity has an attribute called "waitRTIOrder", it will be considered an HLA entity able to collaborate data and asks for time advancement, in this case, the federate updates the attributes and asks for time advancement.

After this step, it will wait for the RTI reply in order to advance in time and process the next entity. If no "waitRTIOrder" attribute exists, SimEntity will be directly sent to the next component to be processed. This process is repeated until the last SimEntity processed (end of the simulation run). This procedure is well explained in the flowchart of Figure III.29. The Java code of the federate and the changes to Jaamsim code are found in Appendix B.

In order to publish and synchronize all data needed to the RTI, we added a small configuration in each of the developed models. This configuration consists of an EntityGenerator that generates SimEntities and send them to an "Assign" object having the "waitRTIOrder" attribute along with all attributes that need to be published to the RTI. This Assign object will then direct all SimEntities to an EntitySink attribute. This way, it will be easier to implement any new model in our framework by just adding the above components and configurations. In the aforementioned Assign object, we initiated the following attributes in the AttributeDefinitionList, `{ waitRTIOrder 0 }{ WIP 0 }{ LeadTime 0 h }{ nbOfDefective 0 }{ nbOfFinalProducts 0 }{ prevNbOfDefective 0 }{ prevNbOfFinalProducts 0 }{ intSimTime 0 }{ productionThroughput 0 }{ defectRate 0 }`. The "waitRTIOrder" attribute gives the authority to the Assign object to pause the simulation, ask for time advancement, and exchange the data that are included in its AttributeDefinitionList. "WIP" calculates the current total WIP of the simulation. "LeadTime" is the time each SimEntity takes from the moment the order is received until the delivery of the finished product. "nbOdDefective" returns the number of cumulative defects. "nbOfFinalProducts" returns the number of cumulative final products produced during the simulation process. The production throughput and the defect rate are returned by the attributes "productionThroughput" and "defectRate". These attributes are calculated per production day. "prevNbOfFinalProducts" and "prevNbOfDefective" represents respectively the number of final products and the number of defective products of the previous day. "intSimTime" returns the floor value of the current simulation time. In Algorithm III.7, we show the pseudo-code for calculating the production throughput and the defect rate during the simulation process. As per this algorithm, intSimTime is calculated in hours; the production throughput and the defect rate are calculated each eight hours of the Simulation time (`intSimTime % 8`). The production throughput is the difference between the current and the previous number of final products produced. The defect rate is the percentage of, the difference between the current and the previous number of defects divided by the output of tested products. Both attributes are calculated per 8 working hours (day).

Algorithm III.7 Calculating the Production throughput and the Defect Rate

```
int nbOfFinalProducts = Number of final products at time t;
int nbOfDefective = Number of defective products at time t;
int prevNbOfFinalProducts = Number of final products at time (t - 8);
int prevNbOfDefective = Number of defective products at time (t - 8);
int productionThroughput;
```

```
float defectRate;
int intSimTime = floor([Simulation].SimTime);

if intSimTime % 8 == 0 then
 do {
   productionThroughput = nbOfFinalProducts - prevNbOfFinalProducts;
   defectRate = (nbOfDefective -
prevNbOfDefective)*100/productionThroughput;
   }
else
 do {
   productionThroughput = productionThroughput;
   defectRate = defectRate;
   }
```

III.6.4 Co-Simulation Platform Operating Instructions

III.6.4.1 Platform Home Interface

We used the JavaFx to develop the Co-Simulation digital platform and deliver an easy GUI desktop application to the user. When this platform is launched, the home interface appears as per Figure III.30 while some services run in the background to create the Federation Execution, gives it a Federation Name and links it to the FOM XML file in which all the Objects/Attributes and all the Interactions/Parameters exist. After the Federation creation process, the Master Federate itself joins the Federation and launches the other federates that, in turn, join the Federation. At this step, it is important to note that the simulations are not yet loaded, they just joined the federation execution as per Figure III.22. Alongside the Master Federate (external application), we have the 5S, SMED, POKA YOKE, Cross Training, PULL, and UCELL Federates connected to the Federation "HLA_Lean".

III.6.4.2 Load, Initialize, and Start Simulations

After launching the application, the user will choose one or more Lean tools to load. In Figure III.30, all exiting tools are selected (5S, SMED, Poka Yoke, Cross training, Pull, and Ucell). The user presses on the "LOAD" button to load these simulations. In the background process, the Master federate publishes the interaction "ScenarioLoad", all subscribed Federates to "ScenarioLoad" interaction will load their simulations respectively. The Federate that has its simulation loaded successfully without errors will publish the interaction "ScenarioLoaded" with its parameter "FederateName". The Federate that has an error while loading its simulation publishes the interaction "ScenarioError" with both parameters, the "FederateName" and the "Error". The Master federate, which is the only federate subscribed to "ScenarioLoaded" and "ScenarioError", gets all scenarios loaded with their federate names and all scenarios that encountered errors along with the federate name and the respective error encountered.

After loading the simulations, user can change the simulation speed factor, which is "1.0" by default. By pressing on the "+" sign button, the simulation will double the speed. User can also slow down the simulation speed by pressing on the "–" sign button. Moreover, user can write the simulation speed factor in the text field. In the background process of each simulation speed change, “SimulationControl” interactions and “RealTimeFactor” are published to all Jaamsim DESs.

Next process is the data input. The user fills in the yearly Market Demand, the Processing Time needed on each machine, the Setup Time needed to switch from a type of product to another, the travel time between workstations, the planned and unplanned down time interval for each workstation with the time needed to fix the downtime on each machine, the number of workers in each shop, the defect rate, and finally the Lean Tools configuration. After filling in the input data, the user presses on the “Send” button to publish the data input to all running federates. These data are published as attributes to all connected scenarios. We will implement the data input and configurations in Chapter IV to experiment the industrial context changes listed in Chapter II.

Figure III.30 Digital Platform Home Interface

After the data input, the user can run, in parallel, all loaded simulations by pressing on the green Start button. Simulation can also be paused or stopped. See Figure III.30.

During the simulation run and based on the publish/subscribe and the time management mechanisms of HLA; if the user changes any of the input data, all running simulations will receive this change. Moreover, the output data sent to the master external application will also change respectively. This will allow us to compare the results of each Lean tool based on any input or context change.

III.6.4.3 Output representation

The Master Federate gets the output data and time from each model and draw graphs representing real-time outputs. We used the Java library JFreeChart to draw the graphs. As per Figure III.31, each Lean tool is represented by a specific colored line. Based on the output results, the user can easily choose which Lean tool is reacting better if any input or context change is introduced. The output data used in this example are: lead time, daily defect rate, total WIP, and the production throughput per day.

The output data are the results calculated by the DESs and sent/updated as object attributes data. When any DES federate calls the *updateAttributeValues()* method to update its attributes, the method *reflectAttributeValues()* of the external application will return the updated values that will be, in turn, printed to real time graphs as per Figure III.31.

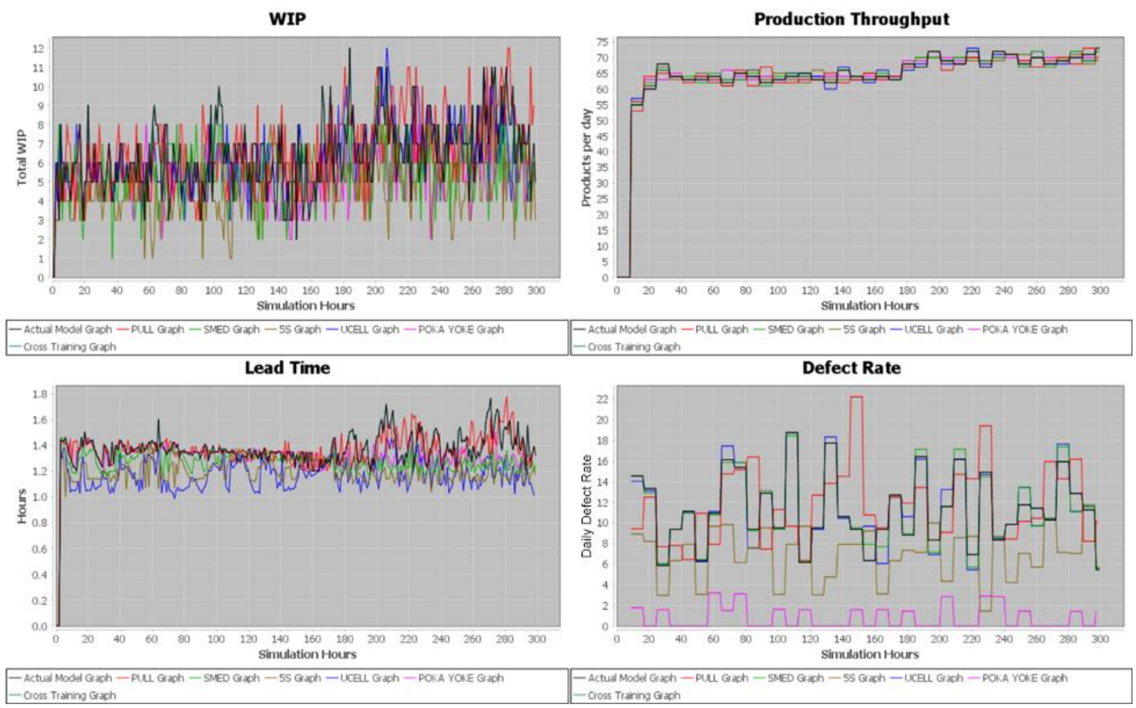


Figure III.31 Example of Output Results

As per Figure III.32, the user can select a specific area of the output result's graph to zoom into this area for more details.

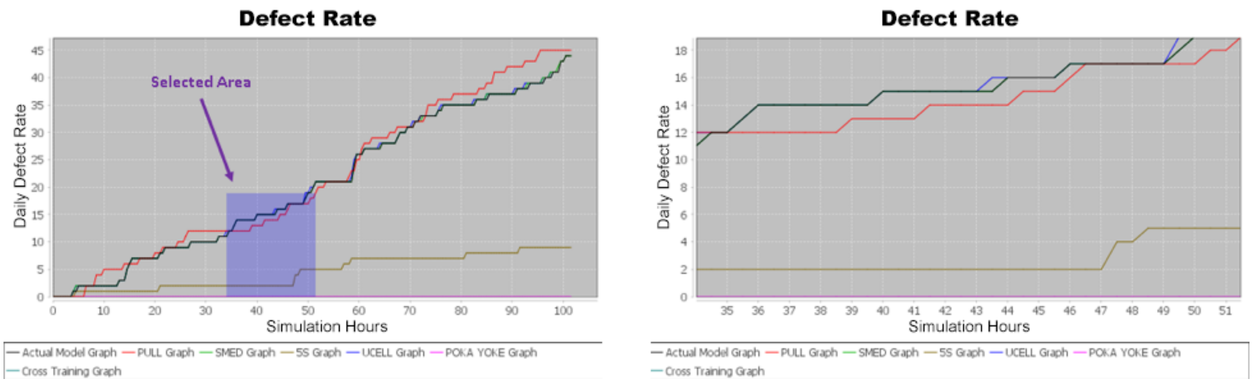


Figure III.32 Zoom in to a selected area

By right clicking on any of the graphs, the user will have multiple options such as copy, save, or print the current graph. See Figure III.33.

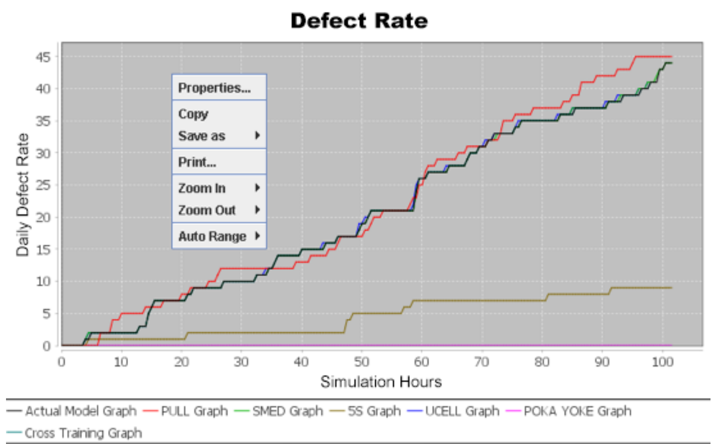


Figure III.33 Real Time Chart utilities

This Co-Simulation framework will be used as a decision-aided system by decision makers in industries to test and select the Lean operational tools that suit their organization and financial targets. Indeed, different contexts are modeled and introduced as input in the DS framework allowing parallel simulations to show the impact on WIP, lead time, production throughput, and defect rate. The results will be shown and analyzed in the next chapter. Furthermore, models and results will be cataloged and used as references for companies facing similar future situations.

III.7 Research Questions identified: RQ.5, RQ.6, RQ.7, RQ.8

- **RQ.5 What is the followed approach to have different DESs running in parallel?**

We have different DESs (components) running on a network of processors. Time synchronization between DESs is essential to have DES federates running in parallel. We used the HLA time management mechanism to control the DESs time advancement and have them all running simultaneously in parallel. Time management services should be first enabled in the connected federates. All DESs are configured as time regulation/constrained enabled so that, each of these federates can control the time advancement of the others in order to achieve a simulation parallelism between all connected federates. We used the event-based time advancement service as we are working on Event-based federates. This service's goal is to process all events in TSO.

We built the DES Federates using a Java-based open source simulator (JaamSim). We worked on the coding part of Jaamsim to make it able to connect, as a federate, to external systems and exchange/collaborate data with the other federates. We added a function to all the existing objects of Jaamsim that scans all attributes assigned to each object of the federate model. if an entity has an attribute called "waitRTIOrder", the federate updates the attributes and asks for time advancement. After this step, it will wait for the RTI reply in order to advance in time and process the next entity. If no "waitRTIOrder" attribute exists, SimEntity will be directly sent to the next component to be processed. This process is repeated until the last SimEntity processed. See the flowchart of Figure III.29.

- **RQ.6 What are the functionalities required to compare Lean tools in parallel through a Digital Platform?**

The goal of this project is to guide decision makers willing to implement Lean Manufacturing in their industries to choose the right Lean tools that suit their industry and economic contexts. For this purpose, we have developed a Co-Simulation framework that simulates a model, based on an aeronautic industry, in parallel and simultaneously with the same model having Lean tools applied on. We developed a graphical interface for the users to choose the Lean tools to load, test, and experiment. The user can also fill in the information related to the market demand, number of references, the setup time and processing time of each machine, the travel time between machines, the planned/unplanned down time of each machine, the defects rate, etc. Users are also able to start/pause/stop the simulations or change the simulation speed factor from this platform. Federates that represent the Lean tools can run on a network of processes, on different machines and different operating systems, which makes this framework powerful and independent from the computer resources.

By varying the economic contexts during the simulation run, the user can easily compare between all simulations' results and choose the ultimate tool/tools that fit its organization production and financial targets.

- **RQ.7 What are the challenges and barriers faced during a Co-Simulation framework development and implementation?**

In this project, we chose Jaamsim as the discrete event simulator because of its reliability, capability, and most essentially, because it is an open-source software that can be edited/modified. Jaamsim by default runs as a black box simulator; users should pause the simulation or wait for the whole simulation run to change an input data. In addition, Jaamsim is not developed to connect to external systems and environments.

One of the difficulties we faced in this project was to discover and understand the Java code of Jaamsim in order to add/change/configure new features/functionalities to make it an HLA compatible DES software able to interact, collaborate, and exchange data with external simulations. In our research, these functionalities were essential in order to run all the Lean tools in parallel, change their input data respectively and check the responsiveness of each tool based on the graphs and results.

The java development of the HLA Time management part, to ensure time synchronization between running DES federates, was also a challenging task, as we couldn't find a clear documentation on the use and development of the HLA time management mechanisms. Hence, a substantial effort was dedicated in developing and configuring these mechanisms in our Java code.

- **RQ.8 Why do we use a Case Study in this research?**

We validated the interest of our simulation platform thanks to the case study. Lean tools simulated as federates were generated from the chosen case study. We chose an aeronautic industry case, referred to Aerocomp in our study, which produces aeronautic fasteners. Aerocomp produces four product types of four different diameters. Product length can vary between a client and another. Therefore, we consider that the type of product changes only if its diameter changes. Aerocomp has four main production shops: the cutting shop, the treatment shop, the assembly shop, and the machining shop. In each shop, it exists one or more machine. One operator with particular skill is needed on each machine to efficiently complete the job and operate at capacity. Figure III.4 shows the production model of Aerocomp.

III.8 Conclusion

HLA standard does not propose nor precise any particular language to describe the behavioral process (choreography) of the federates inside the federation, i.e. group of federates, before the setup and implementation. This IEEE standard also does not point out any specific programing language or software use. In this chapter, we described the methodology followed to define the desired interconnections and data exchange between DESs while running simulations in parallel on a network of processors. The implementation steps are explained using BPMN and the Java library of pRTI. BPMN provides a standard straightforward notation easily readable by non-expert users while Java language describes the technical implementation part.

According to the literature review proposed in Chapter I, many industries are inefficiently implementing Lean tools in their organizations and are facing quality, management, financial, and other failures in their Lean implementations. This simulation framework aims to help manufacturing industries in choosing the right Lean Manufacturing tools that lead the implementation to success. As a concrete result, an actual model configured as Lean free scenario, as well as six scenarios having Lean tools applied on, were developed for this purpose and detailed in the current Chapter.

We will use this Co-Simulation framework in Chapter IV to test and experiment Lean tools behavior for some industrial context changes. Using this developed simulator, we will obtain relative hypothesis that contributes, helps, and conducts Lean tools implementations in the production domain.

CHAPTER IV. **Simulation on the Digital Platform:** Experiments, Results, and Analysis

In the previous chapter, we proposed an HLA-based co-simulation framework that was developed using Java programming, the pitch technologies library (Technologies), and Jaamim DES for the models' development (JaamSimDevelopmentTeam 2002). This constitutes a decision-aided framework for managers to help them in selecting best Lean tools that suit their organization production and financial targets. The main interesting outputs are the possibility for production manager to use the developed digital platform to lead observations simultaneously. This simultaneity based on co-simulation allows real time monitoring of the immediate impact of one Lean solution than another. The possibility offered by the developed platform is the parametrization by the decider regarding the actual data that are under his awareness. The platform developed represents a centralized area where the production line is represented with modular modules that the decider is able to update, to integrate, to move, to remove in order to represent his own industrial system. The market situation update, internal production line execution, and the disruptions of any kind can be represented, modelled, parameterized, and simulated.

The models we used in this framework are built based on the aeronautic case study of Figure III.4 defined in Chapter III. This case study is useful to represent, to model and helpful for carrying out the different simulations. The actual model along with six other models, representing the six Lean tools, are loaded and simulated in parallel. Inputs/Outputs are sent/received to an external developed Java application. The possibility given by the parallel simulation is a huge benefit for companies. Indeed, simulating each tool's impact in a sequential way, waiting results and storing them prevent the decision maker from the direct analysis and force him to cumulate the input data each time. The idea provided through the new digital built platform is to allow a common "input introduction" then data "parallel computation" in several identified and prefigured situations. A color code will be used to display different impacts of various tools running simultaneously in parallel. We will be using this co-simulation framework to experiment the effect of these Lean tools on the Manufacturing process, according to specific economic context factors. In this chapter, using the digital platform, results are presented, and analyses are conducted to determine the tendency of the suitable tools that led the company operating in a competitive environment to reach its potential objectives. Objectives have been retained and identified from the wide consulted literature. The main interest, from simulation point of view, would be to derive a panel of KPIs to analyze the effect of used tools in various contexts. Obviously, the KPIs ought to contribute to the achievement of the expected objectives targeted by the company. We start by introducing them in the following section.

IV.1 Key Performance Indicators (KPIs)

KPIs are the instruments used by decision makers and managers to analyze, understand, and verify whether the organization is on a prosperous path to achieve its fixed objectives or veering off the right track. KPIs' development should start with the company's strategies and the objectives the company is aiming to reach. It is important to design the KPIs based on your unique needs and circumstances (Marr 2012). In the current study and in order to track the four main objectives of the industry (quality, flexibility, cost, and reactivity), we chose four main KPIs. The developed platform is directly linked to the master federate responsible of receiving/sending output/input data from the other federates (Lean tools) and KPI results of all Lean tools federates are shown simultaneously in real-time during the simulation process. The KPIs used are listed as follows:

KP1: Lead-time is the time needed to provide the request to the customer; from the moment the order is received until the delivery of the finished product. Companies aim to continuously decrease the lead times and meet the clients' deadlines. Since the lead-time most likely varies for each order, we calculated the average lead-time which is the total number of lead times divided by the total number of orders placed. For all running simulations concerning the different Lean tools retained in this study, the value of lead-time is automatically updated on the lead-time graph. Each line color of the graph represents a specific Lean tool. Once the reader is familiar with the color code, the interpretation and the relative comparison of scenarios' graphical parallel results become interesting. It induces the decider to perceive the relative differences and ease the understanding of its own system undergoing different disruptions.

The designed KPI can be written as follows:

$\forall\ i = 1, \ldots, n\ ,\ t \in \{0, \ldots, m\}$, and whatever are the assigned values to $D_{XRF_{it}}$, $\widehat{D}_{XRF_{it}}$, $\overline{D_{C_u}}$, Q_{ti}, $D_{XRF_{ite}}$, $D_{XRF_{its}}$, we define the following elements:

Ω_{i0} *:* The simulation time where the product XRF_i started the production process,

Ω_{iF} *:* The simulation time of product XRF_i at the end of the production process,

$$KP1\text{: } leadtime_{XRF_i} = \Omega_{iF} - \Omega_{i0}\ \text{, lead-time of the product } XRF_i.$$

KP2: WIP stands for Work In Progress; it is a production term that describes the partially finished products awaiting completion. WIP is the cost of partially finished products in the production process; it is different from a finished product that is ready to be delivered to the customer. The production process has three stages as per Figure IV.1, raw materials, WIP, and finished products. The cost of production can gather the cost of storage of raw materials, cost of WIP inside the production lines, cost of final products in the warehouse, cost of daily human labor, cost of machines, etc. We consider that the cost of daily human labor and the cost of machines are identical in any scenario. They represent the common stable values of costs and the variation on scenarios will not have impact on this stable part, but it will influence the variable elements as WIP. The cost of raw materials' storage and the cost of finished products' storage are considered neglected because the supplier is considered as an available partner delivering in the required components on time. The finished products are considered to be delivered as soon as they are produced, so mainly the cost of WIP inside the production line can testimonies about the variation of organization and the variation of production costs. Analysts can track the WIP inventory of the company to guarantee that costs are allocated properly (Anastasia 2018). Consequently, it is very important for business managers to keep the WIP at minimal levels. The main goal for keeping the

WIP as low as possible, is to maintain the associated costs with in progress products in the machines and in the queue very low. In fact, WIP products require storages; they also take floor space in addition to several utilities to maintain them. Furthermore, warehouse may consume electricity and labor costs are sometimes needed to preserve the WIP products and keep them secure. Moreover, WIP in queue will hold up the production flow, which leads to slow production rates and more problems for not being able to meet the clients' deadlines.

Raw materials become WIP once they are placed into the manufacturing process. However, they remain unfinished goods if they are not yet gone through the entire manufacturing process.

Figure IV.1 Manufacturing stages

Regarding the case study and the steps through which the products are moving we can write the following:

$\forall\, M_p, p = 1 \dots U,\ \forall\, i = 1, \dots, n\ ,\ \ t \in \{0, \dots, m\},\ D_{XRF_{it}},\ \widehat{D}_{XRF_{it}},\ \overline{D_{C_u}},\ \ Q_{ti},\ D_{XRF_{ite}},\ D_{XRF_{its}}$, we define the following elements:

$queue_{M_{pt}}$: The queue of machine M_p at time t.

$\varepsilon_{M_{pt}} \in \{0,1\}$: Takes the value 1 if M_p is in a working state, otherwise it is equal to 0.

$$KP2: WIP_t = \sum_{p=1}^{U} \left(\varepsilon_{M_{pt}} + queue_{M_{pt}} \right), \text{ work in progress at time } t.$$

KP3: Pr*oduction throughput* refers to the quantity of products that can be produced/manufactured within a period of time. In our co-simulation framework, the production throughput is calculated per day (eight working hours) and is updated during the simulation process. We have clarified the production throughput calculation of all developed models in the previous chapter. Many factors can affect the production throughput, the complexity and nature of the product, machines' setup times, defective products, labors' skills, and so on.

Production throughput can be written as follows:

$\forall\, i = 1, \dots, n\ ,\ \ t \in \{0, \dots, m\},$

$\forall\, k = 0, \dots, 241$, the number of working days over one-year simulation horizon, each day being equivalent to 8 working hours.

$$KP3: Production\ throughput = \sum_{i=1}^{n} \sum_{t=8k+1}^{8k+8} Q_{ti}, \text{ Production throughput per day.}$$

KP4: Defect rate is the percentage of items or products that failed the quality tests. It is used to control and evaluate production, projects, services, programs, or processes. We use the following formula to determine the defect rate,

$$Defect\ rate = \frac{number\ of\ defects}{production\ output} \times 100$$

Defect rate is the best indicator for product quality (Westgard 2019). Companies always aim to reduce the defect rate in order to have better product quality. Furthermore, reducing defect rate improves on-time delivery and production throughput.

Defect Rate can be written as follows:

$\forall\ i = 1, \ldots, n\ ,\ \ t \in \{0, \ldots, m\}, \forall\ k = 0, \ldots, 241\ ,$

$\tau_{ti} \in \{0,1\}$: Takes the value 1 if the product XRF_i is defective, otherwise it is equal to 0.

$$KP4{:}\ Defect\ Rate = \frac{\sum_{i=1}^{n}\sum_{t=8k+1}^{8k+8}\tau_{ti}}{\sum_{i=1}^{n}\sum_{t=8k+1}^{8k+8}Q_{ti}} \times 100, \text{ Defect rate per day}$$

We will be using the aforementioned KPIs to demonstrate how each Lean tool can effectively help a company reaching its main objectives. As per Figure IV.2, we have four industrial contexts to test based on four main industrial objectives. The chosen industrial contexts (market fluctuation, diversification of demand, uncertainty of resources, and market typology) and objectives (quality, cost, flexibility, and reactivity) are explained in detail in Chapter II.

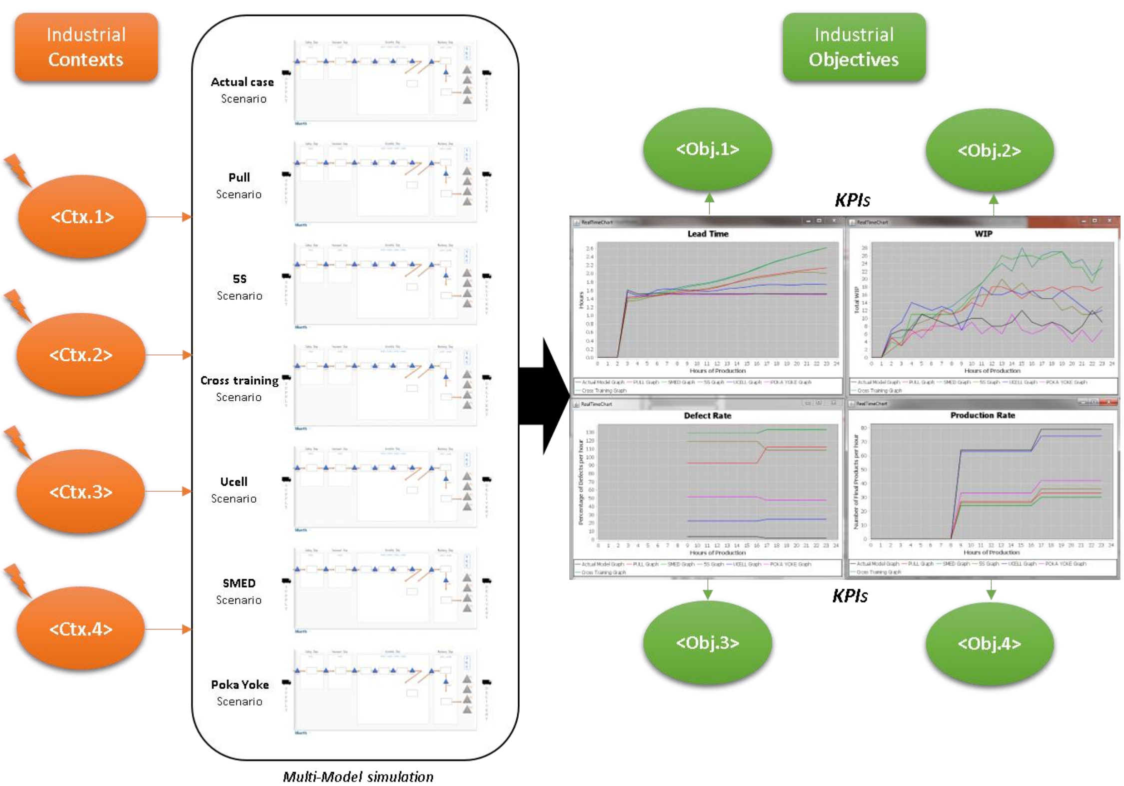


Figure IV.2 Overall concept

In this chapter, we will study each context by introducing the relative changes during the simulation run. The context changes are applied to all simulation models running simultaneously in parallel. All models will send their respective results, during the simulation run, to the master federate that in turn, sends the output results of all models to the respective KPI graphs (Lead-time, WIP, Defect Rate, and Production throughput). Based on the KPI results, we will be able to define the Lean techniques that help the company to reach its objectives. In the KPI graphs, each Lean tool/technique is represented by a line color (See Figure IV.2).

IV.2 ANOVA method for validation

Global Sensitivity Analysis is used to analyze a model by studying the impact of the variability of the input factors of the model on the output variable. Determining the inputs responsible for this variability using sensitivity indexes, the Sensitivity Analysis allows taking the necessary measures to reduce the variance of the output if it is synonymous with inaccuracy, or to lighten the model by fixing the inputs whose variability does not influence the output variable (Jacques 2011).

ANOVA was founded by Fisher (1918). The name Analysis of Variance was derived based on the approach in which the method uses the variance to determine the means whether they are equal or different.

The one-way ANOVA test is a statistical test that will make it possible to compare the averages of several samples and to decide on a difference or a similarity between these means. Variance analysis is used to study the behavior of a qualitative variable to be explained in terms of one or more categorical nominal variables.

Exploratory data analysis is an approach to the analysis of data sets to summarize their main characteristics, often with visual methods. A statistical model can be used or not, but most importantly the objective is to see what the data can tell us beyond modeling. Exploratory data analysis was promoted by John Tukey to encourage statisticians to explore the data, and eventually formulate hypotheses that could lead to collection and scenarios based on new data (Tukey 1977). Through the exploratory analysis of the data, we seek essentially to summarize the distribution of each variable (univariate approach) as well as the relationships between the variables (essentially bivariate approach), the characteristics of which could suggest a recoding or transformation of the measures (Tukey 1977).

Rather than modeling the data directly, we will first focus on describing them using numerical and graphical summaries. The idea of initially describing data using numerical and graphical summaries makes it possible to characterize the shape of a distribution and to identify any influential values. Exploratory data analysis uses a variety of techniques (mainly graphical) to:

- Maximize the understanding of a dataset,
- Extract important variables,
- Detect outliers and anomalies,
- Determine the optimal parameters of the factors,
- Detect errors,
- Check the hypotheses,
- Select appropriate models, and

- Determine relationships between the explanatory variables.

The main statistical concepts used:

In our study, we propose to perform the ANOVA test for validating the output results. Indeed, performing high number of simulations is likely not leading to discriminant results and the interpretation would be useless. Our aim in this statistical section is to compare the results obtained from the seven different models (actual scenario and six Lean tools applied) while simulating different industrial contexts (See Figure IV.3).

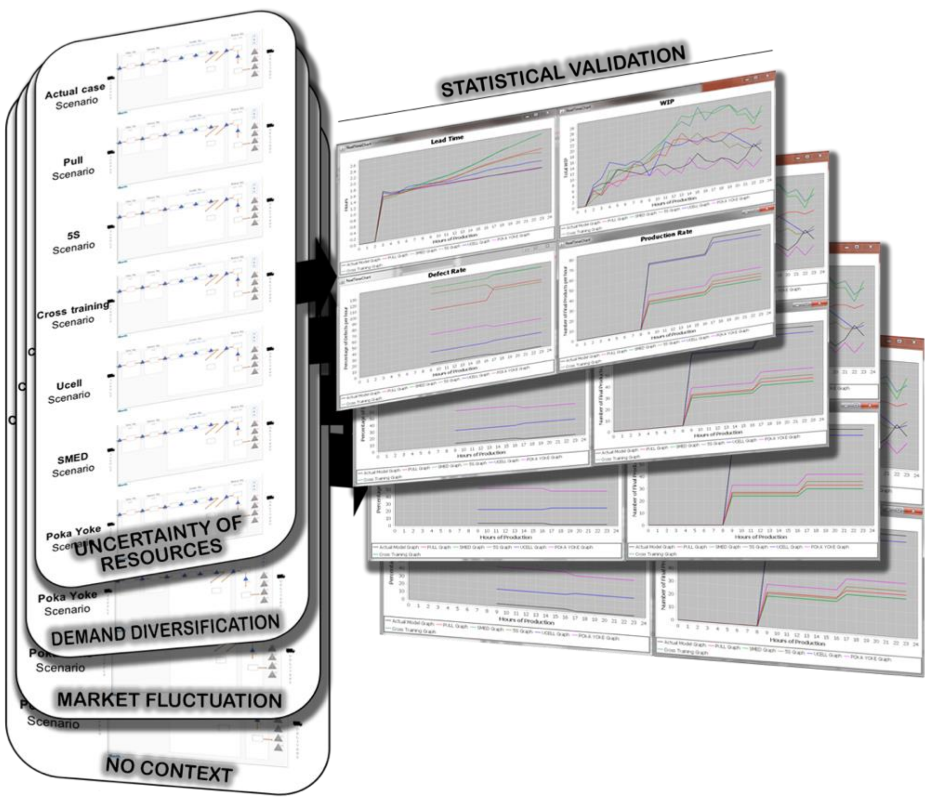


Figure IV.3 Possible configurations of ANOVA statistical analysis

The ANOVA test will allow us to analyse the variability of the results in order to argument their reliability. The mean of each Lean tool will be compared to others in order to perceive the variation of the results under each context run.

The null hypothesis "H_0" in statistics is defined as the case where the means of two samples obtained are remaining the same [$\mu_{Sample1} = \mu_{Sample2}$] with the specificity that samples are taken from equivalent population. In H_0 (null hypothesis), no variation is observed from different samples.

The alternative hypothesis "H_a" is defined as the situation where the means obtained from samples are different (< or >) creating variation and sensitivity in results [$\mu_{Sample1} \neq \mu_{Sample2}$].

Statisticians in general use a confidence level of 95% (Greenhalgh 1997; Shakespeare et al. 2001). In general, the higher the confidence level, the more certain you are that your results are accurate. Having more than 95% confidence level means that minimum one of the groups has a mean significant difference; this will justify our argumentation and simulation analysis. If we fall in the situation where the confidence level is less than 5%, it means the results are almost equivalent [$\mu._{Sample1} = \mu._{Sample2} = \mu._{Sample\ n}$].

p-value is an important statistical value representing the probability of having the null hypothesis correct. The p-value is directly related to the confidence level. When choosing the confidence level 95%, p-value less than 0.05 represents a rejection of the null hypothesis and adoption of the alternative hypothesis showing a discrimination among the obtained values. A p-value less than 0.05 (typically ≤ 0.05) is considered statistically significant.

A p-value higher than 0.05 (> 0.05) is not statistically significant and indicates weak evidence against the null hypothesis. This induces to fail-rejecting the null hypothesis and rejecting the alternative hypothesis.

In our simulations, we monitor the different means (μ) of the different KPIs (WIP, lead-time, Production throughput, and quality rate) undergone different contexts. We can hence justify the necessity to perform ANOVA analysis to sustain the reliability and validity of the results over the different simulated contexts.

Starting with the Neutral scenario, which is context-free (no disruptions are yet simulated). We can write 4 different initial null hypotheses (from statisticians' point of view) that we have to check significance level to continue the analysis.

WIP dependent variable:

H_0: $(\mu.WIP)_{Actual} = (\mu.WIP)_{Pull} = (\mu.WIP)_{5S} = (\mu.WIP)_{Cross} = (\mu.WIP)_{Ucell} = (\mu.WIP)_{SMED} = (\mu.WIP)_{PokaYoke}$

H_a: at least one of the 7 models' μ.WIP differs from the others

Leadtime dependent variable:

H_0: $(\mu.Leadtime)_{Actual} = (\mu.Leadtime)_{Pull} = (\mu.Leadtime)_{5S} = (\mu.Leadtime)_{Cross} = (\mu.Leadtime)_{Ucell} = (\mu.Leadtime)_{SMED} = (\mu.Leadtime)_{PokaYoke}$

H_a: at least one of the 7 models' μ.Leadtime differs from the others

Throughput dependent variable

H_0: $(\mu.Throughput)_{Actual} = (\mu.Throughput)_{Pull} = (\mu.Throughput)_{5S} = (\mu.Throughput)_{Cross} = (\mu.Throughput)_{Ucell} = (\mu.Throughput)_{SMED} = (\mu.Throughput)_{PokaYoke}$

H_a: at least one of the 7 models' μ.Throughput differs from the others

Defect rate dependent variable

H_0: $(\mu.\text{Defect})_{\text{Actual}} = (\mu.\text{Defect})_{\text{Pull}} = (\mu.\text{Defect})_{5S} = (\mu.\text{Defect})_{\text{Cross}} = (\mu.\text{Defect})_{\text{Ucell}} = (\mu.\text{Defect})_{\text{SMED}} = (\mu.\text{Defect})_{\text{PokaYoke}}$

H_a: at least one of the 7 models' µ.Defect differs from the others

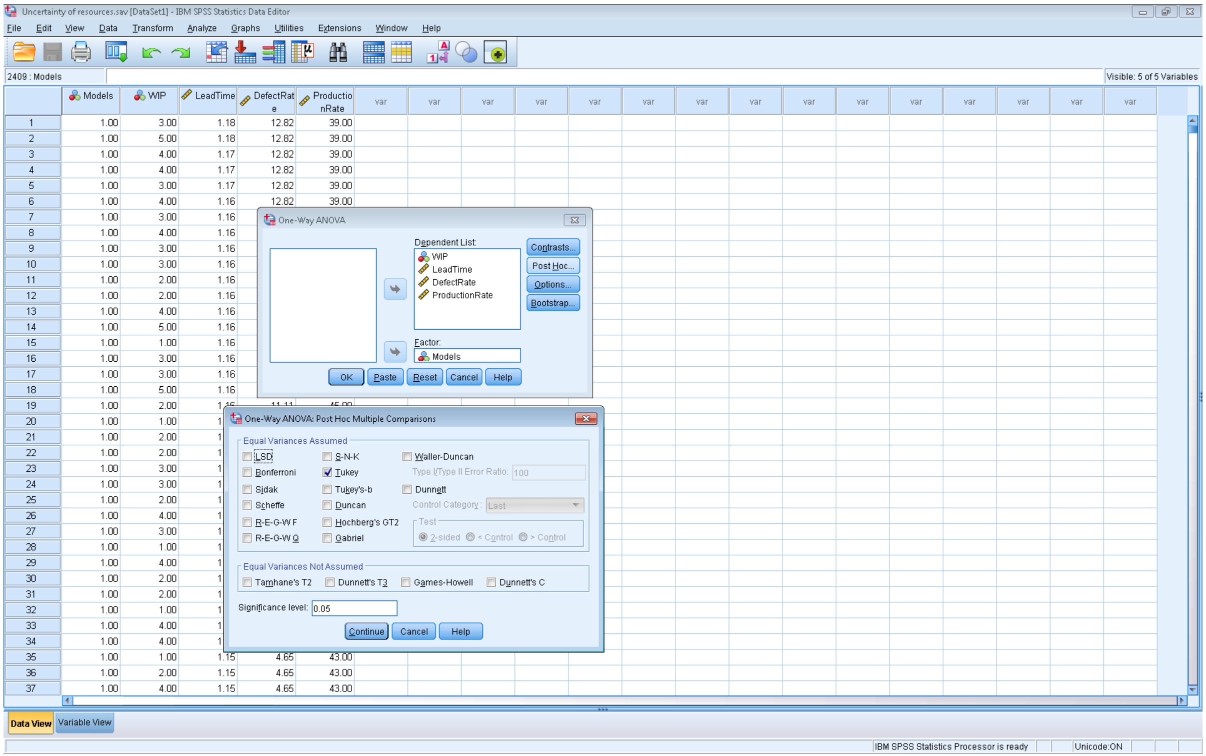


Figure IV.4 One-way ANOVA test on SPSS

When the simulations obtained define the results in significance area (p-value ≤ 0.05) it involves that at least one of the means of samples is significantly different. In this situation the only information that we can get is the variability of the obtained means but we cannot testimony yet about the comparative analysis to know which means are better $(\mu.\text{WIP})_{\text{Actual}}$, $(\mu.\text{WIP})_{\text{Pull}}$, or $(\mu.\text{WIP})_{5S}$, and so on. For this specific situation (p-value ≤ 0.05), we must carry with a *post hoc analysis* (using *Tukey* Kramer test) (Tukey 1977) to find out exactly which groups of means (µ) differ. The ANOVA test and Tukey Kramer post hoc analysis are performed using SPSSv26 software (See Figure IV.3).

In Figure IV.3, using SPSS software, we moved the output variables (WIP, LeadTime, DefectRate, and ProductionThrouput) into the "Dependent List" box and the independent variable (Models) into the "Factor" box. We selected the "Tukey" as the type of multiple comparison test.

SPSS output will appear with six sections:

- Descriptive section
- Test of Homogeneity of Variances
- ANOVA
- Multiple Comparisons

- Grade Point Average
- Graph

IV.3 Simulation Results and Analysis

In this part, we will use the developed framework to generate graphs and output results of different industrial contexts (defined in Chapter II). Each graph will show all operational Lean tools' behavior during each context induction. The tools are represented by colors in order to help the user follow each tool performance. Colors representing the tools are shown in Figure IV.5. The ANOVA test (one-way analysis of variance) will help us determine if there are any statistically significant differences between the means of the seven running models' results. The following contexts will be studied in the section below: market fluctuation, demand diversification, and uncertainty of resources. Several operational Lean tools are also analyzed, compared to each other, and to the actual scenario based on the simulation results and the statistical analysis. The below analyses are conducted to help companies and industries attain their targets and objectives.

Figure IV.5 Lean tools color lines

IV.3.1 Neutral scenario, no context or fluctuation

The first simulation run called neutral scenario is performed over 300h of production (simulation horizon time) with no context changes or severe fluctuations, considered as "Benchmark". All input data (setup time, processing time, travel time, order demand, Lean tools configurations, etc.) can be found in Appendix A.

Simulation graphs

The graphs below display the WIP, the lead-time, the production throughput, and the defect rate of the neutral scenario during a simulation run where no context changes took place. It can be qualified as silent scenario because of missing disruptions and no induction is operated. It is a neutral scenario because it reflects the initial situation of KPIs in neutral mode of simulation.

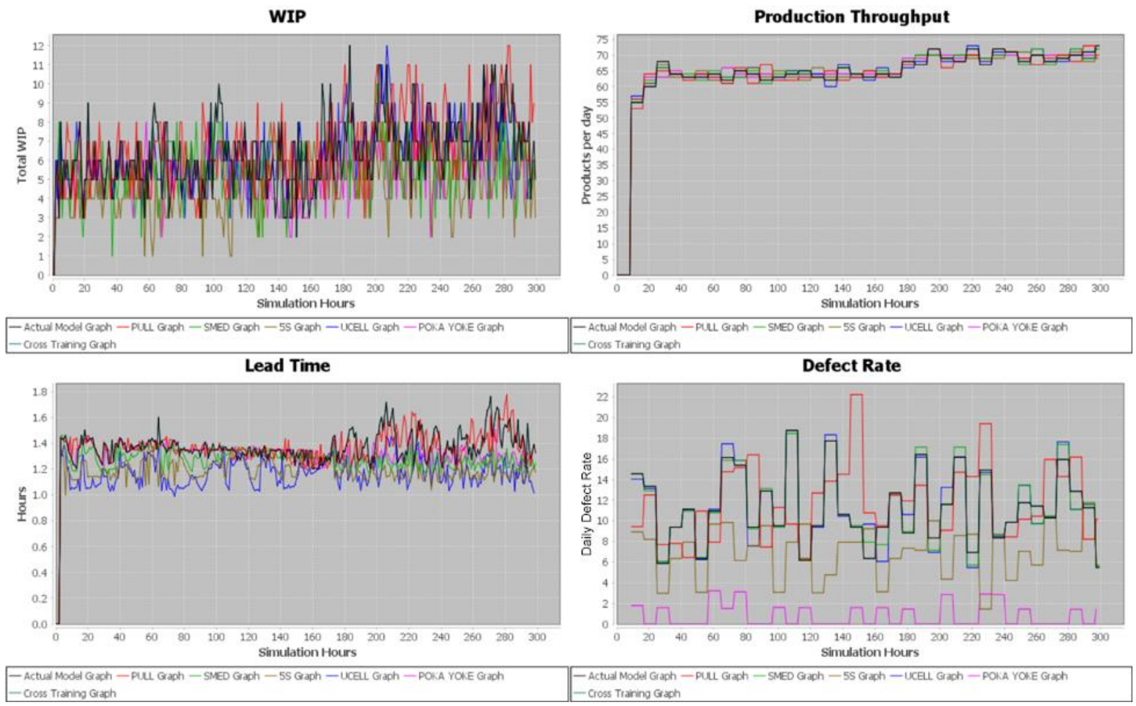


Figure IV.6 Simulation Results of the Neutral Scenario

ANOVA Test

While running the simulation, KPI results (WIP, lead-time, production throughput, and defect rate) were saved in a log file. These results were statistically analyzed using the ANOVA test.

Table IV.1 One-way ANOVA test of the default simulation run

		Sum of Squares[1]	df[2]	Mean Square[3]	F[4]	*p*-value[5]
WIP	Between Groups	946.144	6	157.691	57.159	< 0.0001
	Within Groups	5542.410	2009	2.759		
	Total	6488.554	2015			
Lead-time	Between Groups	13.409	6	2.235	305.393	< 0.0001
	Within Groups	14.701	2009	0.007		
	Total	28.110	2015			
Defect rate	Between Groups	28835.350	6	4805.892	477.226	< 0.0001
	Within Groups	20231.568	2009	10.070		
	Total	49066.918	2015			
Production rate	Between Groups	30.603	6	5.101	0.372	0.897
	Within Groups	27531.556	2009	13.704		
	Total	27562.159	2015			

1. Sum of squares is the squared difference between the values and their means (deviation).
2. DF means the degrees of freedom in the source.
3. The mean square is the sum of squares divided by the degrees of freedom (df)
4. F is the ratio of the mean square between groups and within groups
5. p-value is the probability of having the null hypothesis correct

In the ANOVA test of Table IV.1, we have a significant difference between the WIP means (μ.WIP) with $p<0.0001$, the lead-time means (μ.Leadtime) with $p<0.0001$, and the defect rate means (μ.Defect) with $p<0.0001$ of all running models. With $p<0.05$ for each of the 3 KPIs mentioned above, a closure look at the different models will take place in order to determine those that have contributed to generating the statistical difference using the Tukey-Kramer post hoc analysis. As for the production throughput (Last row of Table IV.1), the difference between the results (µ.Throughput) using different Lean tools is not statistically significant (p>0.05). Tukey-Kramer multiple comparison analysis is thus not justified.

Tukey-Kramer post hoc analysis

p-value ≤ 0.05

p-value > 0.05

Same Model

WIP

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.003	0.000	0.000	0.000
Actual model	0.000		1.000	0.000	0.934	0.000	1.000
Cross training	0.000	1.000		0.000	0.934	0.000	1.000
Poka Yoke	0.003	0.000	0.000		0.000	0.334	0.000
Pull	0.000	0.934	0.934	0.000		0.000	0.786
SMED	0.000	0.000	0.000	0.334	0.000		0.000
Ucell	0.000	1.000	1.000	0.000	0.786	0.000	

Lead-time

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.000	0.000	0.135
Actual model	0.000		1.000	0.000	1.000	0.000	0.000
Cross training	0.000	1.000		0.000	1.000	0.000	0.000
Poka Yoke	0.000	0.000	0.000		0.000	0.000	0.000
Pull	0.000	1.000	1.000	0.000		0.000	0.000
SMED	0.000	0.000	0.000	0.000	0.000		0.000
Ucell	0.135	0.000	0.000	0.000	0.000	0.000	

Defect rate

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.000	0.000	0.000
Actual model	0.000		1.000	0.000	0.835	1.000	1.000
Cross training	0.000	1.000		0.000	0.835	1.000	1.000
Poka Yoke	0.000	0.000	0.000		0.000	0.000	0.000
Pull	0.000	0.835	0.835	0.000		0.771	0.838
SMED	0.000	1.000	1.000	0.000	0.771		1.000
Ucell	0.000	1.000	1.000	0.000	0.838	1.000	

Figure IV.7 Multiple comparisons (Tukey) for the Neutral Scenario

In Figure IV.7, we show the multiple comparison of all models of the neutral scenario for the WIP, lead-time, and defect rate KPIs.

WIP: For the WIP dependent variable, the mean of 5S model $(\mu.\text{WIP})_{5S}$ is significantly different from that of all remaining models ($(\mu.\text{WIP})_{Actual}$, $(\mu.\text{WIP})_{PokaYoke}$, $(\mu.\text{WIP})_{SMED}$, $(\mu.\text{WIP})_{Ucell}$, $(\mu.\text{WIP})_{CrossTraining}$, and $(\mu.\text{WIP})_{Pull}$) with p <= 0.003. The Actual model, cross training, pull, and Ucell do not have a significant mean difference among each other (p>0.05). Though, the mean of the actual model $(\mu.\text{WIP})_{Actual}$ is significantly different from $(\mu.\text{WIP})_{PokaYoke}$, $(\mu.\text{WIP})_{SMED}$, and $(\mu.\text{WIP})_{5S}$ with p<0.0001.

$(\mu.\text{WIP})_{PokaYoke}$ and $(\mu.\text{WIP})_{SMED}$ have a significant difference from the mean of all remaining models (p<0.003). However, among each other, no significant difference exists (p=0.334). For the moment it is impossible to claim that one tool is better than another, the generated results are almost similar.

Lead-time: For the lead-time dependent variable, the mean of *5S, Poka Yoke, and SMED* models is significantly different from that of all remaining models and among each other (p <= 0.0001). $(\mu.\text{Leadtime})_{Actual}$ does not have a significant difference compared to $(\mu.\ \text{Leadtime})_{CrossTraining}$ and $(\mu.\ \text{Leadtime})_{Pull}$ having p=1. It equals to say that actual model (Lean free model) behaves the same as a model where cross training has been implemented or a pull system has been established.

$(\mu.\ \text{Leadtime})_{Ucell}$ is significantly different from the mean of all other models (p<0.0001) except $(\mu.\ \text{Leadtime})_{5S}$ (p=0.135).

Defect rate: It is clear in the defect rate's Tukey post hoc analysis that $(\mu.\ \text{Defect})_{PokaYoke}$ and $(\mu.\ \text{Defect})_{5S}$ have significant differences in comparison to other models (p<0.0001). All other models do not have a significant mean difference among each other. They can be considered as similar in generated results without any interesting potential improvements.

The test of Turkey revealed a significant variation in results regarding the 3 aforementioned KPIs among the 4 KPIs chosen to lead the study. Figure IV.8 highlights the variation of means to better perceive the relevancy of some tools compared to others.

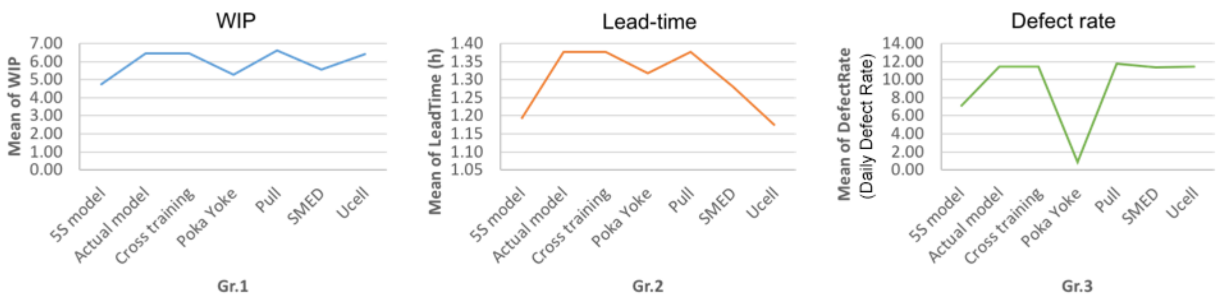


Figure IV.8 Group means (µ. WIP), (µ. Leadtime), and (µ. Defect) of the Neutral Simulation Scenario

Figure IV.8 has three graphs to represent the means of the three different output results (WIP, lead-time, and defect rate) of the neutral scenario. Each dot represents a sample mean. In this scenario, cross training, pull, and Ucell models have almost the same WIP value (~6.5) of the actual model. SMED, Poka yoke, and 5S have smaller WIP values (~ 5.6, 5.2, and 4.8 respectively). As for the lead-time mean result, cross training and pull kept almost the same value (~1.37h) of the actual model's lead-time mean. However, Poka Yoke, 5S, and SMED decreases this value to ~ 1.32h, 1.19h, and 1.17h respectively. The defect rate of the cross training, pull, SMED, and Ucell models remained almost at the same level (~11% of daily defect rate) of the actual model's rate. 5S and Poka Yoke models decreased this rate to ~7% and 1% respectively.

Discussions and interpretation of the neutral scenario

We can realize from Figure IV.6 that the order demand during the current simulation is stable and the company is effortlessly producing the required orders. This also explains why the actual model along with all existing models have almost to the same *production throughput* average. In the neutral scenario, $(\mu.\text{Throughput})_{\text{Actual}} \sim (\mu.\text{Throughput})_{5S} \sim (\mu.\text{Throughput})_{\text{PokaYoke}} \sim (\mu.\text{Throughput})_{\text{SMED}} \sim (\mu.\text{Throughput})_{\text{Ucell}} \sim (\mu.\text{Throughput})_{\text{CrossTraining}} \sim (\mu.\text{Throughput})_{\text{Pull}}$.

The multiple comparison study revealed differences in results of the WIP KPI over the simulation period. The WIP indicator is targeting to be the lowest possible. We can argue that 5S, Poka Yoke, and SMED that are significantly different from the other remaining tools in terms of WIP, are revealed to be interesting in WIP reduction. Indeed, Poka yoke and 5S are in charge of reducing the defective products by quick tools identification and avoid error risks generating defective products. So, 5S and Poka Yoke decrease the present work in progress in the whole system. The SMED that is in charge of reducing the setup times reduces the WIP by decreasing the existence of different references in the workstations. SMED accelerates the treatment of input products and reduces the queuing in front of workstations.

We can realize that pull in this case did not affect the WIP as no overcapacity products exists during the production process and no disturbance in the order book. In addition, we configured the pull model to send a signal to the upstream machine to stop sending goods in process when each WIP exceeds 3. As there is no demand increase the pull model remains non-reactive in neutral scenario justifying the non-impact on WIP.

Referring to Figure IV.8, Ucell and 5S models significantly improved the lead-time KPI even without any disturbance in scenario. Pull and cross training do not have a lead-time significant difference compared to the actual model. In neutral scenario we understand that the importance of these tools is not linked to the acceleration of the production process but is linked to the disturbance constraint. As there is not demand disturbance means the demand is stable, known and smoothed along the year: even if the company is training staff to become cross trained or pull system is implemented, there is no requirement to that in calm context. The results remain the same. We can perceive a little improvement of lead-times (1.35h) in Poka yoke because we expect that mistake proofing system can sustain the production flow system by avoiding losing time in reworking, improving hence the lead-time comparing to actual model. In this simulation scenario, we are unable to identify the Lean tools that help improve the companies' flexibility and reliability objectives, since the simulation environment is stable and no remarkable variations exist. Indeed, when the context is in reference situation without external nor internal disruptions, the production throughput remains almost equivalent.

For the defect rate KPI, the Poka Yoke and 5S models significantly minimize the existing daily defect rate by almost 40% to 90% respectively compared to the actual model and all other tools. Thus, in the absence of any context or fluctuation, out of the seven tested models, the Poka Yoke and 5S tools best performed leading to better production quality. This is an interesting observation that is independent from the context.

<Finding.1> **5S and Poka Yoke** are linked to the "quality" objective and "WIP reduction". They are required to improve production quality independently from the demand variation or internal equipment disturbances. Moreover, whatever the demand is, both tools will reduce defects; thus, reducing the cost associated with materials, rejects, rework and rescheduling, etc. We can outline the necessity of considering both Lean techniques as ***pre-requisites*** independently from the context where it is evolving. This finding is important and interesting regarding our initial hypothesis claiming relevancy of tools to context. *Lean techniques such as 5S, Poka Yoke can be relevant in any context without undergoing yet disruptions.* They still react as best in class tools.

<Finding.2> As per the neutral scenario study, when no contexts or fluctuation arise, **Ucell and 5S** are found to be good tools to use. Both tools decrease the lead-time. The indicator KPI targets the "reactivity" sustaining the delivery. Indeed, Ucell helps to ensure sequence in production flow evolution by doing the operation (n+1) as soon as (n) is ended. 5S is a quick identification of the required tools, components in workstations ensuring accelerated processing time on workstation. Both tools are essential to accelerate production flow.

IV.3.2 Market demand fluctuation <ctx.1>

In this section, we start considering the core hypothesis of our research questions. How the context may influence the choice of the relevant Lean tools? The simulation will be carried out over 500h of production to test the market fluctuation context. The simulation horizon time will differ between studied contexts, it is the time during which the simulation took place. We are trying to keep the system under fluctuation, so basically this time will differ on the number of fluctuations and on the responses of the simulated tools. In this context, we will test the industrial system behavior undergoing rise or fall of the market demand ($\widehat{D}_{XRF_{i(t)}}$). The demand fluctuation may be slight or dramatic depending on the market.

Table IV.2 Market demand fluctuations over the simulation horizon (H=500h)

Production hour (t)	**Market demand ($\widehat{\boldsymbol{D}}_{\boldsymbol{XRF}_{i(t)}}$)**
0 h	Initial state
50 h	15% of market demand *increase*
100 h	Initial state
130 h	30% of market demand *increase*
190 h	Initial state
400 h	15% of market demand *decrease*
450 h	Initial state

As per Table IV.2, the simulation scenario starts at time 0, initial market demand and input data can be found in Appendix A. After 50h of production, we induced the first fluctuation, which is a 15% increase on the market demand. At simulation time 100h, we returned the market demand back to its initial state. At 130h, we made another fluctuation, a 30% of demand increase. At time 190h, we put back the market demand to its initial state. After testing the market demand increase, we tested the demand decrease of 15% at simulation time 400h then we returned the demand to its initial state at time 450h. Thanks to the contribution of HLA, federates, and co-simulation, the built digital platform is able to include the arrival of new data (market fluctuation data) in real-time execution and updates automatically the system for displaying the KPI results simultaneously. In the addition to the market demand fields, we developed a field in the platform responsible of increasing/decreasing the market demand by a certain percentage specified by the user. This will help us test the increase and decrease of demands by just changing the percentage field on the platform. In the following section we suggest to gradually perform the simulations, the observations and finally we comment and deduce the important insights.

Simulation graphs

Figure IV.9 displays the WIP, the lead-time, the production throughput, and the defect rate graphs of the Market fluctuation scenario during a simulation run over 500h of production where multiple fluctuations took place. Each black dashed line of the figure below represents a certain fluctuation at a specific simulation time. Fluctuations are as per Table IV.2.

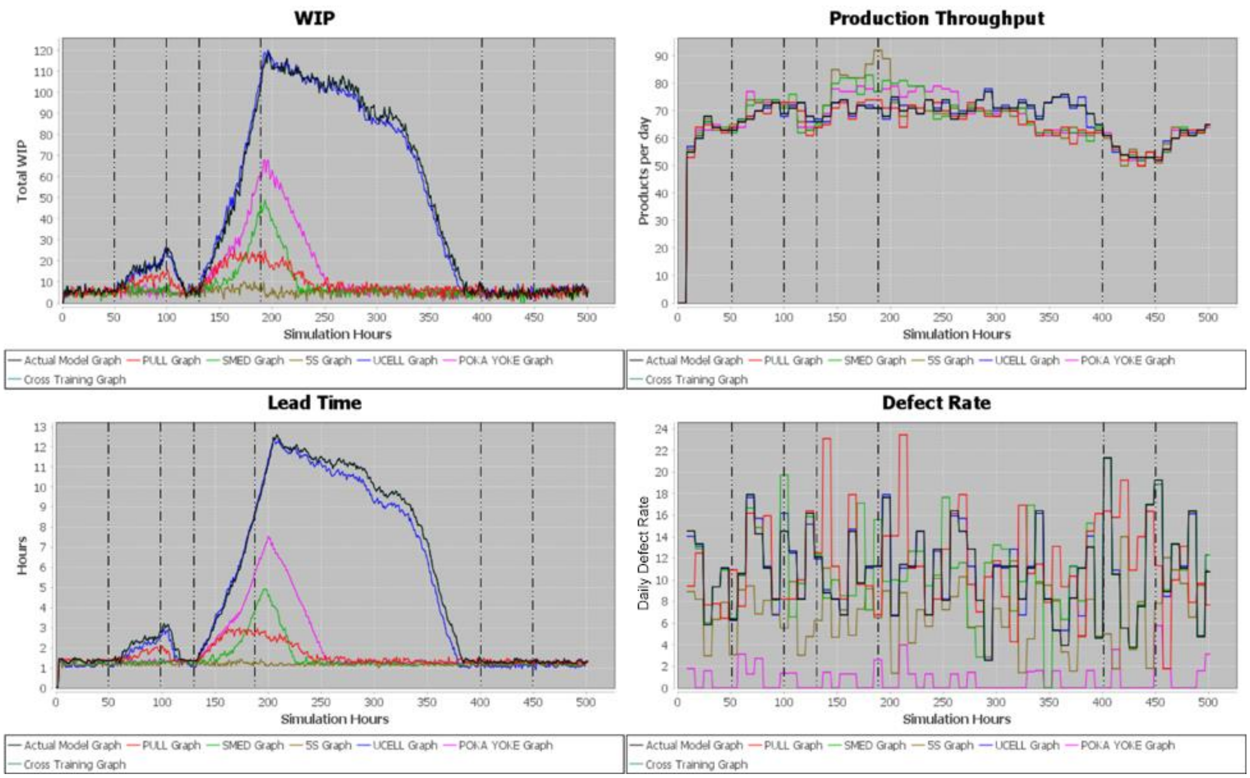


Figure IV.9 Simulation of the Market Fluctuation context

ANOVA Test

In the ANOVA test of Table IV.3, unlike the test of the first simulation scenario, all output results' means are *significantly differen*t.

Table IV.3 One-way ANOVA test of the market fluctuation context

		Sum of Squares	df	Mean Square	F	p-value.
WIP	Between Groups	1023778.290	6	170629.715	213.822	<0.0001
	Within Groups	2675684.333	3353	797.997		
	Total	3699462.624	3359			
Lead-time	Between Groups	10088.494	6	1681.416	212.141	<0.0001
	Within Groups	26575.715	3353	7.926		
	Total	36664.209	3359			
Defect rate	Between Groups	45554.880	6	7592.480	551.690	<0.0001
	Within Groups	46144.731	3353	13.762		
	Total	91699.611	3359			
Production throughput	Between Groups	1618.933	6	269.822	4.969	<0.0001
	Within Groups	182081.333	3353	54.304		
	Total	183700.267	3359			

We have a significant difference between the WIP means (µ.WIP) ($p<0.0001$), the lead-time means (µ.Leadtime) ($p<0.0001$), the defect rate means (µ.Def) ($p<0.0001$), and the production throughput means (µ.Throughput) ($p<0.0001$) of all running models. Thus, for each dependent variable, at least one model differs from the others. Therefore, we will proceed with the Tukey-Kramer post hoc analysis in order to get the multiple mean comparisons to determine the models that have contributed to generating the statistical difference for all dependent variables.

Tukey-Kramer post hoc analysis

In Figure IV.10, we show the multiple comparison of all models of the market fluctuation scenario for the WIP, lead-time, defect rate, and production throughput dependent variables.

p-value ≤ 0.05
p-value > 0.05
Same Model

WIP

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.001	0.246	0.415	0.000
Actual model	0.000		1.000	0.000	0.000	0.000	1.000
Cross training	0.000	1.000		0.000	0.000	0.000	1.000
Poka Yoke	0.001	0.000	0.000		0.483	0.300	0.000
Pull	0.246	0.000	0.000	0.483		1.000	0.000
SMED	0.415	0.000	0.000	0.300	1.000		0.000
Ucell	0.000	1.000	1.000	0.000	0.000	0.000	

Production throughput

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		1.000	1.000	1.000	0.001	1.000	1.000
Actual model	1.000		1.000	1.000	0.001	1.000	1.000
Cross training	1.000	1.000		1.000	0.001	1.000	1.000
Poka Yoke	1.000	1.000	1.000		0.001	1.000	1.000
Pull	0.001	0.001	0.001	0.001		0.001	0.001
SMED	1.000	1.000	1.000	1.000	0.001		1.000
Ucell	1.000	1.000	1.000	1.000	0.001	1.000	

Lead-time

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.225	0.490	0.000
Actual model	0.000		1.000	0.000	0.000	0.000	0.687
Cross training	0.000	1.000		0.000	0.000	0.000	0.687
Poka Yoke	0.000	0.000	0.000		0.256	0.091	0.000
Pull	0.225	0.000	0.000	0.256		0.999	0.000
SMED	0.490	0.000	0.000	0.091	0.999		0.000
Ucell	0.000	0.687	0.687	0.000	0.000	0.000	

Defect rate

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.000	0.000	0.000
Actual model	0.000		1.000	0.000	0.104	1.000	1.000
Cross training	0.000	1.000		0.000	0.104	1.000	1.000
Poka Yoke	0.000	0.000	0.000		0.000	0.000	0.000
Pull	0.000	0.104	0.104	0.000		0.038	0.092
SMED	0.000	1.000	1.000	0.000	0.038		1.000
Ucell	0.000	1.000	1.000	0.000	0.092	1.000	

Figure IV.10 Multiple comparisons (Tukey) for the Market fluctuation Scenario

At this step we mentioned the observation, the discussion and interpretation will be suggested further.

WIP: For the WIP dependent variable, the $(\mu.\text{WIP})_{5S}$ is significantly different from that of all remaining models (p <= 0.0001) except $(\mu.\text{WIP})_{Pull}$ (p=0.246) and $(\mu.\text{WIP})_{SMED}$ (p=0.415) models. The Actual model, cross training, and Ucell (p>0.05) are almost similar in reaction in the market fluctuation context ($(\mu.\text{WIP})_{actual}$ ~ $(\mu.\text{WIP})_{Crosstraining}$ ~ $(\mu.\text{WIP})_{Ucell}$). However, the mean of the actual model is significantly different from that of Poka Yoke, SMED, pull, and 5S (p<0.0001). Among each other, Poka Yoke, pull, and SMED do not have a significant mean difference (p>0.05). However, the WIP mean of Poka Yoke $(\mu.\text{WIP})_{PokaYoke}$ is significantly different from that of 5S $(\mu.\text{WIP})_{5S}$, Actual $(\mu.\text{WIP})_{Actual}$, Cross training $(\mu.\text{WIP})_{CrossTraining}$, and Ucell $(\mu.\text{WIP})_{Ucell}$ models (p<0.0001). $(\mu.\text{WIP})_{Pull}$ and $(\mu.\text{WIP})_{SMED}$ are significantly different from $(\mu.\text{WIP})_{Actual}$, $(\mu.\text{WIP})_{CrossTraining}$, and $(\mu.\text{WIP})_{Ucell}$ (p<0.0001). The mean of Ucell model $(\mu.\text{WIP})_{Ucell}$ is significantly different from that of all remaining models (p<0.0001) except the actual $(\mu.\text{WIP})_{actual}$ and cross training $(\mu.\text{WIP})_{CrossTraining}$ models (p>0.05).

Lead-time: For the lead-time dependent variable, the means $(\mu.\text{Leadtime})_{5S}$ and $(\mu.\text{Leadtime})_{PokaYoke}$ are significantly different from that of all remaining models and among each other (p <= 0.0001) except from $(\mu.\text{Leadtime})_{Pull}$ and $(\mu.\text{Leadtime})_{SMED}$ (p>0.05). $(\mu.\text{Leadtime})_{Actual}$ and $(\mu.\text{Leadtime})_{CrossTraining}$ do not have a significant difference among each other (p=1); however, they do have significant mean differences compared to the means of all remaining models (p<0.0001) except Ucell model (p=0.687). $(\mu.\text{Leadtime})_{Pull}$ and $(\mu.\text{Leadtime})_{SMED}$ are significantly different from the mean of the actual, cross training and Ucell models (p<0.0001).

Production throughput: For the production throughput of the whole market fluctuation scenario, we can clearly see that only the mean of Pull $(\mu.\text{Throughput})_{pull}$ have a significant mean difference compared to the means of all remaining models (p=0.001). All other models do not have a significant mean difference among each other.

Defect rate: In the defect rate's Tukey post hoc analysis, Poka Yoke and 5S models have significant mean differences in comparison to other models (p<0.0001). All other models do not have a significant mean difference among each other.

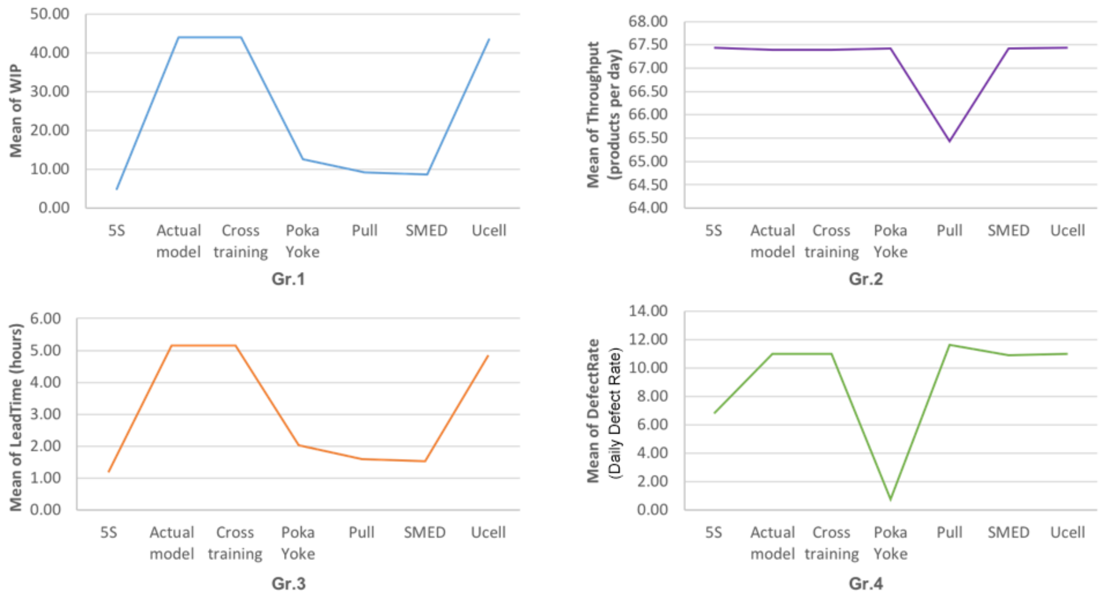


Figure IV.11 Group means of the market fluctuation scenario

The group means of the market fluctuation simulation scenario is represented in Figure IV.11 to easily perceive at a glance the relevancy of Lean tools regarding the expected indictors.

For the whole simulation scenario, 5S has the smallest WIP mean (WIP=5), SMED and pull have also interesting WIP means almost equal to 9, and Poka Yoke has a WIP mean that is almost equal to 13. Cross training and Ucell models have the same WIP mean of the actual model (WIP≈44). As for the lead-time mean results, cross training and Ucell kept almost the same value (~ 5h) of the actual model's lead-time mean. However, Poka Yoke, pull, SMED, and 5S decreases this value to ~ 2h, 1.6h, 1.5h, and 1.1h respectively. It is interesting to observe the same tendency for WIP and lead-times in the context of market fluctuation.

The production throughput was only affected by the pull model that decreases the throughput from ~67.5 to ~65.5 final products per day. The daily defect rate of the cross training, pull, SMED, and Ucell models remained almost at the same level of the actual model's rate. 5S and Poka Yoke models decreased the defect rate by 36% and 90% respectively.

Discussions and interpretation of the Market fluctuation context

Most of the companies focus on maximizing the customer value. They always strive to meet the customer needs and deadlines. Moreover, when focusing on deadlines and market competition, companies should never ignore the importance of the quality targets. When company faces a market demand increase, it should always consider these factors to meet or surpass customer expectation. In this section, we will experiment the market fluctuation scenario to test the performance of each tool when a market fluctuation is induced.

First, we will explain, based on the simulation results and statistical study of the previous section, the importance or uselessness of Lean tools and their potential improvements if they exist, when having market fluctuation context. These analyses concern the overall market fluctuation simulation scenario over the 500h of production (See Figure IV.9). Table IV.4 summarizes the percentage of gains recorded for each KPI regarding each Lean tool scenario simulation.

When having a market fluctuation (increases and decreases) context and based on the group means study of Figure IV.11, 5S showed to be a good performer in terms of WIP and lead-time KPIs, having a significant difference $(\mu.\text{WIP})_{5S}$ and $(\mu.\text{Leadtime})_{5S}$ compared to the actual, cross training, Poka yoke, and Ucell models.

Table IV.4 Lean tools improvements of the market fluctuation scenario based on the group means results

KPI	Lean tool	Improvements compared to the Actual Model (%)	KPI	Lean tool	Improvements compared to the Actual Model (%)
KP1: WIP	5S	88%	KP3: Production throughput	5S	0%
	Cross Training	0%		Cross Training	0%
	Poka Yoke	72%		Poka Yoke	0%
	Pull	77%		Pull	-3%
	SMED	79%		SMED	0%
	Ucell	0%		Ucell	0%
KP2: Lead-time	5S	76%	KP4: Defect rate	5S	41%
	Cross Training	0%		CrossTraining	0%
	Poka Yoke	60%		Poka Yoke	90%
	Pull	68%		Pull	-3%
	SMED	70%		SMED	0%
	Ucell	4%		Ucell	0%

Indeed, based on the group mean study of Figure IV.11 and based on Table IV.4, **5S** scenario was able to minimize the WIP of the actual model by ~88%, the lead time by approximately 76%, and the defect rate by ~36% during 500h of production and throughout all induced market fluctuations. By managing and organizing workspaces, 5S tool improves the processing time, speeds up the flow by getting the right tools and components quickly, and helps on reducing errors of manipulating tools, equipment, and kitting in the front production lines. The clarification of workspace is enhancing the velocity of production and the reliability of the produced components. Defects rate can hence be dropped.

Poka Yoke aims at detecting defects at an early stage, its main target is to improve the product quality. Error prevention at early production stages certainly improves the production flow by reducing reworks. Thus, in addition to the quality improvement (defect reduction), Poka Yoke reduces the WIP and improves the lead-time. Those improvements are directly related to the defects probability that a company faces, as well as to the percentage of defect rate improvement that Poka yoke can bring to the company. Indeed, most the Lean thinking entails the maximization of customer value. Strive to meet the customer needs in terms of conformity and deadlines is the core target. Poka yoke testifies to be a relevant Lean tool for many targets not only "quality" as it can be expected, and not only in disturbed context (Finding.1).

<*Finding.ctx.1.1*> **5S and Poka yoke** are best in class. They can be considered as relevant Lean tools when the decider is inducing a market increase. The previous <Finding.1> outlines the relevancy of that tools event without disturbance. We confirm through this experimentation their maintained effective relevancy when the industrial system is disturbed with a market fluctuation.

Pull and SMED also brought significant improvements for the WIP and lead-time KPIs while confronting market fluctuation, both were able to reduce the WIP during the simulation run by ~77 to 79% and the lead-time by ~68 to 70%. The amount of improvements can be considered as very interesting. The increase of the market forces the company to adapt its own production line organization. The implementation of the **Pull** system allows to treat only pieces required by the market reducing the inflation of storages that can be costly as well as impacting the WIP. So, WIP indicator is interestingly impacted and lead-times are reduced because of the possibility to deliver exactly what is required without handling queues, damages in production lines due to the overproduction. **SMED** is another helpful and relevant tool as it is in charge of reducing the delay of setup times. Indeed, all the delay spent in preparing the production lines is preventing the company from producing. The time consumed for changing reference of products can constitute missing earnings. SMED reduces this non added value time improving the entrance in production lines (Lead times reduced) and absorbing quickly the WIP by introducing easier the references in the process.

<*Finding.ctx.1.2*> PULL and SMED are interesting to reduce WIP and lead-times when industrial system is undergoing a market fluctuation.

Cross training, once again, will have the same behavior of the actual model as we do not have any resource failure requiring the intervention of multi-skills agent. The advantage is not perceptible. Operators and machines are considered always available in this simulation context.

Ucell focuses on reducing the travel time between machines; however, when the market demand of the 4 product references increases, the company will be in need of production speed up and machines' setup time reductions. In such cases, reducing travel time will be insignificant.

<*Finding.ctx.1.3*> **Ucell and Cross training** were found to be without significant improvements on WIP nor lead-times when company is confronting market fluctuations.

In the section below, we will lead different demand fluctuations. For some market fluctuations, we repeated the ANOVA test on the fluctuation period only, in order to validate the output results on this specific period of time. Fluctuations happened as follows,

- *For t=0,* $\forall\ i = 1 \ldots 4,\ \widehat{D}_{XRF_{i(t)}} = D_{XRF_{i0}} + \ 0\%$
  This simulation scenario is performed based on 4 product references. At time t=0, the simulation started with no increase or decrease in the market demand. We can see from

Figure IV.9 that output results are similar to the results studied in the previous section (neutral simulation scenario) as no variations, context changes, or fluctuations took place. Each time the simulation returns to this initial state (0% increase/decrease), the system will almost have the same behavior and no need to comment it again in order not to overload the manuscript.

- *For t=50h*, $\forall\ i = 1 \dots 4,\ \widehat{D}_{XRF_{i(t)}} = D_{XRF_{io}} +\ 15\%$

  At production time t=50h, a market demand increase of 15% is induced to the system. As per Figure IV.9, the WIP was well handled by the PULL, SMED, 5S and Poka Yoke. As for the other models (Actual, Ucell, and cross training), the WIP increased with this fluctuation. Consequently, the WIP associated costs also increased. Moreover, WIP in queue will hold up the production flow, which leads to slow production rates and more problems for not being able to meet the clients' deadlines.

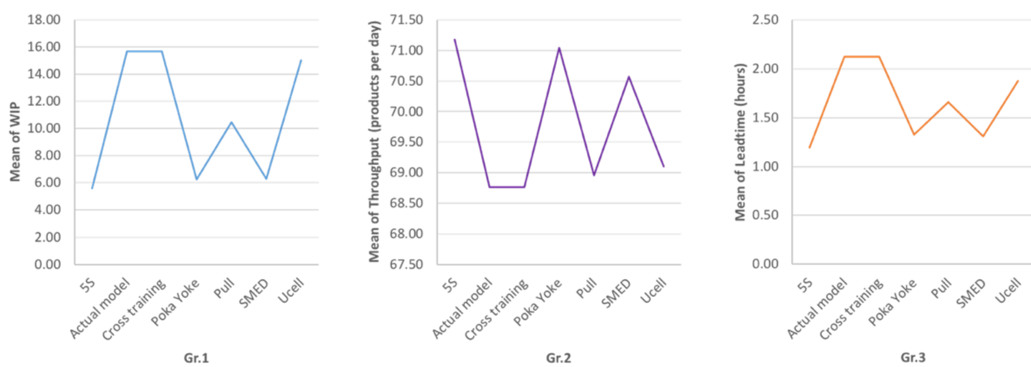


Figure IV.12 Group means of the 15% demand increase between 50h and 100h

In Figure IV.12, we can realize that the WIP means $(\mu.WIP)_{5S}$, $(\mu.WIP)_{PokaYoke}$, and $(\mu.WIP)_{SMED}$ are significantly different from $(\mu.WIP)_{Actual}$, $(\mu.WIP)_{CrossTraining}$, and $(\mu.WIP)_{Ucell}$ where the WIP was reduced by ~60 to 63%. Pull also was able to reduce the WIP during this increase by ~34%. 5S, Poka Yoke, and SMED significantly improved the production throughput. However, $(\mu.WIP)_{Pull}$ and $(\mu.WIP)_{Ucell}$ did not have remarkable improvements in the production throughput KPI. As for the lead-time KPI, 5S, Poka Yoke, and SMED significantly reduced the lead time of the actual model by ~ 40 to 45%. Pull reduced the lead-time of the actual model by approximately 25% during this increase and Ucell reduced it by ~10%. At simulation time t=100h, the order demand is resettled to its initial value "$D_{XRF_{io}}$".

- *For t=130h*, $\forall\ i = 1 \dots 4,\ \widehat{D}_{XRF_{i(t)}} = D_{XRF_{io}} +\ 30\%$

  When the demand is increased by 30% (See Figure IV.9), Ucell's WIP and lead-time also behave similarly to the WIP and lead-time of the actual model. So, having travel time reductions (Ucell) between production machines will not decrease the WIP and lead-time in such fluctuations. Thus, the WIP associated costs will remain increasing and the delay will also remain increasing. It becomes subtill to compare the increase impact: As for the SMED and Poka Yoke, the WIP was controllable when the market increased by 15%. At the 30% increase, both models were not able to control the WIP high increase. However, the Pull model increased to a certain level. At simulation time 160h, it maintained a stable WIP (~20) and a stable lead-time (~3h).

5S maintained a good WIP level as well as good lead-time value, but things might change if the machines' setup or changeover times are higher. At this point, reducing the defect rate and the machines' processing time will not be good enough to stabilize the WIP and lead-time value and therefore SMED might perform better.

The ANOVA test of the production throughput of the whole market fluctuation scenario, the pull had a significant mean difference compared to the means of all remaining models (p=0.001). All other models did not have a significant mean difference among each other. So, we repeated the test for this fluctuation simulation part only, to validate that having a high market increase, 5S, Poka Yoke, and SMED can improve the production throughput of the company (See Figure IV.13). At time 190h, the order demand is resettled to its initial state "$D_{XRF_{i0}}$".

These findings are correlated with <Finding.ctx1.1> and <Finding.ctx1.2>. It reinforces the idea that varying the value of market fluctuation may represent some sensitivity, but global tendency is for the positive impact of 5S, SMED, and Poka Yoke on market fluctuation. Pull model was not able to improve the production throughput during such demand increase.

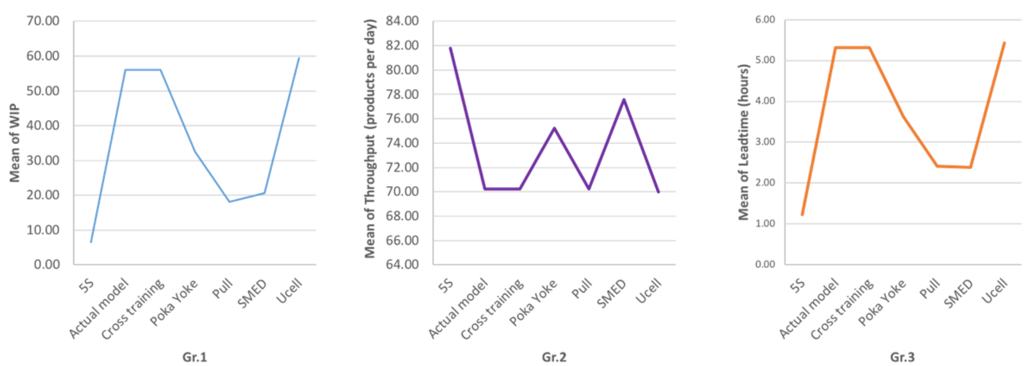


Figure IV.13 Group means of the 30% demand increase between 130h and 190h

- *For t=400h*, $\forall\, i = 1 \ldots 4$, $\widehat{D}_{XRF_{i(t)}} = D_{XRF_{i0}} - 15\%$

  At time 400h, we induced a market demand decrease of 15%, no interesting improvement are shown, and all models have almost the same behavior. In such case, Poka Yoke and 5S are chosen for better quality and for the reduction of rework related costs. We can justify it by several reasons: Poka yoke and 5S are context free tools to improve the quality rate, it has been proved with and without market increases, and it is not surprising to see the same result during market decrease. For the other KPIs, they remain similar to actual model, it is justified by the fact that Lean tools implementation are aiming to optimize the production process to produce effectively and quickly with defect free processes. When the demand is reduced the industrial system is relaxed and no heavy constraint is put over the simulation. The different scenarios behave the same because they are tending to the situation of "context free" as the reduction of order book is just reducing the number of outputs produced. At time 450h, the order demand is resettled to its initial state "$D_{XRF_{i0}}$".

Some findings from the market fluctuation context's study are listed as follows,

<Finding.ctx.1.4> PULL has been found not influencing the improvement of manufacturing throughput. Increasing the market demand and using Pull will not help in serving the entire clients' demands. (Results proven with 15% increase and 30% increase).

<Finding.ctx.1.5> The decrease of the market relaxes the constraints over the system and all implemented Lean tools behave neutrally without interesting impact (market decrease results became almost similar to neutral scenario).

IV.3.3 Demand diversification context <ctx.2>

This context entails the situation where companies are confronted to variety of products to produce based on the market request. The simulations are performed over 1090h of production horizon length. As in previous context, we intend to feed the simulation process with real time input and updated data regarding the product portfolio.

Table IV.5 Number of varieties required by the clients

Production hour (t)	**Number of Product References (n)**
0h	Initial, 2 ref.
50h	4
150h	Initial
300h	8
450h	Initial
650h	16
800h	Initial

As seen in Table IV.5, the simulation scenario starts at time 0h. Initial market demand, machine processing and setup times, and other input data can be found in Appendix A. We started by simulating the company having two types of products (references). After 50h of production, we induced the first change where four product references are required. The overall demand quantity remained the same. We only changed the number of varieties required from the clients. At t=150h, we returned the number of product varieties to its initial state (2 references). At 300h, we tested the variety increase to 8 references. At time 450h, we put back the number of references to 2 then we increased it to 16 at time 650h and finally reset back to 2 references at time 800h.

Simulation graphs

The graph of Figure IV.14 displays the WIP, the production throughput, the lead-time, and the defect rate of the demand diversification scenario during a simulation run over 1090h of production. During this scenario, multiple variations took place. Each black dashed line in Figure IV.14 represents a specific scenario of product portfolio diversification.

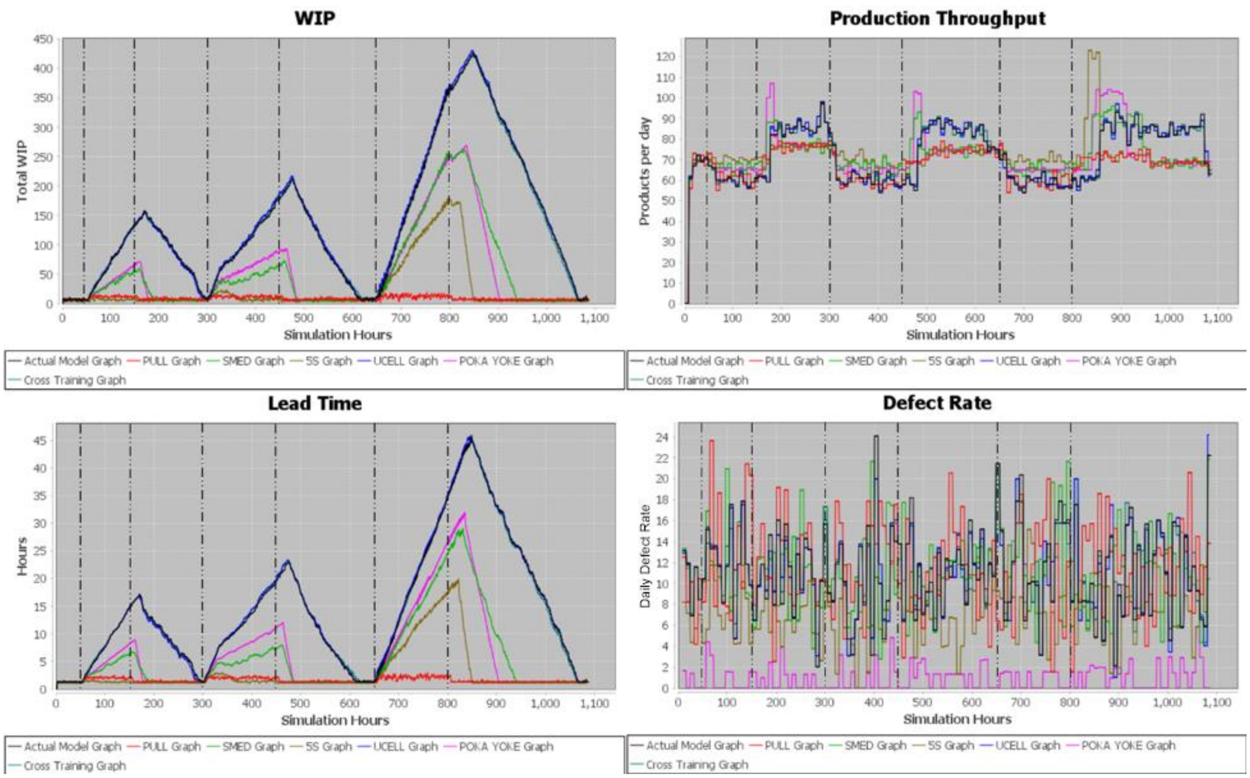


Figure IV.14 Simulation of the demand diversification context

ANOVA test

In the ANOVA test of Table IV.6, all output results' means showed to be significantly different ($p<0.0001$).

Table IV.6 One way ANOVA test of the demand diversification scenario

		Sum of Squares	df	Mean Square	F	p-value
WIP	Between Groups	21884842.461	6	3647473.744	501.048	<0.0001
	Within Groups	55391127.339	7609	7279.686		
	Total	77275969.800	7615			
Lead-time	Between Groups	213295.978	6	35549.330	481.073	<0.0001
	Within Groups	562274.043	7609	73.896		
	Total	775570.021	7615			
Defect rate	Between Groups	99661.158	6	16610.193	1188.358	<0.0001
	Within Groups	106354.320	7609	13.977		
	Total	206015.479	7615			
Production throughput	Between Groups	21997.227	6	3666.204	32.834	<0.0001
	Within Groups	849609.353	7609	111.658		
	Total	871606.580	7615			

We have a significant difference between the WIP means (µ.WIP), the lead-time means (µ.Leadtime), the defect rate means (µ.Defect), and the production throughput means (µ.Throughput) of all running models (as shown in Table IV.6, $p<0.00001$). Thus, for each dependent variable, at least one model differs from the others. Therefore, we will proceed with the Tukey-Kramer post hoc analysis in order to get the multiple mean comparisons to determine the models that have contributed to generating the statistical difference for all dependent variables.

Tukey-Kramer post hoc analysis

In Figure IV.15, we see the multiple comparison of all models of the demand diversification scenario for the WIP, lead-time, defect rate, and production throughput dependent variables. As shown in Figure IV.15, the means that are significant (p-value<0.05) are represented in white boxes. Grey boxes mean that results provided by simulation are not enough discriminated and almost similar. We avoid commenting the observation of Turkey Kramer analysis as it was done in previous sections.

p-value ≤ 0.05
p-value > 0.05
Same Model

Multiple Comparisons (Tukey)

WIP

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.001	0.000	0.000
Actual model	0.000		1.000	0.000	0.000	0.000	0.993
Cross training	0.000	1.000		0.000	0.000	0.000	0.986
Poka Yoke	0.000	0.000	0.000		0.000	1.000	0.000
Pull	0.001	0.000	0.000	0.000		0.000	0.000
SMED	0.000	0.000	0.000	1.000	0.000		0.000
Ucell	0.000	0.993	0.986	0.000	0.000	0.000	

Production throughput

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		1.000	1.000	1.000	0.000	1.000	1.000
Actual model	1.000		1.000	1.000	0.000	1.000	1.000
Cross training	1.000	1.000		1.000	0.000	1.000	1.000
Poka Yoke	1.000	1.000	1.000		0.000	1.000	1.000
Pull	0.000	0.000	0.000	0.000		0.000	0.000
SMED	1.000	1.000	1.000	1.000	0.000		1.000
Ucell	1.000	1.000	1.000	1.000	0.000	1.000	

Lead-time

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.003	0.000	0.000
Actual model	0.000		1.000	0.000	0.000	0.000	1.000
Cross training	0.000	1.000		0.000	0.000	0.000	1.000
Poka Yoke	0.000	0.000	0.000		0.000	0.855	0.000
Pull	0.003	0.000	0.000	0.000		0.000	0.000
SMED	0.000	0.000	0.000	0.855	0.000		0.000
Ucell	0.000	1.000	1.000	0.000	0.000	0.000	

Defect rate

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.000	0.000	0.000
Actual model	0.000		1.000	0.000	0.352	1.000	1.000
Cross training	0.000	1.000		0.000	0.321	1.000	1.000
Poka Yoke	0.000	0.000	0.000		0.000	0.000	0.000
Pull	0.000	0.352	0.321	0.000		0.211	0.376
SMED	0.000	1.000	1.000	0.000	0.211		1.000
Ucell	0.000	1.000	1.000	0.000	0.376	1.000	

Figure IV.15 Multiple comparisons (Tukey) for the demand diversification context's scenario

The group means of demand diversification scenario is represented in Figure IV.16. For the complete simulation scenario, **Pull** has the smallest WIP mean (WIP≈10), **5S** has a WIP mean almost equal to 22, and **Poka Yoke and SMED** have a WIP mean that is almost equal to 50. Cross training and Ucell models have the same WIP mean of the actual model (WIP≈140). As for the lead-time mean results, **cross training and Ucell** kept almost the same value (~ 14h) of the actual model's lead-time mean. However, Poka Yoke and SMED reduced it to ~ 6h. 5S decreases the lead-time to ~3h, and pull model reduced it to 1.9h. The production throughput was only affected by the pull model that decreases the throughput from ~72.4 to ~67.5 products per day. The defect rate of the cross training, pull, SMED, and Ucell models remained almost at the same level (11% of daily defect rate) of the actual model's rate. 5S and Poka Yoke models decreased this rate to 7% and 1% respectively.

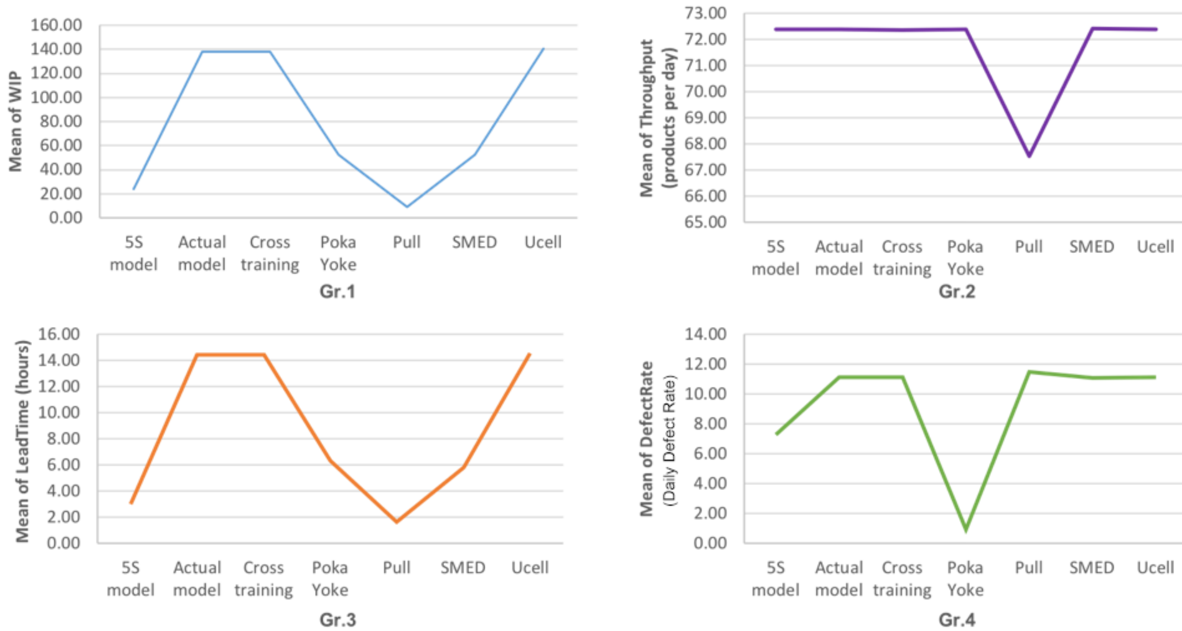


Figure IV.16 Group means of the demand diversification scenario

Discussions and interpretation

Demand Diversification is the necessity in some industrial contexts to widen the range of product portfolio and adapt/organize the production system to be able to deliver this variety. It represents the multiple product references to deal with. The analyses of this paragraph concern the overall demand diversification scenario over the 1090h simulation horizon (See Figure IV.14).

When having references variety and demand diversification context, **pull** showed to be the most powerful tool in terms of WIP and lead-time but not in terms of production throughput. Pull were able to significantly decrease the WIP of the actual model about ~92% and the lead-time about ~86%. However, it reduced the production throughput per day by ~6%. 5S, Poka Yoke, and SMED tools also significantly improved the aforementioned KPIs. During the complete scenario, 5S, Poka Yoke, and SMED were able to decrease the Actual model's WIP by ~84%, 60%, and 62% respectively and the lead-time by ~79%, 57%, and 58% hours respectively.

Once again, cross training has the same behavior of the actual model. Poka Yoke and 5S models are the only operational Lean tools used that focuses on quality improvements, they decreased the Lean free model's defect rate by 92% and 31% respectively.

In the section below, we will study the demand diversification changes and variety. We suggest extending the variety and compare different scenarios. We induced the variety fluctuation as follows:

- *For t=0,* $\forall\, i = 1 \ldots 2, \forall\, u = 1 \ldots k\ ,\ \sum_u \overline{D_{C_{u_t}}}$

  $\sum_u \overline{D_{C_{u_t}}}$ is the sum of all demands required by clients C_u. It is important to note that the overall demand quantity remained the same regardless of the number of references used. At t=0, the simulation started with 2 product references. KPIs results of all models are superimposed and no big differences can be extracted. We can claim that 2 references are considered very low and the behavior of the production system is almost the same.

- *For t=50,* $\forall\, i = 1 \ldots 4, \forall\, u = 1 \ldots k\ ,\ \sum_u \overline{D_{C_{u_t}}}$

  At t=50h, we altered the number of references to 4 product references. In this case, setup time on machines will definitely increase. Thus, the WIP and Lead-time values might be affected. Indeed, as per Figure IV.14, we realized a swift increase of the WIP and lead-time values of the actual, cross training, and ucell models**. Poka Yoke and SMED** improved these KPIs by reducing their values. **Pull and 5S** were able to control this issue and stabilized the WIP and lead-time values. 5S and SMED showed to have the highest production throughput results during this diversification of demand. We repeated the ANOVA test to this part of the simulation for the WIP, production throughput, and lead-time dependent variables in order to have accurate results and precise models' comparisons during such demand diversification. We can deduct from this test and as per Figure IV.17 that **5S, Pull, SMED, and Poka Yoke** significantly improved the WIP by reducing its value by respectively 89%, 84%, 55%, and 49% compared to the actual, cross training, and Ucell models. As per Gr.1 of Figure IV.17, **5S and pull** are the best performer during the increase from 2 to 4 references. Having reduced the WIP to this level, 5S and pull were able to have the best lead-time this simulation period having significant mean differences from that of all remaining models. It is important to notice that although **SMED and Poka Yoke** didn't get the best WIP and lead-time values, but we noticed a good WIP and lead-time improvements compared to the **actual, Ucell, and cross training** models. As for the production throughput (Figure IV.17, Gr.3), we can notice that 5S was the best performer by increasing the throughput of the actual model from 60 products/day to 70 products/day. Once again, as the pull puts a limit to the machine's WIP, it became unable to surpass a certain production rate limit. This is why we see that pull is not having a significant difference in the production throughput compared to the actual model of the company (See Figure IV.17, Gr.2).

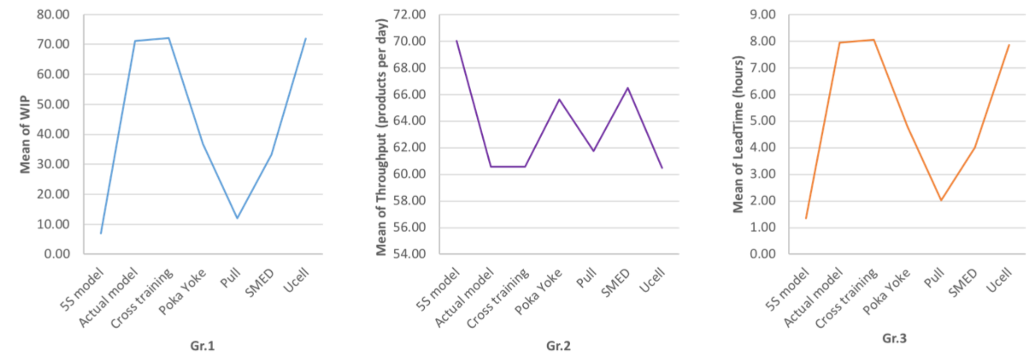


Figure IV.17 Group means of the scenario between 50h and 150h when producing 4 references

- *For t=150,* $\forall\, i = 1 \dots 2, \forall\, u = 1 \dots k\ ,\ \sum_u \overline{D_{c_{u_t}}}$

  At t=150h, we configured the system again with 2 product references. We can notice that in terms of WIP and lead-time values, Poka Yoke had a higher downtrend than the SMED technique when returning back to the initial state (2 references) and this is because the probability of having two consecutive products of different types decreases. Thus, the setup/changeover time on machines will also reduce. Consequently, SMED in 2 product references will not be as effective as it was during the 4 product references production. Therefore, it is interesting to see that even in diversified context SMED can be useless. Two products seem not to be a sufficient threshold to trigger the relevancy of SMED Lean tool.

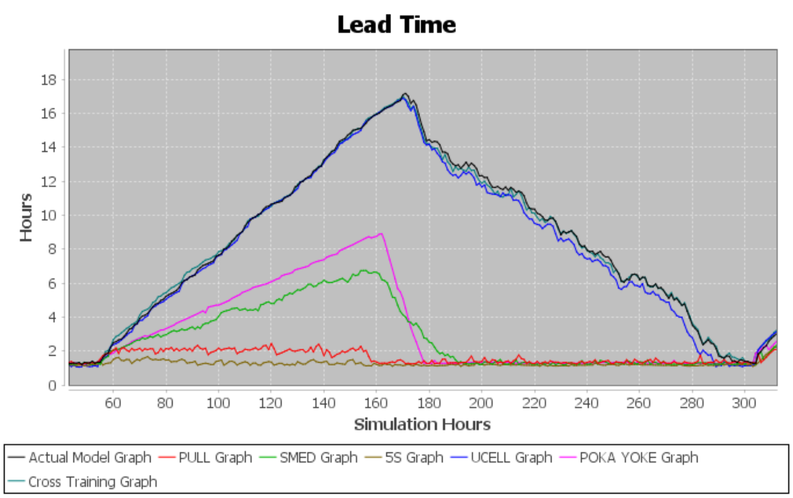


Figure IV.18 Zoom in on Lead time value between 60h and 300h

- *For t=300,* $\forall\, i = 1 \dots 8, \forall\, u = 1 \dots k\ ,\ \sum_u \overline{D_{c_{u_t}}}$

  Once again, at t=300h, we induced a new product diversification by increasing the number of product references to 8 references. We repeated the ANOVA test to this period of production for more accurate results. 5S and pull techniques showed to be the best performers in terms of WIP and lead-time by significantly reducing the WIP and the lead-time of the actual model by respectively ~90% and ~82%. The problem with pull is almost the same; it is true that pull is well controlling the WIP and lead-time, which is a great improvement. However, pull is limiting the production throughput. SMED and Poka Yoke also reduced the WIP by ~60% and ~40% respectively and the lead-time by ~55% and ~35% respectively. During this period SMED had a better control on the WIP and a lower lead-time than the Poka Yoke tool. It is worth noting that at 4 references, both tools had almost close results in terms of WIP and lead-time. When the variety of products increased, SMED had performed better than Poka Yoka and this is due to the number of setup time that increased when the variety increased. 5S, SMED, and Poka Yoke significantly increased the production throughput of the actual model from ~62 products/day to ~70, 67, and 66 products/day respectively.

At t=450h, we put back the number of product references to 2. As per Figure IV.14, the tools have the same reaction that happened on the previous number of references decrease.

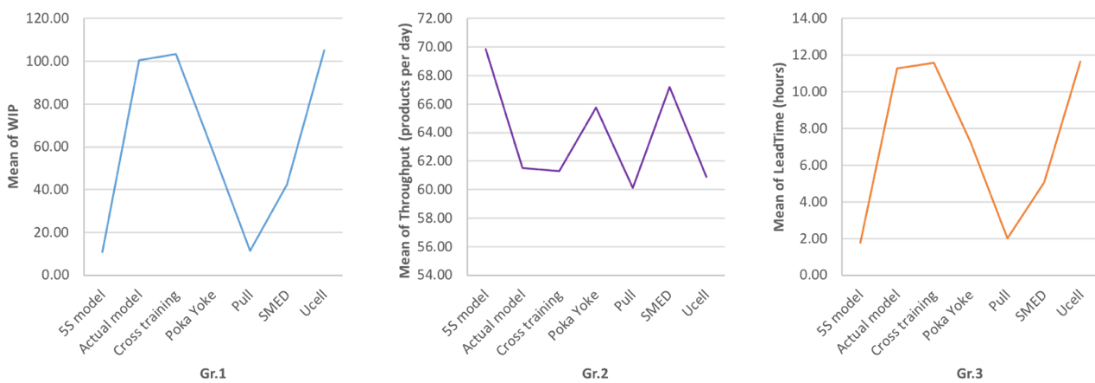


Figure IV.19 Group means of the scenario between 300h and 450h when producing 8 references

- *For t=650,* $\forall\, i = 1 \ldots 16, \forall\, u = 1 \ldots k\ ,\ \sum_u \overline{D_{c_{u_t}}}$

  At t=650h, we parametrized the system to run 16 type of products. During this simulation period, unlike previous fluctuations, 5S was not able anymore to keep a stable WIP and lead-time. These 2 KPIs were increased. The only tool that kept a stable WIP and lead-time was the pull tool. As for the production throughput, 5S had the highest throughput, then came the Poka yoke and SMED tools. We did again the ANOVA test for this simulation part. As per Figure IV.20, pull was a very powerful tool in terms of WIP and lead-time as it was able to decrease the WIP value of the actual model by ~95%. Pull was far way better than other tools when talking about WIP and lead-time KPIs. As usual, 5S, Poka Yoke, and SMED significantly improved the production throughput. At t=800, we returned back the simulation to its initial state (2 references).

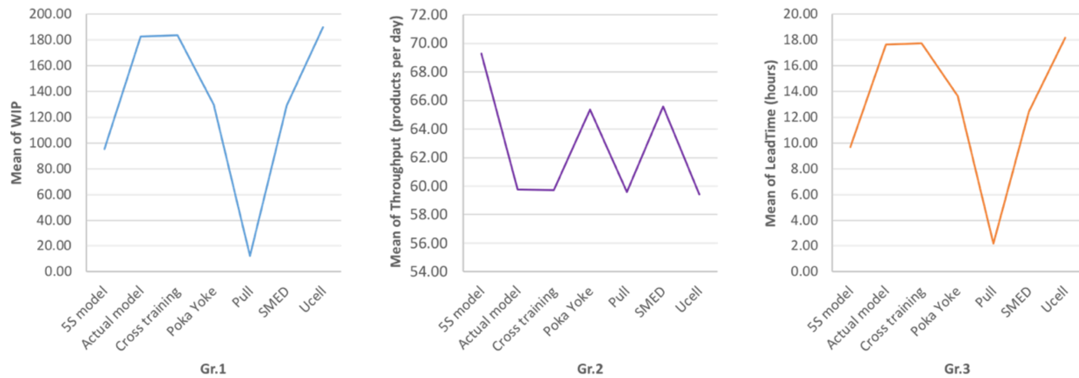


Figure IV.20 Group means of the scenario between 650h and 800h when producing 16 references

<Finding.ctx2.1> Implementing SMED when having minor variety of products results in additional implementation costs without bringing interesting improvements to the operational process.

<Finding.ctx2.2> When having a high demand diversification, 5S technique would help in decreasing the WIP and lead-time KPIs. However, 5S alone, is not able to control the WIP and remove the high overcapacities from the production line. Moreover, it will not lead to the targeted leads-time value. At this level, introducing a pull technique will be more efficient in terms of WIP and lead-time KPIs. Another suggestion would be to use the pull and 5S together in order to have higher throughput and better WIP and lead-time.

<Finding.ctx2.3> Combining different tools would produce powerful improvements. 5S, SMED, Poka Yoke, and pull together would help the company to tackle the cost, quality, and flexibility targets by controlling its WIP, decreasing its lead-time, increasing the production throughput, and finally decreasing the defect rate. In the market fluctuation and demand diversification contexts, Ucell and cross training didn't carry any additional improvement to the production process.

IV.3.4 Uncertainty of resources

The other interesting context to which the company may confront is the non-reliability of its resources. Uncertainty of resources can be any operator absence or machine disruption. To test this kind of disruptions, we added 2 major input fields in the platform called "Number of workers", "Planned down time", and "Unplanned down time". In the number of workers fields, one can choose how many workers exist in each production store. In the Planned down time fields, user should enter the time between down times for each machine along with the time needed to repair the machine. Planned down time could be machine cleaning or maintenance. Same fields exist for unplanned down time that represent any unexpected machine disruptions, errors, interrupts, etc.

Simulation graphs

This simulation scenario is performed over 150h of production to test the uncertainty of resources (machines or operators). Three disruption occurred during this scenario and are listed as follows:

- At *t = 25h*, one operator from the treatment shop was absent for one day
- At *t = 75h*, first machine of the assembly stopped for 2 hours for maintenance issue
- At *t = 100h*, first machine of the machining shop had an unexpected machine error for a whole day of production (8 hours)

Figure IV.21 shows the 4 KPIs of the uncertainty of resources scenario. Each black dashed line of the figure below represents one of the disruptions listed above.

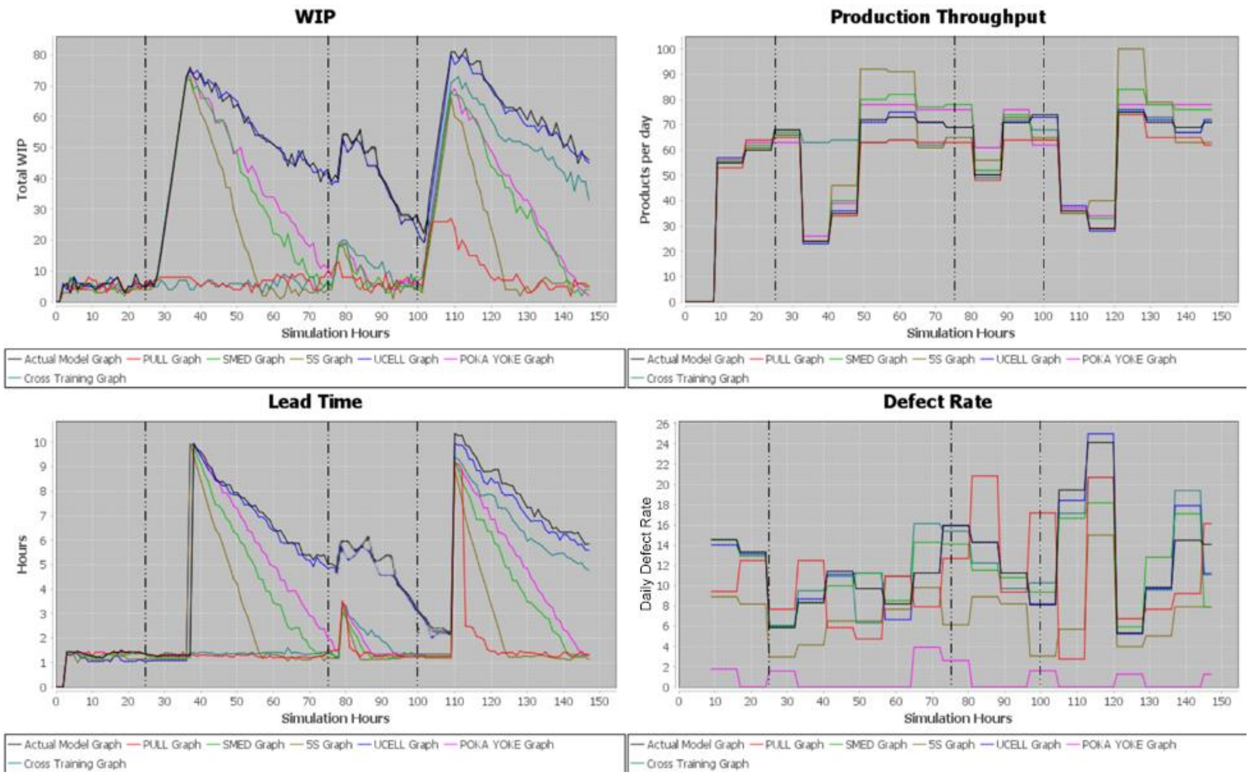


Figure IV.21 Simulation of the uncertainty of resources context

In the ANOVA test of the current scenario, all output results' means are significantly different. We have a significant difference between the means of WIP (µ.WIP) with $p<0.0001$, the lead-time means (µ.Leadtime) with $p<0.0001$, the defect rate means (µ.Def) with $p<0.0001$, and the production throughput means (µ.Throughput) with $p<0.0001$. Thus, for each output dependent variable, at least one model differs from the others and Tukey-Kramer test should be done to get the multiple mean comparisons of the models (See Figure IV.22).

Tukey-Kramer post hoc analysis

WIP: In the multiple comparison test, we realize that $(\mu.WIP)_{CrossTraining}$ is significantly different from all remaining means ($p\leq0.027$) except from the mean of 5S $(\mu.WIP)_{5S}$ where no significant difference exists ($p>0.05$). $(\mu.WIP)_{PokaYoke}$ and $(\mu.WIP)_{SMED}$ are significantly different from all remaining means ($p\leq0.05$) except from each other's means where no significant difference exists ($p>0.05$). The mean of Pull model $(\mu.WIP)_{Pull}$ is significantly different from that of all remaining models ($p<0.0001$). We can see that no significant difference exists between $(\mu.WIP)_{Ucell}$ and $(\mu.WIP)_{Actual}$.

Lead-time: For the lead-time comparison, $(\mu.Leadtime)_{CrossTraining}$ is significantly different from all remaining means ($p\leq0.05$) except from the mean of 5S $(\mu.\ Leadtime)_{5S}$ and the mean of SMED $(\mu.\ Leadtime)_{SMED}$ where no significant difference exists ($p>0.05$). $(\mu.\ Leadtime)_{PokaYoke}$ is significantly different from all remaining means ($p\leq0.05$) except from the mean of SMED $(\mu.\ Leadtime)_{SMED}$ ($p>0.05$). The mean of Pull model $(\mu.\ Leadtime)_{Pull}$ is significantly different from that of all remaining models ($p<0.002$) except from the mean of 5S $(\mu.\ Leadtime)_{5S}$. $(\mu.\ Leadtime)_{SMED}$ is significantly different from all remaining means ($p\leq0.041$) except from the means of cross training $(\mu.\ Leadtime)_{CrossTraining}$ and Poka Yoke where $p>0.05$. we can see again that no significant difference exists between $(\mu.\ Leadtime)_{Ucell}$ and $(\mu.\ Leadtime)_{Actual}$.

Production Throughput: For the production throughput, $(\mu.Throughput)_{Pull}$ and $(\mu.Throughput)_{5S}$ ($p=0.001$), $(\mu.Throughput)_{Pull}$ and $(\mu.Throughput)_{PokaYoke}$ ($p=0.033$), and $(\mu.Throughput)_{Pull}$ and $(\mu.Throughput)_{SMED}$ ($p=0.019$) showed to have a significant differences among each other. All remaining combinations don't have a significant differences between their means.

Defect rate: In the defect rate's Tukey test, Poka Yoke and 5S models have significant mean differences in comparison to other models ($p<0.0001$). All other models do not have a significant mean difference among each other.

p-value ≤ 0.05
p-value > 0.05
Same Model

Multiple Comparisons (Tukey)

WIP

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.993	0.000	0.000	0.003	0.000
Actual model	0.000		0.000	0.000	0.000	0.000	1.000
Cross training	0.993	0.000		0.001	0.000	0.027	0.000
Poka Yoke	0.000	0.000	0.001		0.000	0.975	0.000
Pull	0.000	0.000	0.000	0.000		0.000	0.000
SMED	0.003	0.000	0.027	0.975	0.000		0.000
Ucell	0.000	1.000	0.000	0.000	0.000	0.000	

Production throughput

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.660	0.859	0.967	0.001	0.989	0.679
Actual model	0.660		1.000	0.993	0.203	0.976	1.000
Cross training	0.859	1.000		1.000	0.089	0.998	1.000
Poka Yoke	0.967	0.993	1.000		0.033	1.000	0.994
Pull	0.001	0.203	0.089	0.033		0.019	0.191
SMED	0.989	0.976	0.998	1.000	0.019		0.980
Ucell	0.679	1.000	1.000	0.994	0.191	0.980	

Lead-time

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.906	0.000	0.082	0.041	0.000
Actual model	0.000		0.000	0.000	0.000	0.000	0.999
Cross training	0.906	0.000		0.005	0.002	0.508	0.000
Poka Yoke	0.000	0.000	0.005		0.000	0.543	0.001
Pull	0.082	0.000	0.002	0.000		0.000	0.000
SMED	0.041	0.000	0.508	0.543	0.000		0.000
Ucell	0.000	0.999	0.000	0.001	0.000	0.000	

Defect rate

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.000	0.000	0.000	0.000	0.000	0.000
Actual model	0.000		1.000	0.000	0.439	0.994	1.000
Cross training	0.000	1.000		0.000	0.699	1.000	1.000
Poka Yoke	0.000	0.000	0.000		0.000	0.000	0.000
Pull	0.000	0.439	0.699	0.000		0.857	0.443
SMED	0.000	0.994	1.000	0.000	0.857		0.994
Ucell	0.000	1.000	1.000	0.000	0.443	0.994	

Figure IV.22 Multiple comparisons (Tukey) for the uncertainty of resources scenario

In the group means of Figure IV.23, we can see the exact relative improvements of the Lean tools during an uncertainty of resources context. Unlike other contexts, the cross-training tool showed to offer a significant improvement to the actual scenario by decreasing the actual model's WIP by 60% and decreasing the lead-time by 45%. 5S has almost the same results' improvements in terms of WIP and lead-time. Pull maintained the lowest WIP and best lead-time of the whole scenario study. As for the production throughput, Pull significantly decreased the throughput compared to the 5S, Poka Yoke, and SMED models by ~15%, 12%, and 13% respectively. It is true that in such context cross training should have the highest throughput as all operators are multiskilled and this will potentially reduce the absenteeism and increase the flexibility and reliability. However, it is important to note that we processed the ANOVA test on 150h of production where we had one employee-related disruption and two machine-related disruptions. To overcome this situation, we processed the ANOVA test in the section below to study the cross-training impact on production only at the period of employee disruption.

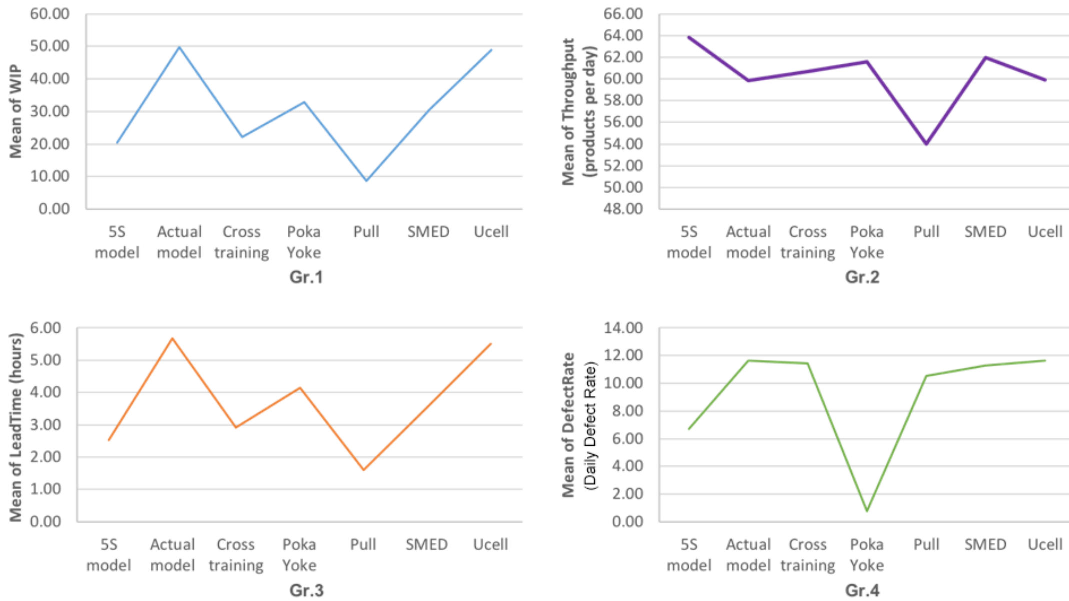


Figure IV.23 Group means of the uncertainty of resources scenario

Discussions and interpretation

We can see below the elements that correspond to operator or machine disturbance. If a disturbance exists, the value will be 1. Otherwise, the value will be 0.

λ_{vt}; $\lambda_{vt} \in \{0,1\}$: *Operator H_v disturbance event (error or absence)*
θ_{pt}; $\theta_{pt} \in \{0,1\}$: *Machine M_p disturbance event (failure, unavailability, or defect)*

We will discuss below the different stages of disturbance.

- *For t=0,* $\forall\, i = 1 \dots 4,\ \lambda_{vt} = \theta_{pt} = 0$
  No interesting improvements to comment. This period is very similar to the first scenario studied where no context or disruptions exist.

- *For t=25h,* $\forall\, i = 1 \dots 4,\ \lambda_{vt} = 1,\ \theta_{pt} = 0$
  This is the only period where cross training model significantly changed its behavior compared to the other models. In this simulation period, λ_{vt} is equal to 1, which means that an employee disturbance exists. At this period, the employee of the treatment shop became absent for 8 working hours (equivalent to one production day). Between 25h and 33h, we can clearly realize from Figure IV.21that cross training was the only performer having its production throughput almost stable and its WIP and lead-time values stable. All other tools except the pull had a high uptrend in their WIPs and lead-

times at this period. Pull didn't have this increase because it does not accept the overcapacity in its production process. After the 8 hours of employee disturbance, we can see that 5S was the fastest tool in returning back to its initial state. In lead-time KPI, SMED started its decrease faster than Poka Yoke until the 45h where they both continued decreasing with a close downward slope. We did the ANOVA test on this period of disruption to validate the significant difference between the cross training and other Lean tools when having employee disturbances ($p<0.001$) (See Figure IV.24).

	5S model	Actual model	Cross training	Poka Yoke	Pull	SMED	Ucell
5S model		0.992	0.001	0.996	0.958	0.999	0.980
Actual model	0.992		0.000	1.000	1.000	1.000	1.000
Cross training	0.001	0.000		0.000	0.000	0.000	0.000
Poka Yoke	0.996	1.000	0.000		1.000	1.000	1.000
Pull	0.958	1.000	0.000	1.000		0.999	1.000
SMED	0.999	1.000	0.000	1.000	0.999		1.000
Ucell	0.980	1.000	0.000	1.000	1.000	1.000	

Figure IV.24 Multiple comparisons for the production throughput during an employee disturbance

As this period of disturbance, cross training had the best performance. It reduced the WIP by ~90%, it increased the throughput by ~25%, and reduced the lead-time by ~75% compared to the actual model (See Figure IV.25). We can conclude from the above that when having employee disturbance, cross training would be the most reliable operational tool to use.

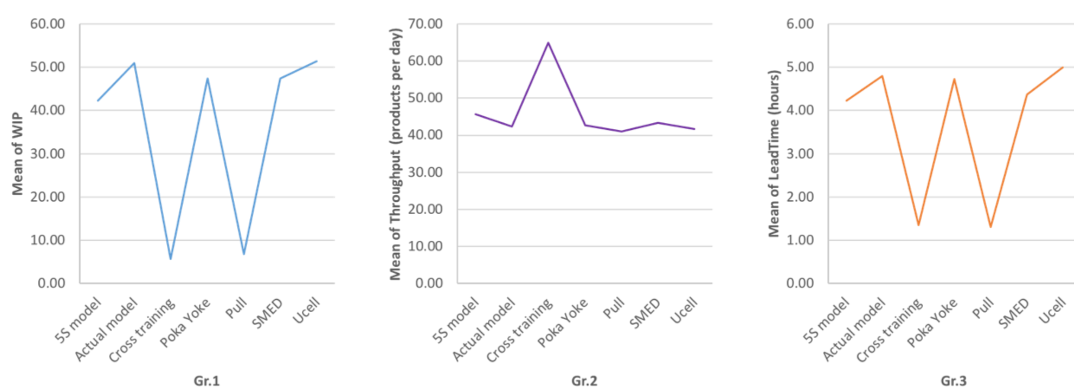


Figure IV.25 Group means of the WIP, Throughput and lead-time during employee disturbance

- *For t=75h*, $\forall\ i = 1 \ldots 4,\ \lambda_{vt} = 0,\ \theta_{pt} = 1$

  At t=75h, an unplanned machine disturbance occurred. The first machine of the assembly line stopped unexpectedly. It took 2h to fix this disturbance and return to the production again. In this type of disturbance, the production stopped for 2 hours. Pull then 5S showed to be the most reliable tools in returning back to the initial stable production state. We repeated this kind of disturbance on the following paragraph for 8 hours (one day) of machine error to have better details and clear answers to this type of disruptions.

- *For t=100h,* $\forall\, i = 1 \dots 4,\ \lambda_{vt} = 0,\ \theta_{pt} = 1$

  We repeated the machine disturbance at t=100h. This time, the disturbance lasted 8h. During this disturbance the production stopped 8h. It is true that Pull didn't allow the WIP overcapacity and that its max WIP allowed is ~25 products in the production process. However, 5S that touched ~65 products in its WIP returned back faster the initial state of production. This explains how fast and reliable this tool is.

<Finding.ctx.3.1> When having an employee disturbance, cross training is a crucial tool to use. When unexpected operator outages occur, other operators can cover if familiar with the equipment. This keeps the manufacturing process going strong and helps in arranging for operator shortages due to sick days, annual leaves, or any retiring operator members.

<Finding.ctx.3.2> When a machine disturbance occurred, the production process will stop during the disturbance time. Thus, a fast and reliable tool should be applied to overcome this issue. 5S showed to be the fastest tool that stabilizes the production process after a machine disruption.

IV.4 Summary, Contributions, and Limitations

Although the simulated scenarios along with their related graphics induced observations and comments; however, the generalization of the global findings may be altered by the complexity and multitude of results. The aim of this section is to clarify the findings in a summarized approach. The target was to gather different elements in a centralized synthetic graphics. We remind the built approach from Chapter II, the objectives that each company is willing to achieve are represented by the selected KPIs. The tools behavior can be represented in the figures and the results developed along the chapter are summarized as "results tendency" in each KPI table. We explain and remind the important findings to ease the understanding of the results.

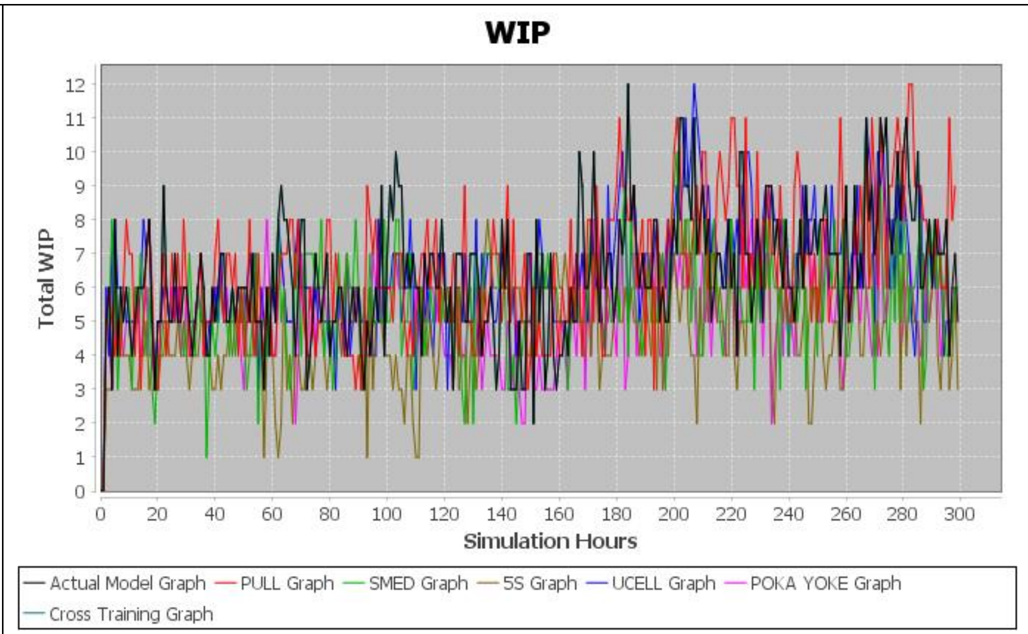
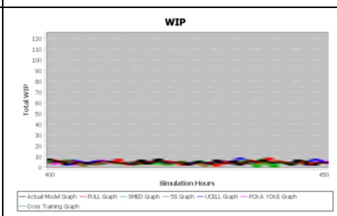
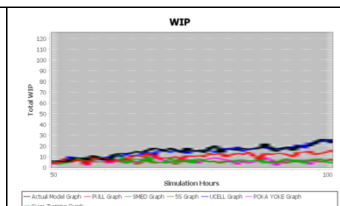
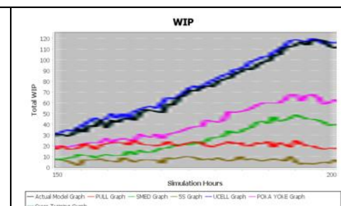
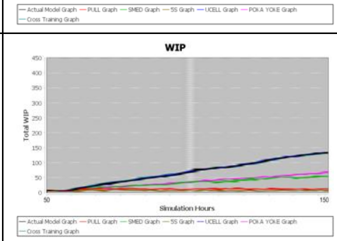
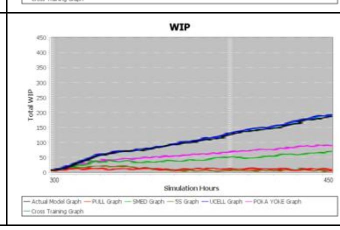
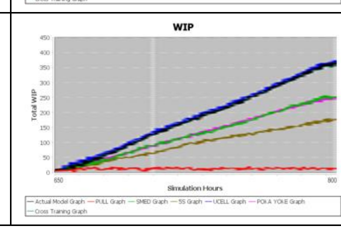
Of course, we remain humble in this research. The target is not to claim finding the relationship between the contexts and the objectives but rather to follow the behavior of manufacturing systems confronted with different contexts and willing to achieve specific objectives. In such situations, the manager will be able to assess the relevance and suitability of each of the discussed Lean tools, and analyze the business behavior and tendency in front of each objective. In this study we tried to analyze deeply different situations. The results can be consulted here after. One of the limitations of this research is the framework defined. Indeed, only 4 industrial contexts were thought, and 4 main core objectives have been defined through wide literature. However, the limited period of the PhD thesis does not allow performing higher number of analysis, hence preventing us from developing the technical support and the built analysis to continue the simulation of the "typology of production" context case.

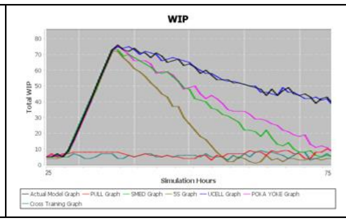
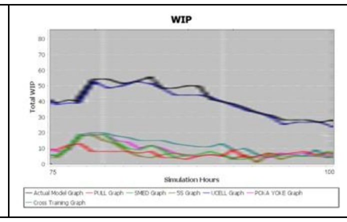
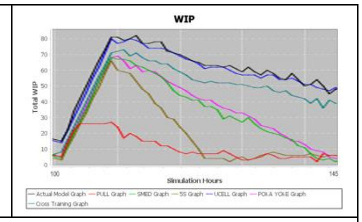
In fact, specifically in this industry, companies may follow the MTS or MTO. Knowing that the configuration and parametrization on the digital platform can be radically different under each case. One of the heaviest parametrizations can be the decoupling point positioning to simulate which parts are under forecasting and which parts are under orders. Moreover, the study case as it designed actually does not permit to move the processes easily. If the study has to go deeper with that point, the digital platform of simulation has to be configured heavily to allow the realization of configurable production lines with movable possibility of positioning the order point.

Therefore, in addition to the reasons listed above, and in light of the development complexity considering the timeframe of this thesis, in addition to the limitation of resources in terms of hosting servers; all these constraints justify the limitation of this work. Nevertheless, we found interesting to keep the typology of production in the core development of our methodology because it is really representative of the industrial structure and organization of production planning. We can evoke the extension of the actual digital platform in conclusion and perspectives.

In the section below, we will summarize the Lean tools' simulation results by KPI (WIP, lead-time, production throughput, and defect rate) for each of the studied industrial contexts.

Table IV.7 WIP regarding the different contexts' analyses

Neutral Mode *(no variation)*			
Market fluctuation *(-15%, +15%, +30%)*			
Demand diversification *(4ref., 8ref., 16ref.)*			

<table>
<tr>
<td>Uncertainty of resources
(Operator(8h), Machine(2h), Machine(8h))</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td colspan="4">Resulted tendency
- Pull is best in class during high demand fluctuations or references diversification.
- In low and medium demand fluctuations, SMED, 5S, and Poka Yoke are good performers.
- Pull had the best influence on reducing the WIP when high demand diversification is undergone.
- In demand diversification, SMED, Poka yoke and 5S reduces the WIP.
- Cross training and pull are the best in class if an operator disruption occurred.
- Cross training and Ucell have no impact on WIP improvement in demand diversification.
- 5S is most reliable tool if a machine disruption occurred.
- Cross training reduces interestingly the WIP when confronted to resource uncertainties.</td>
</tr>
</table>

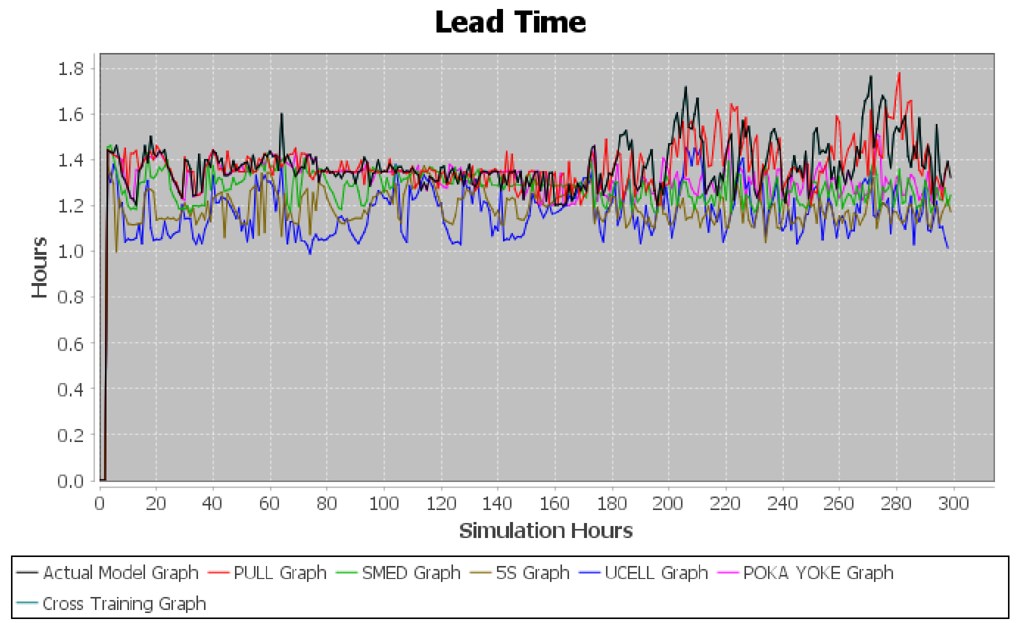
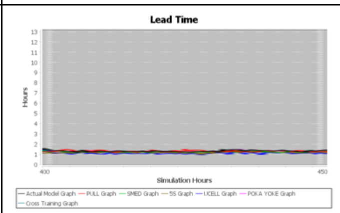
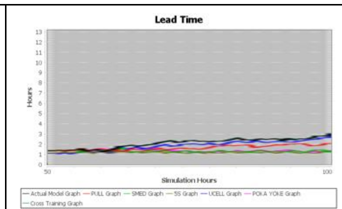
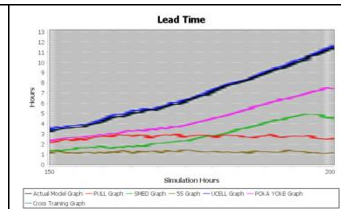
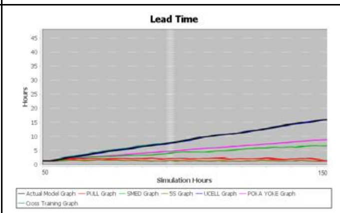
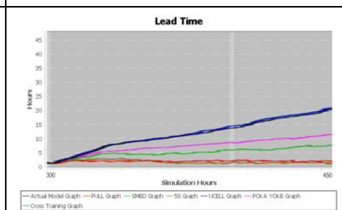
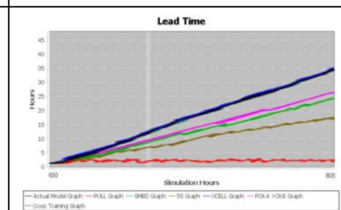
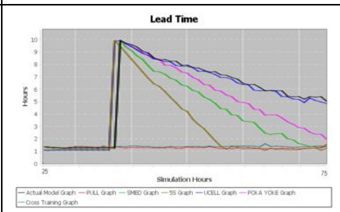
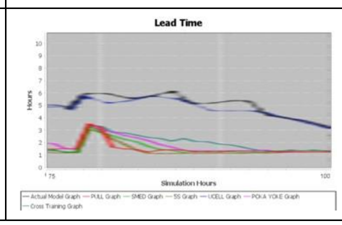
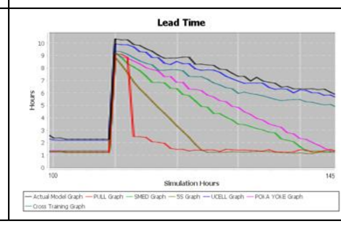
In Table IV.7, we summarize the simulation results of the WIP during the neutral scenario and for all context variations. Starting with the neutral scenario where no context or disruption occurs, all Lean tools results are superimposed but we can see a dominance of the SMED and 5S tools for the lowest WIP values. Indeed, in the neutral scenario, we considered having 4 product references, by reducing the machines setup time, SMED will definitely decreases the WIP. As for 5S, by reducing the defect rate and improving the processing time on machines, 5S contributed in decreasing the WIP value.

When we decreased the market demand by 15%, all tools' results were superimposed, and no interesting outcomes can be determined. When increasing the demand by 15%, Poka Yoke, SMED, and 5S were the best in class. However, after a 30% of demand increase, Poka Yoke and SMED were not able to handle such increase, we have seen an overcapacity in their WIPs. In such increase, 5S and Pull are the best in class.

As for the demand diversification context, when having a 4 to 8 product references, SMED, Poke Yoke, 5S are good performers. When the variety of products become high (e.g. 16 references) those tools will not be able to control the WIP's overcapacity, pull production will be the only solution to keep the WIP at a low level.

For the context of uncertainty of resources, cross training and pull were the only tools capable of reducing the WIP overcapacity if an operator disruption occurred. However, when a machine disturbance occurred 5S showed to be the most reliable tool.

Table IV.8 Lead-time Analysis

Neutral Mode *(no variation)*			
Market fluctuation *(-15%, +15%, +30%)*			
Demand diversification *(4ref., 8ref., 16ref.)*			
Uncertainty of resources *(Operator(8h), Machine(2h), Machine(8h))*			

Resulted tendency

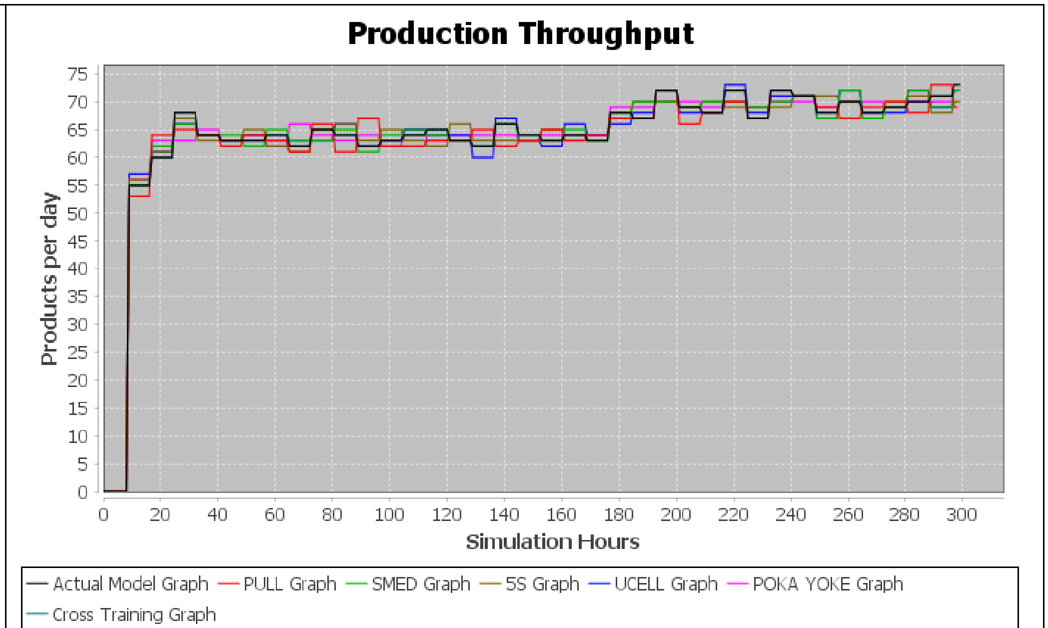
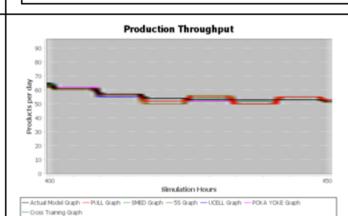
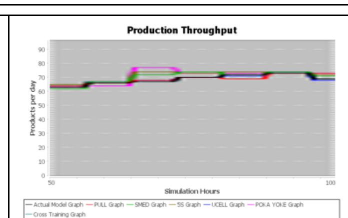
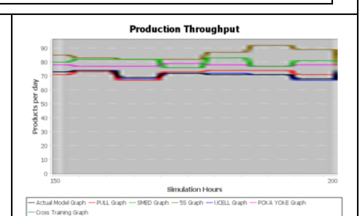
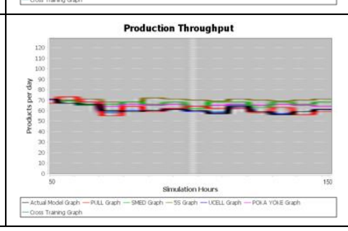
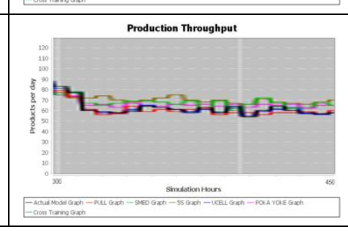
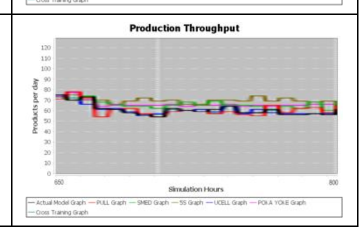
- Ucell and 5S had the best lead-time when no context or variation exists.
- Pull and 5S are best in class during high market increases and during demand diversification contexts.
- Pull and 5S was the best in class until 8 product references. At 16 references, 5S had a high increase in the lead-time. However, pull kept a stable level of lead-time.
- Cross training and Pull showed to be the best techniques to use when having an operator disruption. Cross training seems useless to improve lead-times when the market is undergoing fluctuations.
- Ucell has no interesting impact on improving lead-times when the market is fluctuating.
- During machine disruption, 5S showed to be the most reliable tool.

In Table IV.8, we outline the lead-time results. When no variety of products exists, and when no context arises, Ucell and 5S were the best in class in terms of lead-time. During the market decrease of 15%, the results were superimposed, and no interesting outcomes can be realized. At a 15% of market increase, SMED, Poka Yoke, and 5S were the best in class. However, at a 30% increase, Pull and 5S were the best in class.

For the demand diversification context, when having less than 8 references, Pull and 5S were classified as best tools to use. However, when increasing the number of references to 16, 5S was not able to control the WIP overcapacity which led to an increase in the lead-time. Pull at this level was the best in class.

In the uncertainty of resources context, cross training and Pull showed to be the best techniques to use when having an operator disruption. As for machine disruption, we tested the tool reliability in returning to the stable initial state after a machine disruption induction. 5S showed to be the most reliable tool, it has the highest downward slope in lead-time.

Table IV.9 Production throughput Analysis

Neutral Mode *(no variation)*			
Market fluctuation *(-15%, +15%, +30%)*			
Demand diversification *(4ref., 8ref., 16ref.)*			

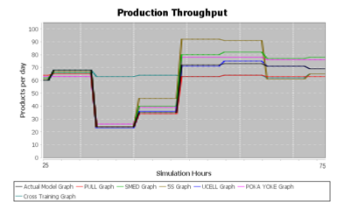
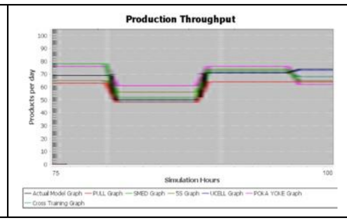
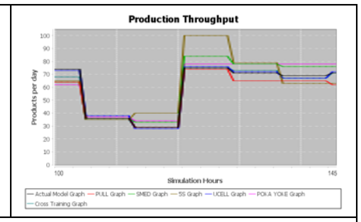
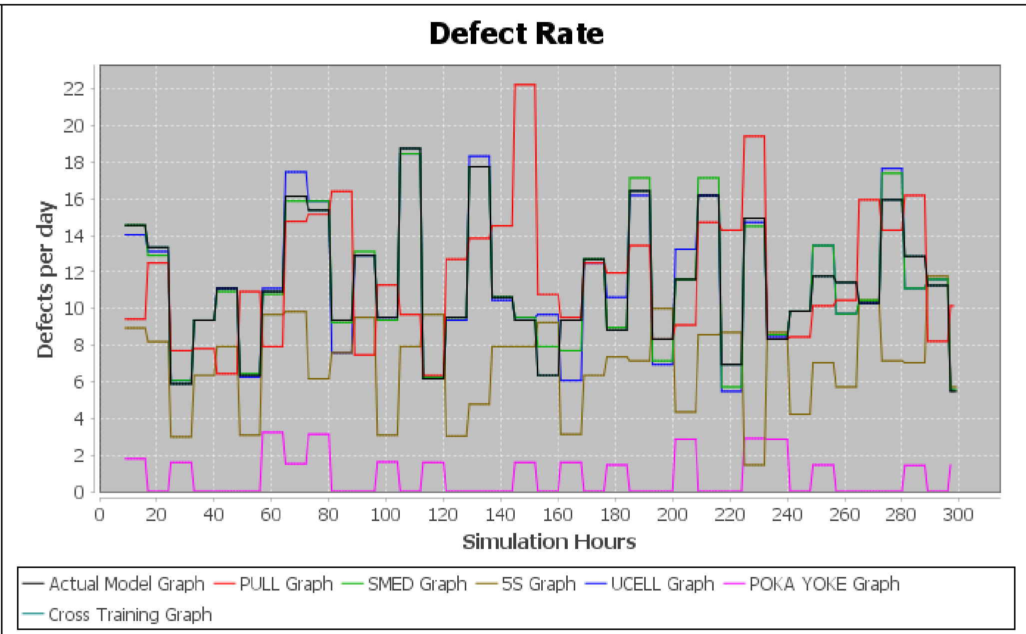
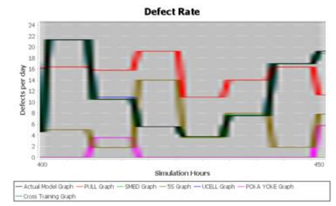
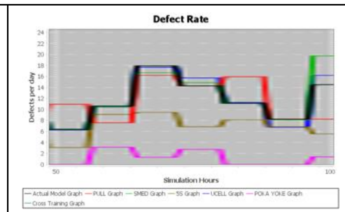
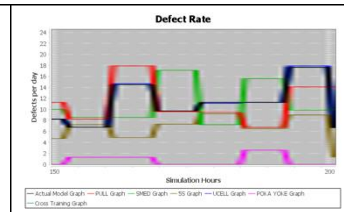
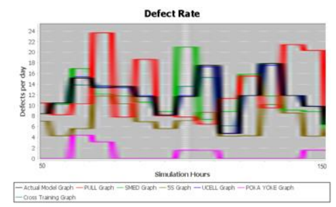
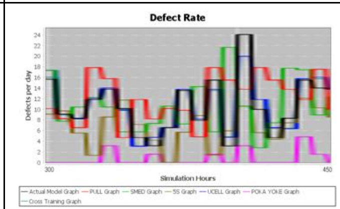
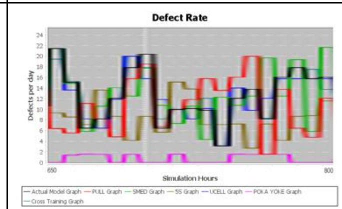
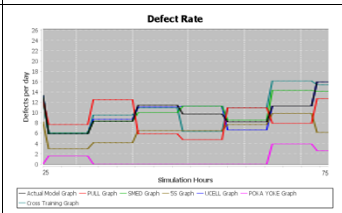
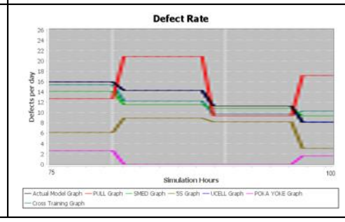
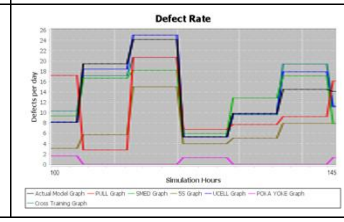
Uncertainty of resources *(Operator(8h), Machine(2h), Machine(8h))*			
Resulted tendency - 5S, SMED, and Poka yoke contributed in increasing the production throughput during market demand increase or demand diversification increase - Pull did not contribute in improving the production throughput during demand increases. A threshold exists over which Pull can no more improve the throughput. - Cross training is an essential tool to use if an operator disturbance arises			

Table IV.9 shows the production throughput of the different contexts analyzed. In the neutral scenario, all models' results are super imposed, and no interesting findings can be determined. 5S, SMED, and Poka yoke have contributed in increasing the production throughput during the market fluctuation or demand diversification contexts. Compared to the actual model, pull was not able to increase during the aforementioned contexts fluctuation.

During the operator disturbance, obviously, cross training had the highest throughput level. When machine disturbance happened, 5S had the highest throughput in returning back to its initial production level.

Table IV.10 Defect Rate Analysis

Neutral Mode *(no variation)*	

Market fluctuation *(-15%, +15%, +30%)*			
Demand diversification *(4ref., 8ref., 16ref..)*			
Uncertainty of resources *(Operator(8h), Machine(2h), Machine(8h))*			
Resulted tendency - Poka Yoke and 5S prior – best in class (Neutral scenario, market fluctuation and demand diversification). - These Lean tools can be considered context-free. - When quality objective is required, Poka Yoke and 5S are prior to be considered.			

In Table IV.10, we outline the defect rate KPI results for all studied contexts and at each fluctuation point. In all contexts and even if no context or disruption exists, we can realize that Poka Yoke and 5S were the only tools that remarkably decreases the daily defect rate of the company regardless what the context is. Both tools react as best in class, they are required as pre-requisites to improve the quality of production independently from the industrial context.

IV.5 Conclusion

The developed Co-Simulation digital platform may offer the possibility for managers and decision makers to lead experiments and observations of different operational Lean tools. For this purpose, the users load then run the Lean tools in parallel in order to select the tools that best fit their companies' profiles and contexts. In Chapter 4, we led a study on these tools based on the aeronautical case study and some contexts' variations. All input data (setup times, processing times, travel times) and Lean tools' parameters can be found in Appendix A. Four scenarios were experimented in this chapter (neutral scenario, market fluctuation, demand diversification, and uncertainty of resources) and different findings were subtracted from the simulation graphs. Output results were statistically validated using ANOVA test and Tukey post hoc analysis.

In the neutral scenario where no contexts or fluctuations arise, the simulation showed that 5S and Ucell are good tools to use to decrease the production lead-time. Poka Yoke had little improvements in the lead-time KPI. However, cross training and pull techniques didn't bring interesting improvements to the production process. We were unable to identify the flexibility and reliability objectives in this scenario since no significant variations exists and Lean tools results are superimposed on the KPI graphs.

Independently from contexts, 5S and Poka yoke aim to reduce the cost associated with rejects, rework and rescheduling. Thus, it is essential to consider both tools as pre-requisites whatever the situation is.

In the market fluctuation scenario, pull failed to increase the production throughput when high market demand increases occur. There is a threshold over which Pull can no more increase its production throughput. 5S showed to be the best in class during this context fluctuations. SMED and Poka yoke improved the WIP and lead-time values. However, during high demand increase (+30%), both were unable to control the WIP overcapacity and consequently had a high WIP and lead-time increases. Moreover, 5S, SMED, and Poka Yoke tools improved the production throughput during demand increases. Ucell and cross training do not have any significant improvements on the production when confronted to market fluctuation context.

In the demand diversification scenario, we tested the production on 2, 4, 8, and 16 references. During this scenario, poka yoke, and SMED decreased the WIP and lead-time values. However, when the variety of products increased to a certain threshold, the aforementioned tools had an overcapacity in their WIPs, which led to an increase in their lead-times. Pull and 5S are best in class during demand diversification context. But, when the number of product references increased to 16 references, 5S had a high increase in the lead-time. Pull maintained a stable level of lead-time.

During an operator disturbance, cross training was the only tool capable of keeping the production running. When a machine disturbance occurred, 5S showed to be the most reliable tool in stabilizing the production process after this disruption. Moreover, during the uncertainty of resources context, pull had a good control on the WIP.

The above results are useful and can be taken into consideration when a company confronts to any of the above contexts. Actually, we presented the use of our co-simulation framework for industries willing to experiment Lean tools' implementation within their production processes. Users can use this tool with different parameters and configurations that suit their companies or industries. They can experiment multiple contexts, changes, and input configurations (Lean tools configuration, market demand, machines' processing and setup times, planned and unplanned down time, etc.)

General Conclusion and Perspectives

The research presented in this Ph.D. dissertation contributes to the comparison of different Lean tools' results based on simulation. The main challenge of this research was to propose an HLA-based co-simulation framework that simulates these Lean tools, along with the actual model of an aeronautic company, simultaneously in parallel in order to explore the results and tools' effects on the production process.

Leading Manufacturers are increasingly adopting Lean in their manufacturing systems. Lean tools and techniques are becoming essential to eliminate or minimize waste and non-value activities from the manufacturing process. However, Lean implementation requires an in-depth study of the company's context to implement the adequate Lean techniques and ensure financial and quality gains. Many manufacturing organizations are inefficiently using Lean tools, considering that Lean brings benefits despite the nature of implemented tools; most of these organizations are experiencing failure.

In this study, we worked on the operational Lean tools that can be technically simulated and configured. Some Lean tools, classified as soft tools in Chapter I, are essential and establish a vital part of Lean thinking to successfully lead a Lean project. However, it is difficult to model and simulate these Lean tools as they are human-related tools (e.g. leadership, managers involvement, etc.). Therefore, we assume that these soft tools are acquired as a prerequisite. Then, the operational tools chosen for simulation in this research are: 5S, SMED, Pull, Poka Yoke, Cross training, and Ucell.

Our proposed methodology combines industrial contexts and objectives. It is important to heed the industrial context that the company is confronted to, relevant Lean tools should be chosen accordingly. Market fluctuation, demand diversification, and uncertainty of resources are the contexts studied in this research. The chosen industrial objectives (quality, cost, reliability, and flexibility) remain within the operational level to manage the flows and control the production process from raw materials to finished products.

We developed a Co-Simulation framework based on the High-Level Architecture (HLA) standard and Discrete Event Simulations. Six Lean Configuration Scenarios developed under JaamSim (DES) as HLA federates are investigated under the aforementioned contexts and compared with an Actual model simulated as a Lean Free scenario. We also developed an external Java Platform directly linked to a master federate responsible of sending/receiving data to/from all other federates, each representing a Lean tool.

Time synchronization between federates is essential to have all DESs running in parallel. Time management mechanisms of HLA are responsible for regulating the advancement of the federates during the simulation run. Event-based time advancement service is used in our study to process all events in time stamp order.

A substantial effort was dedicated to developing a new module for Jaamsim (Java open source DES) in order to transform it to an HLA compatible DES software capable of interacting and exchanging data with external federates. These functionalities were essential in our research to simulate all the Lean tools (federates) in parallel and change their input data during the simulation run to experiment the Lean tools' behavior and responsiveness. Federates' (Lean models) output results are displayed simultaneously in a real time appealing graphical presentation. Those outputs represent the KPIs (WIP, lead-time, production throughput, and defect rate) that will be used to lead the companies to their targets and objectives.

We presented in Chapter IV the study of a neutral scenario where no context or fluctuation exists, the market fluctuation where different fluctuations (demand increases/decreases) took place, the demand diversification context in which the production of different product references is tested (ranging from 2 references to 16 references), and the uncertainty of resources context where machine and operator disruptions are experimented. Output results are statistically validated and analyzed in the last Chapter.

The use of HLA standard broadens our horizons and opens the door for the development of additional Lean tools. Six Lean tools are developed till now, the goal is to expand the built co-simulation framework to gradually integrate other Lean techniques. The co-simulation framework will enable us to create and run multiple Lean scenarios over a broad processors' network. Using our framework and digital platform, we can introduce modifications and disruptions in many variables from conception to commercialization (market demand, travel time, processing time, setup time, planned/unplanned down time, defects, etc.). Different hypothesis leading to different and diverse output results can be explored on this framework.

The limited period of the PhD does not allow testing more contexts and integrating additional modules. Nevertheless, we have been able to present the results in four international conferences listed on the following page. The development of the "Typology of production" context is one of the interesting targets we intend to develop in the near future. In addition, the developed framework supports up to eight machines or workstations in the simulation models, we are aiming to increase this number so that, any company having more than eight machines, will be capable to simulate and explore lean tools' effect on its production line. Moreover, testing combined contexts would also be interesting to analyze. In some cases, companies are faced to market demand increase, demand diversification, and uncertainty of resources at the same time; testing this case will maybe lead to different tools' behavior. Many other contexts and ideas can also be examined using this framework. Finally, the developed framework is a new entry that will assist managers and decision makers in leading experiments of multiple Lean tools and determining the tools that best fit their companies' contexts and profiles.

Publications resulting from this research

- Jalal Possik, Andrea D'Ambrogio, Gregory Zacharewicz, Aicha Amrani, and Bruno Vallespir, "A BPMN/HLA-Based Methodology for Collaborative Distributed DES" presented at the 28th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE-2019), Capri, Italy, 2019.

- Jalal Possik, Aicha Amrani, and Gregory Zacharewicz, "Development of a co-simulation system as a decision-aid in Lean tools implementation" presented at the Proceedings of the 50th Computer Simulation Conference, Bordeaux, France, 2018.

- Aicha Amrani, Jalal Possik, Yves Ducq, and Gregory Zacharewicz, "Contribution to a Lean Maturity Evaluation: Leanness Metrics Calculation," presented at the PMA 2018 - Performance Measurement and Management in a Globally Networked World, Warsaw, Poland, 2018.

- Jalal Possik, Aicha Amrani, and Gregory Zacharewicz, "WIP: Co-simulation system serving the configuration of Lean tools for a manufacturing assembly line," presented at the Works in Progress Symposium, WIP 2018, Part of the 2018 Spring Simulation Multiconference, SpringSim 2018, Baltimore, United States, 2018.

References

Abdulmalek, F. A., and J. Rajgopal. 2007. Analyzing the benefits of lean manufacturing and value stream mapping via simulation: A process sector case study. *International Journal of Production Economics* 107 (1):223-236.

Abraham, A., K. Ganapathi, and K. Motwani. 2012. Setup time reduction through SMED technique in a stamping production line. *SASTECH Journal* 11 (2):47-52.

Achanga, P., E. Shehab, R. Roy, and G. Nelder. 2006. Critical success factors for lean implementation within SMEs. *Journal of Manufacturing Technology Management* 17 (4):460-471.

Agarwal, A., R. Shankar, and M. Tiwari. 2006. Modeling the metrics of lean, agile and leagile supply chain: An ANP-based approach. *European Journal of Operational Research* 173 (1):211-225.

Agarwal, R., R. Green, P. J. Brown, H. Tan, and K. Randhawa. 2013. Determinants of quality management practices: An empirical study of New Zealand manufacturing firms. *International Journal of Production Economics* 142 (1):130-145.

Aguado, S., R. Alvarez, and R. Domingo. 2013. Model of efficient and sustainable improvements in a lean production system through processes of environmental innovation. *Journal of Cleaner Production* 47:141-148.

Al-Araidah, O., A. Momani, M. Khasawneh, and M. Momani. 2010. Lead-time reduction utilizing lean tools applied to healthcare: The inpatient pharmacy at a local hospital. *Journal for Healthcare Quality* 32 (1):59-66.

Alarcón, L. F., S. Diethelm, O. Rojo, and R. Calderón. 2005. Assessing the impacts of implementing lean construction Evaluando los impactos de la implementación de lean construction. *Revista Ingenieria de Construccion*.

Alaskari, O., M. M. Ahmad, N. Dhafr, and R. Pinedo-Cuenca. Critical Successful Factors (CSFs) for Successful implementation of Lean tools and ERP systems. Paper read at Proceedings of the World Congress on Engineering WCE 2012, July 4 - 6, 2012, London, U.K., 2013-08-27.

Alavi-Moghaddam, M., R. Forouzanfar, S. Alamdari, A. Shahrami, H. Kariman, A. Amini, S. Pourbabaee, and A. Shirvani. 2012. Application of queuing analytic theory to decrease waiting times in emergency department: does it make sense? *Archives of trauma research* 1 (3):101.

Alegre, U., J. C. Augusto, and T. Clark. 2016. Engineering context-aware systems and applications: A survey. *Journal of Systems and Software* 117:55-83.

Amrani, A. 2017. Lean and Six Sigma in aeronautic industry *Internal Report Study case, University of Bordeaux, 2017*.

Amrani, A., J. Possik, Y. Ducq, and G. Zacharewicz. 2018. Contribution to a Lean Maturity Evaluation: Leanness Metrics Calculation. In *PMA 2018 - Performance Measurement and Management in a Globally Networked World*. Warsaw, Poland.

Anand, G., and R. Kodali. 2008. Selection of lean manufacturing systems using the PROMETHEE. *Journal of modelling in management* 3 (1):40-70.

Anastasia. *Understanding Work-In-Progress (WIP) When Analyzing Financial Statements* 2018 [cited. Available from https://www.cleverism.com/work-in-progress-wip/.

Angelis, J., R. Conti, C. Cooper, and C. Gill. 2011. Building a high-commitment lean culture. *Journal of Manufacturing Technology Management* 22 (5):569-586.

Antony, J., and R. Banuelas. 2002. Key ingredients for the effective implementation of Six Sigma program. *Measuring business excellence* 6 (4):20-27.

Antony, J., and D. A. Desai. 2009. Assessing the status of Six Sigma implementation in the Indian industry: results from an exploratory empirical study. *Management research news* 32 (5):413-423.

Antony, J., G. Manville, R. Greatbanks, R. Krishnasamy, and D. W. Parker. 2012. Critical success factors for Lean Six Sigma programmes: a view from middle management. *International Journal of Quality & Reliability Management* 29 (1):7-20.

Arbulu, R., G. Ballard, and N. Harper. 2003. Kanban in construction. *Proceedings of IGLC-11, Virginia Tech, Blacksburgh, Virginia, USA*:16-17.

Arunagiri, P., and A. Gnanavelbabu. 2014. Identification of High Impact Lean Production Tools in Automobile Industries using Weighted Average Method. *Procedia Engineering* 97:2072-2080.

Bae, J.-W., and Y.-W. Kim. 2007. Sustainable value on construction project and application of lean construction methods. *Proceedings of IGLC-15*:16-22.

Baglin, G., and M. Capraro. 2000. L'Entreprise Lean Production ou la PME compétitive par l'action collective.

Baglin, G., V. Malleret, and H. Groupe. 2004. *Le développement d'offres de services dans les PMI*: Groupe HEC.

Balin, S. 2007. Amélioration de processus de production de services par la simulation, Paris 9.

Bamber, L., and B. G. Dale. 2000. Lean production: A study of application in a traditional manufacturing environment. *Production Planning & Control* 11 (3):291-298.

Banks, J. 1998. *Handbook of simulation: principles, methodology, advances, applications, and practice*: John Wiley & Sons.

Barnes, C. D., and K. R. Laughery. 1998. Advanced uses for Micro Saint simulation software. Paper read at 1998 Winter Simulation Conference. Proceedings (Cat. No.98CH36274), 13-16 Dec. 1998.

Bartels, N. 2005. Lean, in the most general sense. *Manufacturing Business Technology* 23 (4):32-34.

Bayo-Moriones, A. 2010. 5S use in manufacturing plants: contextual factors and impact on operating performance. *International Journal of Quality & Reliability Management* 27 (2):217-230.

Bazire, M., and P. Brézillon. 2005. Understanding context before using it. Paper read at International and Interdisciplinary Conference on Modeling and Using Context.

Begam, M. S., R. Swamynathan, and J. Sekkizhar. 2013. Current trends on lean management–A review. *International Journal of lean thinking* 4 (2):15-21.

Behrouzi, F., and K. Y. Wong. 2011. Lean performance evaluation of manufacturing systems: A dynamic and innovative approach. *Procedia Computer Science* 3:388-395.

Bell, P. C., and R. M. O'keefe. 1987. Visual interactive simulation—history, recent developments, and major issues. *SIMULATION* 49 (3):109-116.

Bernroider, E. W., C. W. Wong, and K.-h. Lai. 2014. From dynamic capabilities to ERP enabled business improvements: The mediating effect of the implementation project. *International Journal of Project Management* 32 (2):350-362.

Bhamu, J., and K. Singh Sangwan. 2014. Lean manufacturing: literature review and research issues. *International Journal of Operations & Production Management* 34 (7):876-940.

Bhasin, S. 2008. Lean and performance measurement. *Journal of Manufacturing Technology Management* 19 (5):670-684.

———. 2012a. An appropriate change strategy for lean success. *Management Decision* 50 (3):439-458.

———. 2012b. Performance of Lean in large organisations. *Journal of Manufacturing Systems* 31 (3):349-357.

———. 2015. *Lean management beyond manufacturing*. Vol. 10: Springer.

Bhasin, S., and P. Burcher. 2006. Lean viewed as a philosophy. *Journal of Manufacturing Technology Management* 17 (1):56-72.

Bicheno, J. 2008. *The lean toolbox for service systems*: PICSIE books.

Blank, S. 2013. Why the lean start-up changes everything. *Harvard business review* 91 (5):63-72.

Bonavia, T., and J. A. Marin. 2006. An empirical study of lean production in the ceramic tile industry in Spain. *International Journal of Operations & Production Management* 26 (5):505-531.

Bortolini, M., F. G. Galizia, and C. Mora. 2018. Reconfigurable manufacturing systems: Literature review and research trend. *Journal of Manufacturing Systems* 49:93-106.
Bortolotti, T., S. Boscari, and P. Danese. 2015. Successful lean implementation: Organizational culture and soft lean practices. *International Journal of Production Economics* 160:182-201.
Bouanan, Y., S. Gorecki, J. Ribault, G. Zacharewicz, and N. Perry. 2018. Including in HLA federation functional mockup units for supporting interoperability and reusability in distributed simulation. Paper read at Proceedings of the 50th Computer Simulation Conference.
Bozdogan, K. 2010. Towards an integration of the lean enterprise system, total quality management system, six sigma and related enterprise process improvement methods: Massachusetts, MIT.
Božičković, R., M. Radosevic, I. Ćosić, M. Soković, and A. Rikalović. 2012. Integration of simulation and lean tools in effective production systems–Case study. *Strojniški vestnik-Journal of Mechanical Engineering* 58 (11):642-652.
Brunner, D. T., and R. C. Crain. 1991. GPSS/H in the 1990s. Paper read at 1991 Winter Simulation Conference Proceedings., 8-11 Dec. 1991.
Bruun, P., and R. N. Mefford. 2004. Lean production and the Internet. *International Journal of Production Economics* 89 (3):247-260.
Büyüközkan, G., G. Kayakutlu, and İ. S. Karakadılar. 2015. Assessment of lean manufacturing effect on business performance using Bayesian Belief Networks. *Expert Systems with Applications* 42 (19):6539-6551.
Calia, R. C., F. M. Guerrini, and M. de Castro. 2009. The impact of six sigma in the performance of a pollution prevention program. *Journal of Cleaner Production* 17 (15):1303-1310.
Camacho-Miñano, M.-d.-M., J. Moyano-Fuentes, and M. Sacristan-Diaz. 2013. What can we learn from the evolution of research on lean management assessment? *International Journal of Production Research* 51 (4):1098-1116.
Camagu, S. 2010. Investigating factors that negatively influence lean implementation in the eastern cape automotive industry, Nelson Mandela Metropolitan University.
Capraro, M., and G. Baglin. 2002. *L'entreprise étendue et le développement des fournisseurs*: Presses Universitaires Lyon.
Chae, B., and D. L. Olson. 2013. Business analytics for supply chain: A dynamic-capabilities framework. *International Journal of Information Technology & Decision Making* 12 (01):9-26.
Chapman, C. D. 2005. Clean house with lean 5S. *Quality progress* 38 (6):27-32.
Chatfield, D. C., T. P. Harrison, and J. C. Hayya. 2006. SISCO: An object-oriented supply chain simulation system. *Decision Support Systems* 42 (1):422-434.
Chaudron, J.-B. 2012. Architecture de simulation distribuée temps-réel, Toulouse, ISAE.
Chen, G., and D. Kotz. 2000. A survey of context-aware mobile computing research. *Dartmouth Computer Science Technical Report TR2000-381*.
Chen, Z., and K. Hua Tan. 2013. The impact of organization ownership structure on JIT implementation and production operations performance. *International Journal of Operations & Production Management* 33 (9):1202-1229.
Cherrafi, A., S. Elfezazi, A. Chiarini, A. Mokhlis, and K. Benhida. 2016. The integration of lean manufacturing, Six Sigma and sustainability: A literature review and future research directions for developing a specific model. *Journal of Cleaner Production* 139:828-846.
Chiarini, A. 2011. Integrating lean thinking into ISO 9001: a first guideline. *International Journal of Lean Six Sigma* 2 (2):96-117.
———. 2014a. A comparison between time-driven activity-based costing and value stream accounting in a lean Six Sigma manufacturing case study. *International Journal of Productivity and Quality Management* 14 (2):131-148.

———. 2014b. Sustainable manufacturing-greening processes using specific Lean Production tools: an empirical observation from European motorcycle component manufacturers. *Journal of Cleaner Production* 85:226-233.

Chin, L., and B. A. Rafuse. 1993. A small manufacturer adds JIT techniques to MRP. *Production and Inventory Management Journal* 34 (4):18.

Chong, M. Y., J. Prakash, S. L. Ng, R. Ramli, and J. F. Chin. 2013. PARALLEL KANBAN-CONWIP SYSTEM FOR BATCH PRODUCTION IN ELECTRONICS ASSEMBLY. *International Journal of Industrial Engineering* 20.

Coimbra, E. A., K. Institute, and K. I. Staff. 2009. *Total Management Flow: Achieving Excellence with Kaizen and Lean Supply Chains*: Kaizen Institute.

Corbett, C. J., and R. D. Klassen. 2006. Extending the horizons: environmental excellence as key to improving operations. *Manufacturing & Service Operations Management* 8 (1):5-22.

Cortes, H., J. Daaboul, J. Le Duigo, and B. Eynard. 2015. Intégration d'indicateurs Lean dans la stratégie d'entreprise pour le pilotage des performances.

Crema, M., and C. Verbano. 2015. How to combine lean and safety management in health care processes: A case from Spain. *Safety Science* 79:63-71.

Cua, K. O., K. E. McKone, and R. G. Schroeder. 2001. Relationships between implementation of TQM, JIT, and TPM and manufacturing performance. *Journal of Operations Management* 19 (6):675-694.

Dahlgaard, J. J., and S. Mi Dahlgaard-Park. 2006. Lean production, six sigma quality, TQM and company culture. *The TQM Magazine* 18 (3):263-281.

Demeter, K., and Z. Matyusz. 2011. The impact of lean practices on inventory turnover. *International Journal of Production Economics* 133 (1):154-163.

Demirbag, M., E. Tatoglu, K. W. Glaister, and S. Zaim. 2010. Measuring strategic decision making efficiency in different country contexts: A comparison of British and Turkish firms. *Omega* 38 (1-2):95-104.

Dennis, P. 2002. Lean Production Simplified, Productivity. *New York*.

Deshmukh, S., N. Upadhye, and S. Garg. 2010. Lean manufacturing for sustainable development. *Glob. Bus. Manag. Res. Int. J* 2 (1):125.

Detty, R. B., and J. C. Yingling. 2000. Quantifying benefits of conversion to lean manufacturing with discrete event simulation: A case study. *International Journal of Production Research* 38 (2):429-445.

Dey, A. 2001. Understanding and Using Context, Personal and Ubiquitous Computing, Vol. 5.

Dickson, E. W., S. Singh, D. S. Cheung, C. C. Wyatt, and A. S. Nugent. 2009. Application of lean manufacturing techniques in the emergency department. *The Journal of emergency medicine* 37 (2):177-182.

Diego Fernando, M. D., and L. Rivera Cadavid. 2007. Lean manufacturing measurement: the relationship between lean activities and lean metrics. *Estudios gerenciales* 23 (105):69-83.

Dimitrov, D., and M. Saxer. 2012. Productivity improvement in tooling manufacture through high speed 5 axis machining. *Procedia CIRP* 1:277-282.

Dora, M., and X. Gellynck. 2015. House of lean for food processing SMEs. *Trends in Food Science & Technology* 44 (2):272-281.

Drohomeretski, E., S. E. Gouvea da Costa, E. Pinheiro de Lima, and P. A. d. R. Garbuio. 2014. Lean, Six Sigma and Lean Six Sigma: an analysis based on operations strategy. *International Journal of Production Research* 52 (3):804-824.

Eaidgah, Y., A. A. Maki, K. Kurczewski, and A. Abdekhodaee. 2016. Visual management, performance management and continuous improvement: a lean manufacturing approach. *International Journal of Lean Six Sigma* 7 (2):187-210.

Ecology, W. S. D. o. 2007. Lean and Environmental Pilot Project Case Study: Lasco Bathware.

EPA. *Lean Thinking and Methods - 5S* 2017 [cited. Available from https://www.epa.gov/lean/lean-thinking-and-methods-5s.

Fadly Habidin, N., and S. r. Mohd Yusof. 2013. Critical success factors of Lean Six Sigma for the Malaysian automotive industry. *International Journal of Lean Six Sigma* 4 (1):60-82.

Falcone, A., A. Garro, A. DxAmbrogio, and A. Giglio. 2018. Using BPMN and HLA for SoS engineering: lessons learned and future directions. *2018 IEEE International Systems Engineering Symposium (ISSE)*:1-8.

Ferber, J. 1997. La modélisation multi-agent: un outil d'aide à l'analyse de phénomènes complexes. *Blasco F.: Tendances nouvelles en modélisation pour l'environnement, Elsevier*:113-133.

Ferdowsi, B., and A. Haggerty. 2002a. Lean Effects on Aerospace Programs (LEAP) Project: 737 Fuselage Case Study Report.

Ferdowsi, B., and A. Stanke. 2002b. Lean Effects on Aerospace Programs (LEAP) Project: F-16 Case Study Report.

Fernando, M. D. D., and L. R. Cadavid. 2007. Lean manufacturing measurement: the relationship between lean activities and lean metrics. *Estudios gerenciales* 23 (105):69-83.

Fisher, R. A. 1918. 009: The Correlation Between Relatives on the Supposition of Mendelian Inheritance.

Fishman, G. S. 2013. *Discrete-event simulation: modeling, programming, and analysis*: Springer Science & Business Media.

Fishwick, P. A. 1997. Computer simulation: growth through extension. *Transactions of the Society for Computer Simulation* 14 (1):13-24.

Fliedner, G. 2008. Sustainability: a new lean principle. Paper read at Proceedings of the 39th annual meeting of the decision sciences institute, Baltimore, Maryland.

Foures, D. 2015. Validation de modèles de simulation, Université de Toulouse, Université Toulouse III-Paul Sabatier.

Franco-Santos, M., M. Kennerley, P. Micheli, V. Martinez, S. Mason, B. Marr, D. Gray, and A. Neely. 2007. Towards a definition of a business performance measurement system. *International Journal of Operations & Production Management* 27 (8):784-801.

Frein, Y. 1998. Evaluation de performances pour la conception de flux. *Université d'été du pôle productique Rhône Alpes*.

Fujimoto, R. M. 1990. Parallel discrete event simulation. *Commun. ACM* 33 (10):30-53.

Fullerton, R. R., F. A. Kennedy, and S. K. Widener. 2014. Lean manufacturing and firm performance: The incremental contribution of lean management accounting practices. *Journal of Operations Management* 32 (7):414-428.

Fullerton, R. R., and W. F. Wempe. 2009. Lean manufacturing, non-financial performance measures, and financial performance. *International Journal of Operations & Production Management* 29 (3):214-240.

Garrett Jr, R. P., and J. G. Covin. 2015. Internal corporate venture operations independence and performance: A knowledge–based perspective. *Entrepreneurship Theory and Practice* 39 (4):763-790.

Garro, A., and A. Falcone. 2015. On the integration of HLA and FMI for supporting interoperability and reusability in distributed simulation. Paper read at Proceedings of the Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium.

Garvin, D. A. 1993. Manufacturing strategic planning. *California Management Review* 35 (4):85-106.

Garza-Reyes, J. A., G. W. Jacques, M. K. Lim, V. Kumar, and L. Rocha-Lona. 2014. Lean and green–synergies, differences, limitations, and the need for Six Sigma. Paper read at IFIP International Conference on Advances in Production Management Systems.

Gest, G., S. J. Culley, R. I. McIntosh, A. R. Mileham, and G. W. Owen. 1995. Review of fast tool change systems. *Computer Integrated Manufacturing Systems* 8 (3):205-210.

Ghattas, J., P. Soffer, and M. Peleg. 2014. Improving business process decision making based on past experience. *Decision Support Systems* 59:93-107.

Ghosh, M. 2013. Lean manufacturing performance in Indian manufacturing plants. *Journal of Manufacturing Technology Management*.

Goldsman, D., R. E. Nance, and J. R. Wilson. 2010. A brief history of simulation revisited. In *Proceedings of the Winter Simulation Conference*. Baltimore, Maryland: Winter Simulation Conference, 567-574.

Greenhalgh, T. 1997. How to read a paper: Statistics for the non-statistician. II: "Significant" relations and their pitfalls. *BMJ* 315 (7105):422-425.

Greinacher, S., E. Moser, J. Freier, J. Müller, and G. Lanza. 2016. Simulation-based Methodology for the Application of Lean and Green Strategies Depending on External Change Driver Influence. *Procedia CIRP* 48:242-247.

Grief, M. 1995. The visual factory: hiroyuki hirano: Portland OR: Productivity Press.

Gurumurthy, A., and R. Kodali. 2009. Application of benchmarking for assessing the lean manufacturing implementation. *Benchmarking: An International Journal* 16 (2):274-308.

———. 2011. Design of lean manufacturing systems using value stream mapping with simulation: a case study. *Journal of Manufacturing Technology Management* 22 (4):444-473.

Hasan, M. A., R. Shankar, and J. Sarkis. 2007. A study of barriers to agile manufacturing. *International Journal of Agile Systems and Management* 2 (1):1-22.

Henao, R., W. Sarache, and I. Gómez. 2018. Lean manufacturing and sustainable performance: Trends and future challenges. *Journal of Cleaner Production*.

Henderson, B. A. 2003. *Lean transformation: how to change your business into a lean enterprise*: Oaklea Press.

Herrmann, C., S. Thiede, J. Stehr, and L. Bergmann. 2008. An environmental perspective on Lean Production. In *Manufacturing Systems and Technologies for the New Frontier*: Springer, 83-88.

Herron, C., and C. Hicks. 2008. The transfer of selected lean manufacturing techniques from Japanese automotive manufacturing into general manufacturing (UK) through change agents. *Robotics and Computer-Integrated Manufacturing* 24 (4):524-531.

Hibadullah, S., N. Habidin, N. Fuzi, and A. Desa. 2014. Critical success factors of lean manufacturing practices for the Malaysian automotive manufacturers. *International Journal of Quality and Innovation* 2 (3-4):256-271.

Hines, P., P. Found, G. Griffiths, and R. Harrison. 2008. Staying Lean; Thriving, not Just Surviving, Lean Enterprise Research Centre, Cardiff University, Cardiff.

Hines, P., M. Holweg, and N. Rich. 2004. Learning to evolve: A review of contemporary lean thinking. *International Journal of Operations & Production Management* 24 (10):994-1011.

Ho, C.-F., Y.-M. Tai, Y.-M. Tai, and Y.-P. Chi. 2005. A structural approach to measuring uncertainty in supply chains. *International Journal of Electronic Commerce* 9 (3):91-114.

Hodge, G. L., K. Goforth Ross, J. A. Joines, and K. Thoney. 2011. Adapting lean manufacturing principles to the textile industry. *Production Planning & Control* 22 (3):237-247.

Hoekstra, S., J. Romme, and S. Argelo. 1992. *Integral logistic structures: developing customer-oriented goods flow*: McGraw-Hill Book Co Ltd.

Hofer, C., C. Eroglu, and A. R. Hofer. 2012. The effect of lean production on financial performance: The mediating role of inventory leanness. *International Journal of Production Economics* 138 (2):242-253.

Hollocks, B. W. 2006. Forty years of discrete-event simulation—a personal reflection. *Journal of the Operational Research Society* 57 (12):1383-1399.

Hong, P., W. J. Doll, E. Revilla, and A. Y. Nahm. 2011. Knowledge sharing and strategic fit in integrated product development proejcts: An empirical study. *International Journal of Production Economics* 132 (2):186-196.

Hu, Q., R. Mason, S. J. Williams, and P. Found. 2015. Lean implementation within SMEs: a literature review. *Journal of Manufacturing Technology Management* 26 (7):980-1012.
Ingemansson, A., and G. S. Bolmsjö. 2004. Improved efficiency with production disturbance reduction in manufacturing systems based on discrete-event simulation. *Journal of Manufacturing Technology Management* 15 (3):267-279.
JaamSim: Discrete-Event Simulation Software Version 2019-01.
Jagdev, H., and J. Browne. 1998. The extended enterprise-a context for manufacturing. *Production Planning & Control* 9 (3):216-229.
Jain, S., N. F. Choong, and W. Lee. 2002. Manufacturing supply chain applications: modeling computer assembly operations for supply chain integration. Paper read at Proceedings of the 34th conference on Winter simulation: exploring new frontiers.
Jasti, N. V. K., and R. Kodali. 2015. Lean production: literature review and trends. *International Journal of Production Research* 53 (3):867-885.
Jeon, S. M., and G. Kim. 2016. A survey of simulation modeling techniques in production planning and control (PPC). *Production Planning & Control* 27 (5):360-377.
Jeyaraman, K., and L. Kee Teo. 2010. A conceptual framework for critical success factors of lean Six Sigma. *International Journal of Lean Six Sigma* 1 (3):191-215.
Jones, D. 2009. The real meaning of lean. *Plant Graphics* 59 (6):24-24.
Julie Yazici, H. 2005. Influence of flexibilities on manufacturing cells for faster delivery using simulation. *Journal of Manufacturing Technology Management* 16 (8):825-841.
Kadry, S. 2013. Six sigma methodology for the environment sustainable development. In *Mechanism Design for Sustainability*: Springer, 61-76.
Karam, A.-A., M. Liviu, V. Cristina, and H. Radu. 2018. The contribution of lean manufacturing tools to changeover time decrease in the pharmaceutical industry. A SMED project. *Procedia Manufacturing* 22:886-892.
Karlsson, C., and P. Åhlström. 1996. Assessing changes towards lean production. *International Journal of Operations & Production Management* 16 (2):24-41.
Kaye, M., and R. Anderson. 1999. Continuous improvement: The ten essential criteria. *International Journal of Quality & Reliability Management* 16 (5):485-509.
Kelton, W., R. Sadowski, and D. Sturrock. 2007. Simulation with Arena. New York: McGrawHill: Inc.
Khanchanapong, T., D. Prajogo, A. S. Sohal, B. K. Cooper, A. C. Yeung, and T. C. E. Cheng. 2014. The unique and complementary effects of manufacturing technologies and lean practices on manufacturing operational performance. *International Journal of Production Economics* 153:191-203.
Kilpatrick, J. 2003. Lean principles. *Utah Manufacturing Extension Partnership* 68:1-5.
King, A. A., and M. J. Lenox. 2001. Lean and green? An empirical examination of the relationship between lean production and environmental performance. *Production and operations management* 10 (3):244-256.
King, D. H., and H. S. Harrison. 2013. "JaamSim" open-source simulation software. In *Proceedings of the 2013 Grand Challenges on Modeling and Simulation Conference*. Toronto, Ontario, Canada: Society for Modeling & Simulation International, 1-6.
Kochan, T. A., R. D. Lansbury, and J. P. MacDuffie. 1997. *After lean production: Evolving employment practices in the world auto industry*: Cornell University Press.
Kojima, S., and R. Kaplinsky. 2004. The use of a lean production index in explaining the transition to global competitiveness: the auto components sector in South Africa. *Technovation* 24 (3):199-206.
Kováčová, L. 2013. The integration of lean management and sustainability. *Transfer inovácií* 26:195-199.
Krafcik, J. F. 1988. Triumph of the lean production system. *Mit Sloan Management Review* 30 (1):41.
Kumar, B. S., and S. S. Abuthakeer. 2012. Implementation of lean tools and techniques in an automotive industry. *Journal of Applied Sciences(Faisalabad)* 12 (10):1032-3037.

Kumar, M., and J. Antony. 2008. Comparing the quality management practices in UK SMEs. *Industrial Management & Data Systems* 108 (9):1153-1166.
Kumar, M., J. Antony, R. Singh, M. Tiwari, and D. Perry. 2006. Implementing the Lean Sigma framework in an Indian SME: a case study. *Production Planning and Control* 17 (4):407-423.
Kumar, M., J. Antony, and M. K. Tiwari. 2011. Six Sigma implementation framework for SMEs – a roadmap to manage and sustain the change. *International Journal of Production Research* 49 (18):5449-5467.
Kundu, G., and B. M. Manohar. 2012. Critical success factors for implementing lean practices in it support services. *International Journal for Quality Research* 6 (4):301-312.
Lande, M., R. L. Shrivastava, and D. Seth. 2016. Critical success factors for Lean Six Sigma in SMEs (small and medium enterprises). *The TQM Journal* 28 (4):613-635.
Lander, E., and J. K. Liker. 2007. The Toyota Production System and art: making highly customized and creative products the Toyota way. *International Journal of Production Research* 45 (16):3681-3698.
Langenwalter, G. 2006. Life" is Our Ultimate Customer: From Lean to Sustainability. *Target* 22 (1):5-15.
Laureani, A., and J. Antony. 2012. Critical success factors for the effective implementation of Lean Sigma. *International Journal of Lean Six Sigma* 3 (4):274-283.
———. 2016. Leadership – a critical success factor for the effective implementation of Lean Six Sigma. *Total Quality Management & Business Excellence*:1-22.
Le Fur, J. 1994. Dynamique du système Pêche Artisanale et intelligence artqicielle: L e Prqjet Mo PA.
Le Moigne, J. 1990. La modelisation des systemes complexes/Le Moigne Jean-Louis: Paris.
Leandro-Elizondo, R. 2018. Méthodologie pour l'évaluation de la performance de l'amélioration continue des processus industriels. PhD Thesis, Institut National Polytechnique de Toulouse.
Lee, Q. 2007. Implementing lean manufacturing. *Management services* 51 (3):14.
Lehtinen, U., and M. Torkko. 2005. The lean concept in the food industry: A case study of contract a manufacturer. *Journal of Food Distribution Research* 36 (856-2016-56436):57-67.
Leong, G. K., D. L. Snyder, and P. T. Ward. 1990. Research in the process and content of manufacturing strategy. *Omega* 18 (2):109-122.
Lian, Y.-H., and H. Van Landeghem. 2007. Analysing the effects of Lean manufacturing using a value stream mapping-based simulation generator. *International Journal of Production Research* 45 (13):3037-3058.
Liker, J., and M. Rother. 2011. Why lean programs fail. *Lean Enterprise Institute*:45-79.
Liker, J. K. 1997. *Becoming lean: Inside stories of US manufacturers*: CRC Press.
Liker, J. K. 2004. The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. *McGraw-Hill, New York, NY.*
Liker, J. K. 2016. *O modelo Toyota: 14 princípios de gestão do maior fabricante do mundo*: Bookman Editora.
Long, Q. 2014. Distributed supply chain network modelling and simulation: integration of agent-based distributed simulation and improved SCOR model. *International Journal of Production Research* 52 (23):6899-6917.
Longoni, A., and R. Cagliano. 2011. How to Align Lean Manufacturing and Sustainability: the Role of Organisational Responsibility and Worker Commitment.
Longoni, A., R. Golini, and R. Cagliano. 2014. The role of New Forms of Work Organization in developing sustainability strategies in operations. *International Journal of Production Economics* 147:147-160.
Lyonnet, B. 2010. Amélioration de la performance industrielle: vers un système de production Lean adapté aux entreprises du pôle de compétitivité Arve Industries Haute-Savoie Mont-Blanc.

Lyons, A. C., K. Vidamour, R. Jain, and M. Sutherland. 2013. Developing an understanding of lean thinking in process industries. *Production Planning & Control* 24 (6):475-494.

Mackelprang, A. W., and A. Nair. 2010. Relationship between just-in-time manufacturing practices and performance: A meta-analytic investigation. *Journal of Operations Management* 28 (4):283-302.

Malhotra, M. K., C. Singhal, G. Shang, and R. E. Ployhart. 2014. A critical evaluation of alternative methods and paradigms for conducting mediation analysis in operations management research. *Journal of Operations Management* 32 (4):127-137.

Mali, Y. R., and K. Inamdar. 2012. Changeover time reduction using SMED technique of lean manufacturing. *International Journal of Engineering Research and Applications* 2 (3):2441-2445.

Mamat, R. C., B. M. Deros, M. N. Ab Rahman, M. K. Omar, and S. Abdullah. 2015. Soft lean practices for successful lean production system implementation in Malaysia automotive Smes: a proposed framework. *Jurnal teknologi* 77 (27).

Mann, R., and D. Kehoe. 1995. Factors affecting the implementation and success of TQM. *International Journal of Quality & Reliability Management* 12 (1):11-23.

Marodin, G. A., and T. A. Saurin. 2013. Implementing lean production systems: research areas and opportunities for future studies. *International Journal of Production Research* 51 (22):6663-6680.

Marr, B. 2012. *Key Performance Indicators (KPI): The 75 measures every manager needs to know*: Pearson UK.

Marshall, D. 2015. Classifying lean production: a review of empirical research. *International Journal of Supply Chain Management* 4 (4):71-81.

Martinez Sànchez, A., and M. Pérez Pérez. 2001. Lean indicators and manufacturing strategies. *International Journal of Operations & Production Management* 21 (11):1433-1452.

Martínez-Jurado, P. J., and J. Moyano-Fuentes. 2014. Key determinants of lean production adoption: evidence from the aerospace sector. *Production Planning & Control* 25 (4):332-345.

Marudhamuthu, R., and M. Krishnaswamy. 2011. The development of green environment through lean implementation in a garment industry. *Journal of Engineering and Applied Sciences* 6 (9):104-111.

Maskell, B. H., and R. Pojasek. 2008. For lean to be green the performance measurements must change. *BMA, Hanover*.

Matsui, Y. 2007. An empirical analysis of just-in-time production in Japanese manufacturing companies. *International Journal of Production Economics* 108 (1):153-164.

McDonald, T., K. P. Ellis, E. M. Van Aken, and C. Patrick Koelling. 2009. Development and application of a worker assignment model to evaluate a lean manufacturing cell. *International Journal of Production Research* 47 (9):2427-2447.

McKone, K. E., R. G. Schroeder, and K. O. Cua. 2001. The impact of total productive maintenance practices on manufacturing performance. *Journal of Operations Management* 19 (1):39-58.

Melton, T. 2005. The Benefits of Lean Manufacturing: What Lean Thinking has to Offer the Process Industries. *Chemical Engineering Research and Design* 83 (6):662-673.

Melvin, A., and D. Baglee. 2008. Value stream mapping: A dairy industry prospective. Paper read at 2008 IEEE International Engineering Management Conference.

Mezgebe, T. T., H. B. El Haouzi, G. Demesure, and A. Thomas. 2018. A negotiation-based control approach for disturbed industrial context. *IFAC-PapersOnLine* 51 (11):1255-1260.

Miller, D. C., and J. A. Thorpe. 1995. SIMNET: the advent of simulator networking. *Proceedings of the IEEE* 83 (8):1114-1123.

Miller, G., J. Pawloski, and C. R. Standridge. 2010. A case study of lean, sustainable manufacturing. *Journal of Industrial Engineering and Management (JIEM)* 3 (1):11-32.

Mishra, R. P., G. Anand, and R. Kodali. 2006. Development of a framework for world-class maintenance systems. *Journal of Advanced Manufacturing Systems* 5 (02):141-165.

Misra, J. 1986. Distributed discrete-event simulation. *ACM Computing Surveys (CSUR)* 18 (1):39-65.

Mor, R. S., S. Singh, and A. Bhardwaj. 2016. Learning on lean production: A review of opinion and research within environmental constraints. *Operations and Supply Chain Management: An International Journal* 9 (1):61-72.

Moreira, F., A. Alves, and R. Sousa. 2010. Towards Eco–efficient Lean Production Systems. Balanced Automation Systems for Future Manufacturing Networks: Berlin, Heidelberg: Springer.

Moyano-Fuentes, J., and M. Sacristán-Díaz. 2012. Learning on lean: a review of thinking and research. *International Journal of Operations & Production Management* 32 (5):551-582.

Myers, F., and F. Stewart. 2002. Motion and time study for lean manufacturing: Prentice Hall, Upper Saddle River, New Jersey, USA.

Nahmens, I. 2009. From lean to green construction: A natural extension. Paper read at Construction Research Congress 2009: Building a Sustainable Future.

Neema, H., J. Gohl, Z. Lattmann, J. Sztipanovits, G. Karsai, S. Neema, T. Bapty, J. Batteh, H. Tummescheit, and C. Sureshkumar. 2014. Model-based integration platform for FMI co-simulation and heterogeneous simulations of cyber-physical systems. Paper read at Proceedings of the 10 th International Modelica Conference; March 10-12; 2014; Lund; Sweden.

Netland, T., and K. Ferdows. 2014. What to expect from corporate lean programs. *Mit Sloan Management Review* 55 (4):83.

Netland, T. H. 2016. Critical success factors for implementing lean production: the effect of contingencies. *International Journal of Production Research* 54 (8):2433-2448.

Ng, R., J. S. C. Low, and B. Song. 2015. Integrating and implementing Lean and Green practices based on proposition of Carbon-Value Efficiency metric. *Journal of Cleaner Production* 95:242-255.

Nudurupati, S. S., U. S. Bititci, V. Kumar, and F. T. Chan. 2011. State of the art literature review on performance measurement. *Computers & industrial engineering* 60 (2):279-290.

O'Neill, P., and A. S. Sohal. 1999. Business Process Reengineering A review of recent literature. *Technovation* 19 (9):571-581.

Ohno, T. 1988. *Toyota production system: beyond large-scale production*: crc Press.

Okoshi, C. Y., E. P. de Lima, and S. E. G. Da Costa. 2019. Performance cause and effect studies: Analyzing high performance manufacturing companies. *International Journal of Production Economics* 210:27-41.

Olhager, J. 2010. The role of the customer order decoupling point in production and supply chain management. *Computers in Industry* 61 (9):863-868.

Omogbai, O., and K. Salonitis. 2017. The Implementation of 5S Lean Tool Using System Dynamics Approach. *Procedia CIRP* 60:380-385.

Ören, T. I., and B. P. Zeigler. 1979. Concepts for advanced simulation methodologies. *SIMULATION* 32 (3):69-82.

Paez, O., J. Dewees, A. Genaidy, S. Tuncel, W. Karwowski, and J. Zurada. 2004. The lean manufacturing enterprise: An emerging sociotechnological system integration. *Human Factors and Ergonomics in Manufacturing & Service Industries* 14 (3):285-306.

Pampanelli, A., P. Found, and A. M. Bernardes. 2011. A lean and green Kaizen model. Paper read at POMS annual conference, Reno, Nevada, USA.

Pampanelli, A. B., P. Found, and A. M. Bernardes. 2014. A Lean & Green Model for a production cell. *Journal of Cleaner Production* 85:19-30.

Pande, P. S., and L. Holpp. 2001. *What is six sigma?*: McGraw-Hill Professional.

Panwar, A., B. P. Nepal, R. Jain, and A. P. S. Rathore. 2015. On the adoption of lean manufacturing principles in process industries. *Production Planning & Control* 26 (7):564-587.

Park, C., and D. Linich. 2008. Green Lean Six Sigma: using Lean to help drive results in the wholly sustainable enterprise. *Deloitte Consulting, New York*.

Parry, G. C., and C. E. Turner. 2006. Application of lean visual process management tools. *Production Planning & Control* 17 (1):77-86.

Pay, R. 2008. Everybody's Jumping on the Lean Bandwagon, but Many Are Being Taken for a Ride. *Industry week*.

Pepper, M. P., and T. A. Spedding. 2010. The evolution of lean Six Sigma. *International Journal of Quality & Reliability Management* 27 (2):138-155.

Perera, C., A. Zaslavsky, P. Christen, and D. Georgakopoulos. 2013. Context aware computing for the internet of things: A survey. *IEEE communications surveys & tutorials* 16 (1):414-454.

Plonka, F. E. 1997. Developing a lean and agile work force. *Human Factors and Ergonomics in Manufacturing & Service Industries* 7 (1):11-20.

Pojasek, R. B. 1999. Quality toolbox: Poka-yoke and zero waste. *Environmental Quality Management* 9 (2):91-97.

Pool, A., J. Wijngaard, and D.-J. Van der Zee. 2011. Lean planning in the semi-process industry, a case study. *International Journal of Production Economics* 131 (1):194-203.

Possik, J., A. D'Ambrogio, G. Zacharewicz, A. Amrani, and B. Vallespir. 2019. A BPMN/HLA-Based Methodology for Collaborative Distributed DES. In *28th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE-2019)*. Capri, Italy.

Possik, J. J., A. A. Amrani, and G. Zacharewicz. 2018. Development of a co-simulation system as a decision-aid in Lean tools implementation. In *Proceedings of the 50th Computer Simulation Conference*. Bordeaux, France: Society for Computer Simulation International, 1-12.

PQA. 2003. Practical framework for implementation of Six Sigma in SMEs. *Process Quality Associates, available at: www.pqa.net/sixsigma/ (accessed March 7, 2005)*.

Prajogo, D., M. Chowdhury, A. C. Yeung, and T. C. E. Cheng. 2012. The relationship between supplier management and firm's operational performance: A multi-dimensional perspective. *International Journal of Production Economics* 136 (1):123-130.

Prakash, J., and J. F. Chin. 2017. Effects of inventory classifications on CONWIP system: a case study. *Journal of Management Analytics* 4 (3):296-320.

R. Jadhav, J., S. S. Mantha, and S. B. Rane. 2014. Exploring barriers in lean implementation. *International Journal of Lean Six Sigma* 5 (2):122-148.

Radam, A., M. L. Abu, and A. M. Abdullah. 2008. Technical efficiency of small and medium enterprise in Malaysia: A stochastic frontier production model. *International Journal of economics and Management* 2 (2):395-408.

Radnor, Z. 2000. Changing to a lean organisation: the case of a chemicals company. *International Journal of Manufacturing Technology and Management* 1 (4-5):444-454.

Radnor, Z., and P. Walley. 2008. Learning to walk before we try to run: Adapting lean for the public sector. *Public Money and Management* 28 (1):13-20.

Radnor, Z., P. Walley, A. Stephens, and G. Bucci. 2006. Evaluation of the lean approach to business management and its use in the public sector. *Scottish executive social research* 20.

Radziwill, N. 2013. Leading and managing the lean management process. *The Quality Management Journal* 20 (2):61.

Rahman, N. A. A., S. M. Sharif, and M. M. Esa. 2013. Lean Manufacturing Case Study with Kanban System Implementation. *Procedia Economics and Finance* 7:174-180.

Rahman, S.-u. 1998. Theory of constraints: a review of the philosophy and its applications. *International Journal of Operations & Production Management* 18 (4):336-355.

Robinson, S. 2005. Discrete-event simulation: From the pioneers to the present, what next? *Journal of The Operational Research Society - J OPER RES SOC* 56:619-629.

Rosenberger, P., and D. Gerhard. 2018. Context-awareness in industrial applications: definition, classification and use case. *Procedia CIRP* 72:1172-1177.

Rota-Frantz, K., C. Thierry, and G. Bel. 2001. Gestion des flux dans les chaînes logistiques (Supply Chain Management). *Performances Industrielles et Gestion Des Flux, Hermes Science-Lavoisier Paris*:103-128.

Rothenberg, S., F. K. Pil, and J. Maxwell. 2001. Lean, green, and the quest for superior environmental performance. *Production and operations management* 10 (3):228-243.

Rother, M. 2009. Toyota KATA.

Rymaszewska, A. D. 2016. Rethinking the Applicability of Lean Philosophy. Doctoral Thesis.

Sargent, R. G. 2010. Verification and validation of simulation models. Paper read at Proceedings of the 2010 Winter Simulation Conference.

Saurin, T. A., and C. F. Ferreira. 2009. The impacts of lean production on working conditions: A case study of a harvester assembly line in Brazil. *International Journal of Industrial Ergonomics* 39 (2):403-412.

Saurin, T. A., J. L. D. Ribeiro, and G. Vidor. 2012. A framework for assessing poka-yoke devices. *Journal of Manufacturing Systems* 31 (3):358-366.

Scherrer-Rathje, M., T. A. Boyle, and P. Deflorin. 2009. Lean, take two! Reflections from the second attempt at lean implementation. *Business horizons* 52 (1):79-88.

Seppälä, P., and S. Klemola. 2004. How do employees perceive their organization and job when companies adopt principles of lean production? *Human Factors and Ergonomics in Manufacturing & Service Industries* 14 (2):157-180.

Serrano Lasa, I., R. d. Castro, and C. O. Laburu. 2009. Extent of the use of Lean concepts proposed for a value stream mapping application. *Production Planning & Control* 20 (1):82-98.

Shah, R., and P. T. Ward. 2003. Lean manufacturing: context, practice bundles, and performance. *Journal of Operations Management* 21 (2):129-149.

Shah, R., and P. T. Ward. 2007. Defining and developing measures of lean production. *Journal of Operations Management* 25 (4):785-805.

Shahidul, M., and S. Syed Shazali. 2011. Dynamics of manufacturing productivity: lesson learnt from labor intensive industries. *Journal of Manufacturing Technology Management* 22 (5):664-678.

Shakespeare, T. P., V. J. Gebski, M. J. Veness, and J. Simes. 2001. Improving interpretation of clinical studies by use of confidence levels, clinical significance curves, and risk-benefit contours. *The Lancet* 357 (9265):1349-1353.

Sharman, G. 1984. Rediscovery of logistics. *Harvard business review* 5:71-79.

Shingo, S., and A. P. Dillon. 1989. *A study of the Toyota production system: From an Industrial Engineering Viewpoint*: CRC Press.

Sievert, N. 2016. Modelica Models in a Distributed Environment Using FMI and HLA.

Simpson, D. F., and D. J. Power. 2005. Use the supply relationship to develop lean and green suppliers. *Supply chain management: An international Journal* 10 (1):60-68.

Singh, B., S. Garg, S. Sharma, and C. Grewal. 2010. Lean implementation and its benefits to production industry. *International Journal of Lean Six Sigma* 1 (2):157-168.

Slack, N. 1991. The Manufacturing Advantage (London: Mercury).

Smalley, A. 2004. *Creating level pull: a lean production-system improvement guide for production-control, operations, and engineering professionals*: Lean Enterprise Institute.

Smalley, A., and T. Harada. 2009. Lean lives on the floor. *Manufacturing Engineering* 142 (5):83-+.

Smart, P., D. Tranfield, P. Deasley, R. Levene, A. Rowe, and J. Corley. 2003. Integrating 'lean'and 'high reliability'thinking. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 217 (5):733-739.

Sobral, M. C., A. B. L. d. Sousa Jabbour, and C. J. Chiappetta Jabbour. 2013. Green benefits from adopting lean manufacturing: a case study from the automotive sector. *Environmental Quality Management* 22 (3):65-72.

Sohal, A. S. 1996. Developing a lean production organization: an Australian case study. *International Journal of Operations & Production Management* 16 (2):91-102.
Sohal, A. S., and A. Egglestone. 1994. Lean production: experience among Australian organizations. *International Journal of Operations & Production Management* 14 (11):35-51.
Soltero, C., and G. Waldrip. 2002. Using kaizen to reduce waste and prevent pollution. *Environmental Quality Management* 11 (3):23-38.
Sorooshian, S., and S. A. M. Ali. 2017. Lean Practices Pertaining Hard and Soft Factors in Service Sectors. *Calitatea* 18 (161):80-86.
Stentoft-Arlbjørn, J., P. Freytag, and T. Damgaard. 2008. The beauty of measurements. *European Business Review* 20 (2):112-127.
Stentoft-Arlbjørn, J., and P. Vagn Freytag. 2013. Evidence of lean: a review of international peer-reviewed journal articles. *European Business Review* 25 (2):174-205.
Sugimori, Y., K. Kusunoki, F. Cho, and S. Uchikawa. 1977. Toyota production system and Kanban system Materialization of just-in-time and respect-for-human system. *International Journal of Production Research* 15 (6):553-564.
Suri, R., and S. De Treville. 1986. Getting from "just-in-case" to "just-in-time": Insights from a simple model. *Journal of Operations Management* 6 (3-4):295-304.
Taggart, P. 2009. The effectiveness of lean manufacturing audits in driving improvements in operational performance.
Taggart, P., and F. Kienhofer. 2012. *The effectiveness of lean manufacturing audits in measuring operational performance improvements*. Vol. 24.
Tahir, A. C., and R. Darton. 2010. The process analysis method of selecting indicators to quantify the sustainability performance of a business operation. *Journal of Cleaner Production* 18 (16-17):1598-1607.
Tan, K. H., and K. Platts. 2003. Linking objectives to actions: A decision support approach based on cause–effect linkages. *Decision sciences* 34 (3):569-593.
Tan, W., Y. Chai, and Y. Liu. 2011. A message-driving formalism for modeling and simulation of multi-agent supply chain systems. *Journal of Systems Science and Systems Engineering* 20 (4):385-399.
Taner, M. T. 2013. Critical Success Factors for Six Sigma Implementation in Large-scale Turkish Construction Companies. *International Review of Management and Marketing* 3 (4):212-225.
Technologies, P. *Pitch pRTI* [cited March 2, 2019. Available from http://pitchtechnologies.com/products/prti/.
Tersine, R. J., and J. G. Wacker. 2000. Customer-aligned inventory strategies: agility maxims. *International Journal of Agile Management Systems* 2 (2):114-120.
Thomas, A. J., M. Francis, R. Fisher, and P. Byard. 2016. Implementing Lean Six Sigma to overcome the production challenges in an aerospace company. *Production Planning & Control* 27 (7-8):591-603.
Timans, W., J. Antony, K. Ahaus, and R. van Solingen. 2012. Implementation of Lean Six Sigma in small-and medium-sized manufacturing enterprises in the Netherlands. *Journal of the Operational Research Society* 63 (3):339-353.
Torielli, R., R. Abrahams, R. Smillie, and R. Voigt. 2011a. Using lean methodologies for economically and environmentally sustainable foundries. *China Foundry* 8 (1):74-88.
Torielli, R. M., R. A. Abrahams, R. W. Smillie, and R. C. Voigt. 2011b. Using lean methodologies for economically and environmentally sustainable foundries. *China Foundry* 8 (1):74-88.
Tremblay, M. A., and A. D. Dibb. 2003. Modelling and restoration of bighorn sheep habitat within and adjacent to Kootenay National Park, British Columbia. Paper read at Making ecosystem based management work: connecting managers and researchers. Proc. of Fifth International Conf. on Sci. and Manage. of Protected Areas.
Tukey, J. 1977. Exploratory data analysis. *Reading: Addison-Wesley*.

Ulutas, B. 2011. An application of SMED Methodology. *World academy of science, engineering and technology* 79:101.
Vais, A., V. Miron, M. Pedersen, and J. Folke. 2006. "Lean and Green" at a Romanian secondary tissue paper and board mill—putting theory into practice. *Resources, Conservation and Recycling* 46 (1):44-74.
Villarreal, B., J. A. Garza-Reyes, and V. Kumar. 2016. A lean thinking and simulation-based approach for the improvement of routing operations. *Industrial Management & Data Systems* 116 (5):903-925.
Vinodh, S., K. Arvind, and M. Somanaathan. 2011a. Tools and techniques for enabling sustainability through lean initiatives. *Clean Technologies and Environmental Policy* 13 (3):469-479.
Vinodh, S., K. R. Arvind, and M. Somanaathan. 2011b. Tools and techniques for enabling sustainability through lean initiatives. *Clean Technologies and Environmental Policy* 13 (3):469-479.
Vinodh, S., S. Gautham, and A. Ramiya R. 2011c. Implementing lean sigma framework in an Indian automotive valves manufacturing organisation: a case study. *Production Planning & Control* 22 (7):708-722.
Vlachos, I. 2015. Applying lean thinking in the food supply chains: a case study. *Production Planning & Control* 26 (16):1351-1367.
Voss, C. A. 2005. Paradigms of manufacturing strategy re-visited. *International Journal of Operations & Production Management* 25 (12):1223-1227.
Wan, H.-d., and F. Frank Chen. 2008. A leanness measure of manufacturing systems for quantifying impacts of lean initiatives. *International Journal of Production Research* 46 (23):6567-6584.
Wang, F.-K., and K.-S. Chen. 2012. Application of Lean Six Sigma to a panel equipment manufacturer. *Total Quality Management & Business Excellence* 23 (3-4):417-429.
Westgard, J. *Defect Rates, Quality and Productivity* 2019 [cited. Available from https://www.westgard.com/guest5.htm.
Wheelwright, S. C. 1978. Reflecting corporate strategy in manufacturing decisions. *Business horizons* 21 (1):57-66.
White, P., and R. Ingalls. 2009. *Introduction to Simulation*.
Wilson, A. 2010. Sustainable Manufacturing: Comparing Lean, Six Sigma, and Total Quality Manufacturing. *USA: Strategic Sustainability Consulting, Washington, DC*.
Womack, J. P., and D. T. Jones. 1996. *Lean thinking: banish waste and create wealth in your corporation*: Simon and Schuster.
———. 2003. *Lean thinking: banish waste and create wealth in your corporation*.
Womack, J. P., D. T. Jones, D. Roos, and M. I. o. Technology. 1990. *The machine that changed the world: based on the Massachusetts Institute of Technology 5-million dollar 5-year study on the future of the automobile*: Rawson Associates.
Yang, C. C., T. M. Yeh, and K. J. Yang. 2012. The implementation of technical practices and human factors of the toyota production system in different industries. *Human Factors and Ergonomics in Manufacturing & Service Industries* 22 (6):541-555.
Yoo, T., H. Cho, and E. Yücesan. 2010. Hybrid algorithm for discrete event simulation based supply chain optimization. *Expert Systems with Applications* 37 (3):2354-2361.
Youssef, J., G. Zacharewicz, D. Chen, and F. Vernadat. 2017. *EOS: enterprise operating systems*.
Yusuf, Y. Y., A. Gunasekaran, A. Musa, M. Dauda, N. M. El-Berishy, and S. Cang. 2014. A relational study of supply chain agility, competitiveness and business performance in the oil and gas industry. *International Journal of Production Economics* 147:531-543.
Yusuf, Y. Y., M. Sarhadi, and A. Gunasekaran. 1999. Agile manufacturing:: The drivers, concepts and attributes. *International Journal of Production Economics* 62 (1-2):33-43.
Zacharewicz, G. 2006. Un environnement G-DEVS/HLA: Application à la modélisation et simulation distribuée de workflow.

Zacharewicz, G., C. Frydman, and N. Giambiasi. 2006. Lookahead computation in G-DEVS/HLA environment.
———. 2008. G-DEVS/HLA environment for distributed simulations of workflows. *SIMULATION* 84 (5):197-213.
Zengin, A. 2011. Modeling discrete event scalable network systems. *Information Sciences* 181 (5):1028-1043.
Zengin, A., H. Sarjoughian, and H. Ekiz. 2013. Discrete event modeling of swarm intelligence based routing in network systems. *Information Sciences* 222:81-98.
Zhang, M., and A. Awasthi. 2014. Using Six Sigma to achieve sustianble manufacturing. Paper read at Proceedings of the 2014 International Conference on Innovative Design and Manufacturing (ICIDM).
Zhao, Y., Y. Li, S. H. Lee, and L. Bo Chen. 2011. Entrepreneurial orientation, organizational learning, and performance: evidence from China. *Entrepreneurship Theory and Practice* 35 (2):293-317.
Zhao, Z., and R. De Souza. 2000. Genetic production line-balancing for the hard disk drive industry. *The International Journal of Advanced Manufacturing Technology* 16 (4):297-302.

Appendix A

ANOVA using SPSS

In this section, we will see the steps to perform a One way ANOVA test using SPSS (See .

- Open SPSS application.
- Go to Analyze - Compare Means One Way ANOVA.
- A dialog box appears.
- A list of all the dependent variables measured appears.
- By using the upper arrow button, move the output variables into the "dependent list".
- By using the down arrow button, move the independent variable to the "factor" box.
- Click on the Post Hoc button to select the type of multiple comparison test (in our case, we selected the "Tukey" test)
- Click Continue and it will take you to the One way ANOVA dialog box
- User can also go to options then click on Means plot to get the ANOVA graph of the means.
- Click Continue
- Click Ok

SPSS output will appear with six sections:

- Descriptive section
- Test of Homogeneity of Variances
- ANOVA
- Multiple Comparisons
- Grade Point Average
- Graph

Run1: Neutral Scenario (no context or fluctuation)

Market Demand - no demand increase

168.6666667[h]	337	328	340	337
153.3333333[h]	324	342	330	339
168.6666667[h]	321	322	330	334
153.3333333[h]	322	320	338	320
145.6666667[h]	324	331	325	320
161.0000000[h]	324	326	334	331
168.6666667[h]	330	329	320	330
168.6666667[h]	325	331	335	328
153.3333333[h]	331	320	334	336
176.3333333[h]	322	329	333	328
161.0000000[h]	323	325	330	335
153.3333333[h]	320	334	331	342

Setup Time

2.2	2.2	2.2	2.2
1.5	1.5	1.5	1.5
1	1.8	2.5	2.8
0	0	0	0
0	0	0	0
0	0	0	0
1.2	1.5	2	2.1
0	0	0	0

Processing Time

8[min]	1[min]	3[min]	2.9[min]	1.8[min]	2.5[min]	3.6[min]	3.3[min]

Travel Time

1.5	2	2.2	1.4	1.2	3.5	3	1.8	1.2

Planned and Unplanned Down Time

1932	1932	1932	1932	1932	1932	1932	1932
0	0	0	0	0	0	0	0

Number Of Workers

1	1	4	2

Defects % = 0.1

Lean Tools Configuration

SMED Setup Time Reduction

0.4	0.3	0.2	0	0	0	0.4	0

5S Processing Time and defects reduction

0.3	0.25	0.3	0.25	0.1	0.25	0.4	0.3	0.03

UCell Travel Time Reduction

0.7	0.6	0.8	0.4	0.4	0.8	0.7	0.5	0.5

PokaYoke Defects Reduction 0.09

Pull

3	3	3	3	3	3	3	3

ANOVA Test

Descriptives

		N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
						Lower Bound	Upper Bound		
WIP	5S model	288	4.7361	1.42152	.08376	4.5712	4.9010	1.00	8.00
	Actual model	288	6.4653	1.83240	.10797	6.2528	6.6778	2.00	12.00
	Cross training	288	6.4653	1.83240	.10797	6.2528	6.6778	2.00	12.00
	Poka Yoke	288	5.2639	1.33297	.07855	5.1093	5.4185	2.00	9.00
	Pull	288	6.6146	1.95437	.11516	6.3879	6.8413	3.00	12.00
	SMED	288	5.5590	1.53132	.09023	5.3814	5.7366	1.00	11.00
	Ucell	288	6.4167	1.62301	.09564	6.2284	6.6049	3.00	12.00
	Total	2016	5.9315	1.79447	.03997	5.8532	6.0099	1.00	12.00
Lead-time	5S model	288	1.1907	.06549	.00386	1.1831	1.1983	1.04	1.37
	Actual model	288	1.3763	.10111	.00596	1.3646	1.3880	1.20	1.76
	Cross training	288	1.3763	.10111	.00596	1.3646	1.3880	1.20	1.76
	Poka Yoke	288	1.3175	.06325	.00373	1.3101	1.3248	1.20	1.51
	Pull	288	1.3774	.09456	.00557	1.3664	1.3883	1.20	1.78
	SMED	288	1.2805	.06095	.00359	1.2734	1.2875	1.17	1.43
	Ucell	288	1.1723	.09916	.00584	1.1608	1.1838	.99	1.45
	Total	2016	1.2987	.11811	.00263	1.2935	1.3039	.99	1.78
Defect rate	5S model	288	7.0260	2.51055	.14794	6.7348	7.3172	1.45	11.76
	Actual model	288	11.3996	3.40395	.20058	11.0048	11.7944	5.88	18.75
	Cross training	288	11.3996	3.40395	.20058	11.0048	11.7944	5.88	18.75
	Poka Yoke	288	.8487	1.09956	.06479	.7212	.9762	.00	3.23
	Pull	288	11.7531	3.66818	.21615	11.3276	12.1785	6.35	22.22
	SMED	288	11.3683	3.57900	.21089	10.9532	11.7834	5.71	18.46
	Ucell	288	11.4016	3.68005	.21685	10.9748	11.8284	5.48	18.46
	Total	2016	9.3138	4.93466	.10990	9.0983	9.5294	.00	22.22

Multiple Comparisons

Tukey HSD

Dependent Variable	(I) Models	(J) Models	Mean Difference (I-J)	Std. Error	p-value	95% Confidence Interval	
						Lower Bound	Upper Bound
WIP	5S model	Actual model	-1.72917*	.13841	.000	-2.1377	-1.3207
		Cross training	-1.72917*	.13841	.000	-2.1377	-1.3207
		Poka Yoke	-.52778*	.13841	.003	-.9363	-.1193
		Pull	-1.87847*	.13841	.000	-2.2870	-1.4700
		SMED	-.82292*	.13841	.000	-1.2314	-.4144
		Ucell	-1.68056*	.13841	.000	-2.0891	-1.2721
	Actual model	5S model	1.72917*	.13841	.000	1.3207	2.1377
		Cross training	.00000	.13841	1.000	-.4085	.4085
		Poka Yoke	1.20139*	.13841	.000	.7929	1.6099
		Pull	-.14931	.13841	.934	-.5578	.2592
		SMED	.90625*	.13841	.000	.4977	1.3148
		Ucell	.04861	.13841	1.000	-.3599	.4571
	Cross training	5S model	1.72917*	.13841	.000	1.3207	2.1377
		Actual model	.00000	.13841	1.000	-.4085	.4085
		Poka Yoke	1.20139*	.13841	.000	.7929	1.6099
		Pull	-.14931	.13841	.934	-.5578	.2592

		SMED	.90625*	.13841	.000	.4977	1.3148
		Ucell	.04861	.13841	1.000	-.3599	.4571
	Poka Yoke	5S model	.52778*	.13841	.003	.1193	.9363
		Actual model	-1.20139*	.13841	.000	-1.6099	-.7929
		Cross training	-1.20139*	.13841	.000	-1.6099	-.7929
		Pull	-1.35069*	.13841	.000	-1.7592	-.9422
		SMED	-.29514	.13841	.334	-.7036	.1134
		Ucell	-1.15278*	.13841	.000	-1.5613	-.7443
	Pull	5S model	1.87847*	.13841	.000	1.4700	2.2870
		Actual model	.14931	.13841	.934	-.2592	.5578
		Cross training	.14931	.13841	.934	-.2592	.5578
		Poka Yoke	1.35069*	.13841	.000	.9422	1.7592
		SMED	1.05556*	.13841	.000	.6471	1.4641
		Ucell	.19792	.13841	.786	-.2106	.6064
	SMED	5S model	.82292*	.13841	.000	.4144	1.2314
		Actual model	-.90625*	.13841	.000	-1.3148	-.4977
		Cross training	-.90625*	.13841	.000	-1.3148	-.4977
		Poka Yoke	.29514	.13841	.334	-.1134	.7036
		Pull	-1.05556*	.13841	.000	-1.4641	-.6471
		Ucell	-.85764*	.13841	.000	-1.2661	-.4491
	Ucell	5S model	1.68056*	.13841	.000	1.2721	2.0891
		Actual model	-.04861	.13841	1.000	-.4571	.3599
		Cross training	-.04861	.13841	1.000	-.4571	.3599
		Poka Yoke	1.15278*	.13841	.000	.7443	1.5613
		Pull	-.19792	.13841	.786	-.6064	.2106
		SMED	.85764*	.13841	.000	.4491	1.2661
Lead-time	5S model	Actual model	-.18564*	.00713	.000	-.2067	-.1646
		Cross training	-.18564*	.00713	.000	-.2067	-.1646
		Poka Yoke	-.12680*	.00713	.000	-.1478	-.1058
		Pull	-.18670*	.00713	.000	-.2077	-.1657
		SMED	-.08982*	.00713	.000	-.1109	-.0688
		Ucell	.01833	.00713	.135	-.0027	.0394
	Actual model	5S model	.18564*	.00713	.000	.1646	.2067
		Cross training	.00000	.00713	1.000	-.0210	.0210
		Poka Yoke	.05884*	.00713	.000	.0378	.0799
		Pull	-.00106	.00713	1.000	-.0221	.0200
		SMED	.09582*	.00713	.000	.0748	.1169
		Ucell	.20397*	.00713	.000	.1829	.2250
	Cross training	5S model	.18564*	.00713	.000	.1646	.2067
		Actual model	.00000	.00713	1.000	-.0210	.0210
		Poka Yoke	.05884*	.00713	.000	.0378	.0799
		Pull	-.00106	.00713	1.000	-.0221	.0200
		SMED	.09582*	.00713	.000	.0748	.1169
		Ucell	.20397*	.00713	.000	.1829	.2250
	Poka Yoke	5S model	.12680*	.00713	.000	.1058	.1478
		Actual model	-.05884*	.00713	.000	-.0799	-.0378
		Cross training	-.05884*	.00713	.000	-.0799	-.0378
		Pull	-.05990*	.00713	.000	-.0809	-.0389
		SMED	.03698*	.00713	.000	.0159	.0580
		Ucell	.14513*	.00713	.000	.1241	.1662
	Pull	5S model	.18670*	.00713	.000	.1657	.2077
		Actual model	.00106	.00713	1.000	-.0200	.0221
		Cross training	.00106	.00713	1.000	-.0200	.0221
		Poka Yoke	.05990*	.00713	.000	.0389	.0809
		SMED	.09688*	.00713	.000	.0758	.1179
		Ucell	.20503*	.00713	.000	.1840	.2261
	SMED	5S model	.08982*	.00713	.000	.0688	.1109
		Actual model	-.09582*	.00713	.000	-.1169	-.0748
		Cross training	-.09582*	.00713	.000	-.1169	-.0748

		Poka Yoke	-.03698*	.00713	.000	-.0580	-.0159
		Pull	-.09688*	.00713	.000	-.1179	-.0758
		Ucell	.10815*	.00713	.000	.0871	.1292
	Ucell	5S model	-.01833	.00713	.135	-.0394	.0027
		Actual model	-.20397*	.00713	.000	-.2250	-.1829
		Cross training	-.20397*	.00713	.000	-.2250	-.1829
		Poka Yoke	-.14513*	.00713	.000	-.1662	-.1241
		Pull	-.20503*	.00713	.000	-.2261	-.1840
		SMED	-.10815*	.00713	.000	-.1292	-.0871
Defect rate	5S model	Actual model	-4.37358*	.26445	.000	-5.1541	-3.5931
		Cross training	-4.37358*	.26445	.000	-5.1541	-3.5931
		Poka Yoke	6.17729*	.26445	.000	5.3968	6.9578
		Pull	-4.72709*	.26445	.000	-5.5076	-3.9466
		SMED	-4.34229*	.26445	.000	-5.1228	-3.5618
		Ucell	-4.37563*	.26445	.000	-5.1561	-3.5952
	Actual model	5S model	4.37358*	.26445	.000	3.5931	5.1541
		Cross training	.00000	.26445	1.000	-.7805	.7805
		Poka Yoke	10.55086*	.26445	.000	9.7704	11.3313
		Pull	-.35351	.26445	.835	-1.1340	.4270
		SMED	.03128	.26445	1.000	-.7492	.8118
		Ucell	-.00205	.26445	1.000	-.7825	.7784
	Cross training	5S model	4.37358*	.26445	.000	3.5931	5.1541
		Actual model	.00000	.26445	1.000	-.7805	.7805
		Poka Yoke	10.55086*	.26445	.000	9.7704	11.3313
		Pull	-.35351	.26445	.835	-1.1340	.4270
		SMED	.03128	.26445	1.000	-.7492	.8118
		Ucell	-.00205	.26445	1.000	-.7825	.7784
	Poka Yoke	5S model	-6.17729*	.26445	.000	-6.9578	-5.3968
		Actual model	-10.55086*	.26445	.000	-11.3313	-9.7704
		Cross training	-10.55086*	.26445	.000	-11.3313	-9.7704
		Pull	-10.90437*	.26445	.000	-11.6848	-10.1239
		SMED	-10.51958*	.26445	.000	-11.3001	-9.7391
		Ucell	-10.55292*	.26445	.000	-11.3334	-9.7724
	Pull	5S model	4.72709*	.26445	.000	3.9466	5.5076
		Actual model	.35351	.26445	.835	-.4270	1.1340
		Cross training	.35351	.26445	.835	-.4270	1.1340
		Poka Yoke	10.90437*	.26445	.000	10.1239	11.6848
		SMED	.38479	.26445	.771	-.3957	1.1653
		Ucell	.35145	.26445	.838	-.4290	1.1319
	SMED	5S model	4.34229*	.26445	.000	3.5618	5.1228
		Actual model	-.03128	.26445	1.000	-.8118	.7492
		Cross training	-.03128	.26445	1.000	-.8118	.7492
		Poka Yoke	10.51958*	.26445	.000	9.7391	11.3001
		Pull	-.38479	.26445	.771	-1.1653	.3957
		Ucell	-.03334	.26445	1.000	-.8138	.7471
	Ucell	5S model	4.37563*	.26445	.000	3.5952	5.1561
		Actual model	.00205	.26445	1.000	-.7784	.7825
		Cross training	.00205	.26445	1.000	-.7784	.7825
		Poka Yoke	10.55292*	.26445	.000	9.7724	11.3334
		Pull	-.35145	.26445	.838	-1.1319	.4290
		SMED	.03334	.26445	1.000	-.7471	.8138

***. The mean difference is significant at the 0.05 level.**

Run 2: Market Fluctuation

Initial market demand:

168.6666667[h]	337	328	340	337
153.3333333[h]	324	342	330	339
168.6666667[h]	321	322	330	334
153.3333333[h]	322	320	338	320
145.6666667[h]	324	331	325	320
161.0000000[h]	324	326	334	331
168.6666667[h]	330	329	320	330
168.6666667[h]	325	331	335	328
153.3333333[h]	331	320	334	336
176.3333333[h]	322	329	333	328
161.0000000[h]	323	325	330	335
153.3333333[h]	320	334	331	342

Other inputs are the same as the previous simulation run.

ANOVA Test

Descriptives									
		N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
						Lower Bound	Upper Bound		
WIP	5S	480	4.9500	1.64666	.07516	4.8023	5.0977	1.00	10.00
	Actual model	480	44.0646	41.90264	1.91258	40.3065	47.8227	2.00	119.00
	Cross training	480	44.0646	41.90264	1.91258	40.3065	47.8227	2.00	119.00
	Poka Yoke	480	12.5917	15.61689	.71281	11.1910	13.9923	2.00	68.00
	Pull	480	9.1354	5.42941	.24782	8.6485	9.6224	1.00	25.00
	SMED	480	8.5958	8.87228	.40496	7.8001	9.3916	.00	49.00
	Ucell	480	43.4063	41.46714	1.89271	39.6872	47.1253	2.00	120.00
	Total	3360	23.8298	33.18672	.57253	22.7072	24.9523	.00	120.00
Lead-time	5S	480	1.1834	.06929	.00316	1.1772	1.1896	1.00	1.53
	Actual model	480	5.1486	4.19603	.19152	4.7722	5.5249	1.20	12.60
	Cross training	480	5.1486	4.19603	.19152	4.7722	5.5249	1.20	12.60
	Poka Yoke	480	2.0222	1.53080	.06987	1.8849	2.1595	1.20	7.53
	Pull	480	1.6087	.48364	.02207	1.5653	1.6520	1.19	2.99
	SMED	480	1.5260	.74855	.03417	1.4589	1.5931	1.15	4.92
	Ucell	480	4.8590	4.13835	.18889	4.4878	5.2301	.92	12.36
	Total	3360	3.0709	3.30382	.05700	2.9592	3.1827	.92	12.60
Defect rate	5S	480	6.7830	2.89935	.13234	6.5229	7.0430	1.37	14.00
	Actual model	480	10.9918	4.04780	.18476	10.6288	11.3548	2.60	21.31
	Cross training	480	10.9918	4.04780	.18476	10.6288	11.3548	2.60	21.31
	Poka Yoke	480	.7795	1.20372	.05494	.6716	.8875	.00	5.77
	Pull	480	11.6333	4.30708	.19659	11.2470	12.0196	1.79	23.44
	SMED	480	10.9037	4.29008	.19581	10.5189	11.2884	.00	21.31
	Ucell	480	10.9805	4.09331	.18683	10.6134	11.3476	2.56	21.31
	Total	3360	9.0091	5.22491	.09014	8.8324	9.1858	.00	23.44
Production	5S	480	67.4333	8.93935	.40802	66.6316	68.2351	50.00	92.00
	Actual model	480	67.4000	6.52907	.29801	66.8144	67.9856	52.00	77.00

Throughput	Cross training	480	67.4000	6.52907	.29801	66.8144	67.9856	52.00	77.00
	Poka Yoke	480	67.4167	8.01729	.36594	66.6976	68.1357	52.00	79.00
	Pull	480	65.4333	6.24180	.28490	64.8735	65.9931	50.00	74.00
	SMED	480	67.4167	8.30784	.37920	66.6716	68.1618	52.00	83.00
	Ucell	480	67.4333	6.53461	.29826	66.8473	68.0194	52.00	78.00
	Total	3360	67.1333	7.39520	.12758	66.8832	67.3835	50.00	92.00

Multiple Comparisons

Tukey HSD

Dependent Variable	(I) Models	(J) Models	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
						Lower Bound	Upper Bound
WIP	5S	Actual model	-39.11458*	1.82345	.000	-44.4940	-33.7352
		Cross training	-39.11458*	1.82345	.000	-44.4940	-33.7352
		Poka Yoke	-7.64167*	1.82345	.001	-13.0211	-2.2623
		Pull	-4.18542	1.82345	.246	-9.5648	1.1940
		SMED	-3.64583	1.82345	.415	-9.0252	1.7336
		Ucell	-38.45625*	1.82345	.000	-43.8357	-33.0768
	Actual model	5S	39.11458*	1.82345	.000	33.7352	44.4940
		Cross training	.00000	1.82345	1.000	-5.3794	5.3794
		Poka Yoke	31.47292*	1.82345	.000	26.0935	36.8523
		Pull	34.92917*	1.82345	.000	29.5498	40.3086
		SMED	35.46875*	1.82345	.000	30.0893	40.8482
		Ucell	.65833	1.82345	1.000	-4.7211	6.0377
	Cross training	5S	39.11458*	1.82345	.000	33.7352	44.4940
		Actual model	.00000	1.82345	1.000	-5.3794	5.3794
		Poka Yoke	31.47292*	1.82345	.000	26.0935	36.8523
		Pull	34.92917*	1.82345	.000	29.5498	40.3086
		SMED	35.46875*	1.82345	.000	30.0893	40.8482
		Ucell	.65833	1.82345	1.000	-4.7211	6.0377
	Poka Yoke	5S	7.64167*	1.82345	.001	2.2623	13.0211
		Actual model	-31.47292*	1.82345	.000	-36.8523	-26.0935
		Cross training	-31.47292*	1.82345	.000	-36.8523	-26.0935
		Pull	3.45625	1.82345	.483	-1.9232	8.8357
		SMED	3.99583	1.82345	.300	-1.3836	9.3752
		Ucell	-30.81458*	1.82345	.000	-36.1940	-25.4352
	Pull	5S	4.18542	1.82345	.246	-1.1940	9.5648
		Actual model	-34.92917*	1.82345	.000	-40.3086	-29.5498
		Cross training	-34.92917*	1.82345	.000	-40.3086	-29.5498
		Poka Yoke	-3.45625	1.82345	.483	-8.8357	1.9232
		SMED	.53958	1.82345	1.000	-4.8398	5.9190
		Ucell	-34.27083*	1.82345	.000	-39.6502	-28.8914
	SMED	5S	3.64583	1.82345	.415	-1.7336	9.0252
		Actual model	-35.46875*	1.82345	.000	-40.8482	-30.0893
		Cross training	-35.46875*	1.82345	.000	-40.8482	-30.0893
		Poka Yoke	-3.99583	1.82345	.300	-9.3752	1.3836
		Pull	-.53958	1.82345	1.000	-5.9190	4.8398
		Ucell	-34.81042*	1.82345	.000	-40.1898	-29.4310
	Ucell	5S	38.45625*	1.82345	.000	33.0768	43.8357
		Actual model	-.65833	1.82345	1.000	-6.0377	4.7211
		Cross training	-.65833	1.82345	1.000	-6.0377	4.7211
		Poka Yoke	30.81458*	1.82345	.000	25.4352	36.1940
		Pull	34.27083*	1.82345	.000	28.8914	39.6502
		SMED	34.81042*	1.82345	.000	29.4310	40.1898
LeadTime	5S	Actual model	-3.96514*	.18173	.000	-4.5013	-3.4290
		Cross training	-3.96514*	.18173	.000	-4.5013	-3.4290
		Poka Yoke	-.83883*	.18173	.000	-1.3749	-.3027
		Pull	-.42524	.18173	.225	-.9614	.1109
		SMED	-.34259	.18173	.490	-.8787	.1935

		Ucell	-3.67557*	.18173	.000	-4.2117	-3.1395
	Actual model	5S	3.96514*	.18173	.000	3.4290	4.5013
		Cross training	.00000	.18173	1.000	-.5361	.5361
		Poka Yoke	3.12631*	.18173	.000	2.5902	3.6624
		Pull	3.53990*	.18173	.000	3.0038	4.0760
		SMED	3.62255*	.18173	.000	3.0864	4.1587
		Ucell	.28957	.18173	.687	-.2465	.8257
	Cross training	5S	3.96514*	.18173	.000	3.4290	4.5013
		Actual model	.00000	.18173	1.000	-.5361	.5361
		Poka Yoke	3.12631*	.18173	.000	2.5902	3.6624
		Pull	3.53990*	.18173	.000	3.0038	4.0760
		SMED	3.62255*	.18173	.000	3.0864	4.1587
		Ucell	.28957	.18173	.687	-.2465	.8257
	Poka Yoke	5S	.83883*	.18173	.000	.3027	1.3749
		Actual model	-3.12631*	.18173	.000	-3.6624	-2.5902
		Cross training	-3.12631*	.18173	.000	-3.6624	-2.5902
		Pull	.41359	.18173	.256	-.1225	.9497
		SMED	.49624	.18173	.091	-.0399	1.0324
		Ucell	-2.83674*	.18173	.000	-3.3729	-2.3006
	Pull	5S	.42524	.18173	.225	-.1109	.9614
		Actual model	-3.53990*	.18173	.000	-4.0760	-3.0038
		Cross training	-3.53990*	.18173	.000	-4.0760	-3.0038
		Poka Yoke	-.41359	.18173	.256	-.9497	.1225
		SMED	.08265	.18173	.999	-.4535	.6188
		Ucell	-3.25033*	.18173	.000	-3.7864	-2.7142
	SMED	5S	.34259	.18173	.490	-.1935	.8787
		Actual model	-3.62255*	.18173	.000	-4.1587	-3.0864
		Cross training	-3.62255*	.18173	.000	-4.1587	-3.0864
		Poka Yoke	-.49624	.18173	.091	-1.0324	.0399
		Pull	-.08265	.18173	.999	-.6188	.4535
		Ucell	-3.33298*	.18173	.000	-3.8691	-2.7969
	Ucell	5S	3.67557*	.18173	.000	3.1395	4.2117
		Actual model	-.28957	.18173	.687	-.8257	.2465
		Cross training	-.28957	.18173	.687	-.8257	.2465
		Poka Yoke	2.83674*	.18173	.000	2.3006	3.3729
		Pull	3.25033*	.18173	.000	2.7142	3.7864
		SMED	3.33298*	.18173	.000	2.7969	3.8691
DefectRate	5S	Actual model	-4.20884*	.23946	.000	-4.9153	-3.5024
		Cross training	-4.20884*	.23946	.000	-4.9153	-3.5024
		Poka Yoke	6.00345*	.23946	.000	5.2970	6.7099
		Pull	-4.85030*	.23946	.000	-5.5567	-4.1439
		SMED	-4.12068*	.23946	.000	-4.8271	-3.4142
		Ucell	-4.19755*	.23946	.000	-4.9040	-3.4911
	Actual model	5S	4.20884*	.23946	.000	3.5024	4.9153
		Cross training	.00000	.23946	1.000	-.7064	.7064
		Poka Yoke	10.21229*	.23946	.000	9.5058	10.9187
		Pull	-.64147	.23946	.104	-1.3479	.0650
		SMED	.08816	.23946	1.000	-.6183	.7946
		Ucell	.01128	.23946	1.000	-.6952	.7177
	Cross training	5S	4.20884*	.23946	.000	3.5024	4.9153
		Actual model	.00000	.23946	1.000	-.7064	.7064
		Poka Yoke	10.21229*	.23946	.000	9.5058	10.9187
		Pull	-.64147	.23946	.104	-1.3479	.0650
		SMED	.08816	.23946	1.000	-.6183	.7946
		Ucell	.01128	.23946	1.000	-.6952	.7177
	Poka Yoke	5S	-6.00345*	.23946	.000	-6.7099	-5.2970
		Actual model	-10.21229*	.23946	.000	-10.9187	-9.5058
		Cross training	-10.21229*	.23946	.000	-10.9187	-9.5058
		Pull	-10.85375*	.23946	.000	-11.5602	-10.1473

		SMED	-10.12413*	.23946	.000	-10.8306	-9.4177
		Ucell	-10.20100*	.23946	.000	-10.9074	-9.4946
	Pull	5S	4.85030*	.23946	.000	4.1439	5.5567
		Actual model	.64147	.23946	.104	-.0650	1.3479
		Cross training	.64147	.23946	.104	-.0650	1.3479
		Poka Yoke	10.85375*	.23946	.000	10.1473	11.5602
		SMED	.72962*	.23946	.038	.0232	1.4361
		Ucell	.65275	.23946	.092	-.0537	1.3592
	SMED	5S	4.12068*	.23946	.000	3.4142	4.8271
		Actual model	-.08816	.23946	1.000	-.7946	.6183
		Cross training	-.08816	.23946	1.000	-.7946	.6183
		Poka Yoke	10.12413*	.23946	.000	9.4177	10.8306
		Pull	-.72962*	.23946	.038	-1.4361	-.0232
		Ucell	-.07687	.23946	1.000	-.7833	.6296
	Ucell	5S	4.19755*	.23946	.000	3.4911	4.9040
		Actual model	-.01128	.23946	1.000	-.7177	.6952
		Cross training	-.01128	.23946	1.000	-.7177	.6952
		Poka Yoke	10.20100*	.23946	.000	9.4946	10.9074
		Pull	-.65275	.23946	.092	-1.3592	.0537
		SMED	.07687	.23946	1.000	-.6296	.7833
Production Throughput	5S	Actual model	.03333	.47567	1.000	-1.3700	1.4366
		Cross training	.03333	.47567	1.000	-1.3700	1.4366
		Poka Yoke	.01667	.47567	1.000	-1.3866	1.4200
		Pull	2.00000*	.47567	.001	.5967	3.4033
		SMED	.01667	.47567	1.000	-1.3866	1.4200
		Ucell	.00000	.47567	1.000	-1.4033	1.4033
	Actual model	5S	-.03333	.47567	1.000	-1.4366	1.3700
		Cross training	.00000	.47567	1.000	-1.4033	1.4033
		Poka Yoke	-.01667	.47567	1.000	-1.4200	1.3866
		Pull	1.96667*	.47567	.001	.5634	3.3700
		SMED	-.01667	.47567	1.000	-1.4200	1.3866
		Ucell	-.03333	.47567	1.000	-1.4366	1.3700
	Cross training	5S	-.03333	.47567	1.000	-1.4366	1.3700
		Actual model	.00000	.47567	1.000	-1.4033	1.4033
		Poka Yoke	-.01667	.47567	1.000	-1.4200	1.3866
		Pull	1.96667*	.47567	.001	.5634	3.3700
		SMED	-.01667	.47567	1.000	-1.4200	1.3866
		Ucell	-.03333	.47567	1.000	-1.4366	1.3700
	Poka Yoke	5S	-.01667	.47567	1.000	-1.4200	1.3866
		Actual model	.01667	.47567	1.000	-1.3866	1.4200
		Cross training	.01667	.47567	1.000	-1.3866	1.4200
		Pull	1.98333*	.47567	.001	.5800	3.3866
		SMED	.00000	.47567	1.000	-1.4033	1.4033
		Ucell	-.01667	.47567	1.000	-1.4200	1.3866
	Pull	5S	-2.00000*	.47567	.001	-3.4033	-.5967
		Actual model	-1.96667*	.47567	.001	-3.3700	-.5634
		Cross training	-1.96667*	.47567	.001	-3.3700	-.5634
		Poka Yoke	-1.98333*	.47567	.001	-3.3866	-.5800
		SMED	-1.98333*	.47567	.001	-3.3866	-.5800
		Ucell	-2.00000*	.47567	.001	-3.4033	-.5967
	SMED	5S	-.01667	.47567	1.000	-1.4200	1.3866
		Actual model	.01667	.47567	1.000	-1.3866	1.4200
		Cross training	.01667	.47567	1.000	-1.3866	1.4200
		Poka Yoke	.00000	.47567	1.000	-1.4033	1.4033
		Pull	1.98333*	.47567	.001	.5800	3.3866
		Ucell	-.01667	.47567	1.000	-1.4200	1.3866
	Ucell	5S	.00000	.47567	1.000	-1.4033	1.4033
		Actual model	.03333	.47567	1.000	-1.3700	1.4366
		Cross training	.03333	.47567	1.000	-1.3700	1.4366

		Poka Yoke	.01667	.47567	1.000	-1.3866	1.4200
		Pull	2.00000*	.47567	.001	.5967	3.4033
		SMED	.01667	.47567	1.000	-1.3866	1.4200
***. The mean difference is significant at the 0.05 level.**							

Run 3: Diversification of demand

Market demand for 2 product references

168.6666667[h]	732	745
153.3333333[h]	733	736
168.6666667[h]	707	730
153.3333333[h]	706	724
145.6666667[h]	721	710
161.0000000[h]	715	732
168.6666667[h]	725	715
168.6666667[h]	722	729
153.3333333[h]	716	737
176.3333333[h]	716	727
161.0000000[h]	713	732
153.3333333[h]	719	740

Same overall total quantity is used for all varieties. (e.g. for 4 product references, value is divided by 2 so that the total quantity of the month remains the same)

Setup Times																			
2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]
1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	[min]	[min]	[min]	[min]
3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]	3.5[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]
2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]	2.8[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]

Descriptives									
		N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
						Lower Bound	Upper Bound		
WIP	5S model	1088	24.0156	43.56012	1.32061	21.4244	26.6069	2.00	180.00
	Actual model	1088	138.1967	114.02662	3.45694	131.4137	144.9797	4.00	426.00
	Cross training	1088	137.8153	113.45244	3.43953	131.0664	144.5641	4.00	428.00
	Poka Yoke	1088	52.1360	69.91477	2.11960	47.9771	56.2950	2.00	269.00
	Pull	1088	8.9706	3.07093	.09310	8.7879	9.1533	2.00	19.00
	SMED	1088	52.6287	70.31014	2.13159	48.4462	56.8112	2.00	262.00
	Ucell	1088	140.7013	115.52389	3.50233	133.8292	147.5734	3.00	431.00
	Total	7616	79.2092	100.73660	1.15431	76.9464	81.4719	2.00	431.00

Lead-time	5S model	1088	3.0119	4.30447	.13050	2.7559	3.2680	1.00	19.91
	Actual model	1088	14.4482	11.43731	.34674	13.7678	15.1285	1.20	45.90
	Cross training	1088	14.4155	11.37303	.34480	13.7390	15.0921	1.20	45.90
	Poka Yoke	1088	6.3020	7.52001	.22798	5.8547	6.7494	1.21	31.96
	Pull	1088	1.6173	.40241	.01220	1.5933	1.6412	1.19	2.94
	SMED	1088	5.8249	6.91189	.20955	5.4137	6.2360	1.16	28.99
	Ucell	1088	14.5209	11.58008	.35107	13.8320	15.2097	1.03	45.71
	Total	7616	8.5915	10.09196	.11564	8.3648	8.8182	1.00	45.90
Defect rate	5S model	1088	7.2559	2.84024	.08611	7.0870	7.4249	.00	15.15
	Actual model	1088	11.1268	4.25875	.12911	10.8734	11.3801	2.04	24.07
	Cross training	1088	11.1183	4.03997	.12248	10.8780	11.3586	2.06	22.22
	Poka Yoke	1088	.9519	1.19205	.03614	.8810	1.0229	.00	5.88
	Pull	1088	11.4636	4.50545	.13659	11.1956	11.7316	1.54	23.64
	SMED	1088	11.0834	4.06014	.12309	10.8419	11.3249	2.78	21.88
	Ucell	1088	11.1332	4.13666	.12541	10.8871	11.3793	1.03	24.19
	Total	7616	9.1619	5.20134	.05960	9.0450	9.2787	.00	24.19
Production throughput	5S model	1088	72.3897	8.22961	.24950	71.9002	72.8793	61.00	123.00
	Actual model	1088	72.3897	12.62125	.38264	71.6389	73.1405	54.00	98.00
	Cross training	1088	72.3676	12.60704	.38221	71.6177	73.1176	55.00	97.00
	Poka Yoke	1088	72.3897	10.59772	.32129	71.7593	73.0201	60.00	107.00
	Pull	1088	67.5294	7.01051	.21254	67.1124	67.9464	54.00	79.00
	SMED	1088	72.4044	8.55256	.25929	71.8957	72.9132	60.00	96.00
	Ucell	1088	72.3750	12.69032	.38473	71.6201	73.1299	55.00	97.00
	Total	7616	71.6922	10.69856	.12259	71.4519	71.9325	54.00	123.00

Multiple Comparisons

Tukey HSD

Dependent Variable	(I) Models	(J) Models	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
						Lower Bound	Upper Bound
WIP	5S model	Actual model	-114.18107*	3.65811	.000	-124.9692	-103.3929
		Cross training	-113.79963*	3.65811	.000	-124.5878	-103.0115
		Poka Yoke	-28.12040*	3.65811	.000	-38.9086	-17.3322
		Pull	15.04504*	3.65811	.001	4.2569	25.8332
		SMED	-28.61305*	3.65811	.000	-39.4012	-17.8249
		Ucell	-116.68566*	3.65811	.000	-127.4738	-105.8975
	Actual model	5S model	114.18107*	3.65811	.000	103.3929	124.9692
		Cross training	.38143	3.65811	1.000	-10.4067	11.1696
		Poka Yoke	86.06066*	3.65811	.000	75.2725	96.8488
		Pull	129.22610*	3.65811	.000	118.4379	140.0143
		SMED	85.56801*	3.65811	.000	74.7798	96.3562
		Ucell	-2.50460	3.65811	.993	-13.2928	8.2836
	Cross training	5S model	113.79963*	3.65811	.000	103.0115	124.5878
		Actual model	-.38143	3.65811	1.000	-11.1696	10.4067
		Poka Yoke	85.67923*	3.65811	.000	74.8911	96.4674
		Pull	128.84467*	3.65811	.000	118.0565	139.6328
		SMED	85.18658*	3.65811	.000	74.3984	95.9748
		Ucell	-2.88603	3.65811	.986	-13.6742	7.9021
	Poka Yoke	5S model	28.12040*	3.65811	.000	17.3322	38.9086
		Actual model	-86.06066*	3.65811	.000	-96.8488	-75.2725
		Cross training	-85.67923*	3.65811	.000	-96.4674	-74.8911
		Pull	43.16544*	3.65811	.000	32.3773	53.9536
		SMED	-.49265	3.65811	1.000	-11.2808	10.2955

		Ucell	-88.56526*	3.65811	.000	-99.3534	-77.7771
	Pull	5S model	-15.04504*	3.65811	.001	-25.8332	-4.2569
		Actual model	-129.22610*	3.65811	.000	-140.0143	-118.4379
		Cross training	-128.84467*	3.65811	.000	-139.6328	-118.0565
		Poka Yoke	-43.16544*	3.65811	.000	-53.9536	-32.3773
		SMED	-43.65809*	3.65811	.000	-54.4463	-32.8699
		Ucell	-131.73070*	3.65811	.000	-142.5189	-120.9425
	SMED	5S model	28.61305*	3.65811	.000	17.8249	39.4012
		Actual model	-85.56801*	3.65811	.000	-96.3562	-74.7798
		Cross training	-85.18658*	3.65811	.000	-95.9748	-74.3984
		Poka Yoke	.49265	3.65811	1.000	-10.2955	11.2808
		Pull	43.65809*	3.65811	.000	32.8699	54.4463
		Ucell	-88.07261*	3.65811	.000	-98.8608	-77.2844
	Ucell	5S model	116.68566*	3.65811	.000	105.8975	127.4738
		Actual model	2.50460	3.65811	.993	-8.2836	13.2928
		Cross training	2.88603	3.65811	.986	-7.9021	13.6742
		Poka Yoke	88.56526*	3.65811	.000	77.7771	99.3534
		Pull	131.73070*	3.65811	.000	120.9425	142.5189
		SMED	88.07261*	3.65811	.000	77.2844	98.8608
LeadTime	5S model	Actual model	-11.43621*	.36856	.000	-12.5231	-10.3493
		Cross training	-11.40358*	.36856	.000	-12.4905	-10.3166
		Poka Yoke	-3.29010*	.36856	.000	-4.3770	-2.2032
		Pull	1.39467*	.36856	.003	.3077	2.4816
		SMED	-2.81293*	.36856	.000	-3.8999	-1.7260
		Ucell	-11.50891*	.36856	.000	-12.5958	-10.4220
	Actual model	5S model	11.43621*	.36856	.000	10.3493	12.5231
		Cross training	.03263	.36856	1.000	-1.0543	1.1196
		Poka Yoke	8.14611*	.36856	.000	7.0592	9.2330
		Pull	12.83088*	.36856	.000	11.7440	13.9178
		SMED	8.62328*	.36856	.000	7.5363	9.7102
		Ucell	-.07271	.36856	1.000	-1.1596	1.0142
	Cross training	5S model	11.40358*	.36856	.000	10.3166	12.4905
		Actual model	-.03263	.36856	1.000	-1.1196	1.0543
		Poka Yoke	8.11348*	.36856	.000	7.0266	9.2004
		Pull	12.79825*	.36856	.000	11.7113	13.8852
		SMED	8.59065*	.36856	.000	7.5037	9.6776
		Ucell	-.10533	.36856	1.000	-1.1923	.9816
	Poka Yoke	5S model	3.29010*	.36856	.000	2.2032	4.3770
		Actual model	-8.14611*	.36856	.000	-9.2330	-7.0592
		Cross training	-8.11348*	.36856	.000	-9.2004	-7.0266
		Pull	4.68477*	.36856	.000	3.5978	5.7717
		SMED	.47717	.36856	.855	-.6098	1.5641
		Ucell	-8.21882*	.36856	.000	-9.3057	-7.1319
	Pull	5S model	-1.39467*	.36856	.003	-2.4816	-.3077
		Actual model	-12.83088*	.36856	.000	-13.9178	-11.7440
		Cross training	-12.79825*	.36856	.000	-13.8852	-11.7113
		Poka Yoke	-4.68477*	.36856	.000	-5.7717	-3.5978
		SMED	-4.20760*	.36856	.000	-5.2945	-3.1207
		Ucell	-12.90359*	.36856	.000	-13.9905	-11.8167
	SMED	5S model	2.81293*	.36856	.000	1.7260	3.8999
		Actual model	-8.62328*	.36856	.000	-9.7102	-7.5363
		Cross training	-8.59065*	.36856	.000	-9.6776	-7.5037
		Poka Yoke	-.47717	.36856	.855	-1.5641	.6098

		Pull	4.20760*	.36856	.000	3.1207	5.2945
		Ucell	-8.69599*	.36856	.000	-9.7829	-7.6091
	Ucell	5S model	11.50891*	.36856	.000	10.4220	12.5958
		Actual model	.07271	.36856	1.000	-1.0142	1.1596
		Cross training	.10533	.36856	1.000	-.9816	1.1923
		Poka Yoke	8.21882*	.36856	.000	7.1319	9.3057
		Pull	12.90359*	.36856	.000	11.8167	13.9905
		SMED	8.69599*	.36856	.000	7.6091	9.7829
DefectRate	5S model	Actual model	-3.87084*	.16029	.000	-4.3436	-3.3981
		Cross training	-3.86237*	.16029	.000	-4.3351	-3.3896
		Poka Yoke	6.30399*	.16029	.000	5.8313	6.7767
		Pull	-4.20769*	.16029	.000	-4.6804	-3.7350
		SMED	-3.82744*	.16029	.000	-4.3002	-3.3547
		Ucell	-3.87726*	.16029	.000	-4.3500	-3.4045
	Actual model	5S model	3.87084*	.16029	.000	3.3981	4.3436
		Cross training	.00847	.16029	1.000	-.4643	.4812
		Poka Yoke	10.17483*	.16029	.000	9.7021	10.6476
		Pull	-.33685	.16029	.352	-.8096	.1359
		SMED	.04340	.16029	1.000	-.4293	.5161
		Ucell	-.00642	.16029	1.000	-.4791	.4663
	Cross training	5S model	3.86237*	.16029	.000	3.3896	4.3351
		Actual model	-.00847	.16029	1.000	-.4812	.4643
		Poka Yoke	10.16636*	.16029	.000	9.6936	10.6391
		Pull	-.34532	.16029	.321	-.8180	.1274
		SMED	.03493	.16029	1.000	-.4378	.5076
		Ucell	-.01489	.16029	1.000	-.4876	.4578
	Poka Yoke	5S model	-6.30399*	.16029	.000	-6.7767	-5.8313
		Actual model	-10.17483*	.16029	.000	-10.6476	-9.7021
		Cross training	-10.16636*	.16029	.000	-10.6391	-9.6936
		Pull	-10.51168*	.16029	.000	-10.9844	-10.0390
		SMED	-10.13143*	.16029	.000	-10.6042	-9.6587
		Ucell	-10.18125*	.16029	.000	-10.6540	-9.7085
	Pull	5S model	4.20769*	.16029	.000	3.7350	4.6804
		Actual model	.33685	.16029	.352	-.1359	.8096
		Cross training	.34532	.16029	.321	-.1274	.8180
		Poka Yoke	10.51168*	.16029	.000	10.0390	10.9844
		SMED	.38025	.16029	.211	-.0925	.8530
		Ucell	.33044	.16029	.376	-.1423	.8032
	SMED	5S model	3.82744*	.16029	.000	3.3547	4.3002
		Actual model	-.04340	.16029	1.000	-.5161	.4293
		Cross training	-.03493	.16029	1.000	-.5076	.4378
		Poka Yoke	10.13143*	.16029	.000	9.6587	10.6042
		Pull	-.38025	.16029	.211	-.8530	.0925
		Ucell	-.04981	.16029	1.000	-.5225	.4229
	Ucell	5S model	3.87726*	.16029	.000	3.4045	4.3500
		Actual model	.00642	.16029	1.000	-.4663	.4791
		Cross training	.01489	.16029	1.000	-.4578	.4876
		Poka Yoke	10.18125*	.16029	.000	9.7085	10.6540
		Pull	-.33044	.16029	.376	-.8032	.1423
		SMED	.04981	.16029	1.000	-.4229	.5225
Production Throughput	5S model	Actual model	.00000	.45305	1.000	-1.3361	1.3361
		Cross training	.02206	.45305	1.000	-1.3140	1.3582
		Poka Yoke	.00000	.45305	1.000	-1.3361	1.3361

		Pull	4.86029*	.45305	.000	3.5242	6.1964
		SMED	-.01471	.45305	1.000	-1.3508	1.3214
		Ucell	.01471	.45305	1.000	-1.3214	1.3508
	Actual model	5S model	.00000	.45305	1.000	-1.3361	1.3361
		Cross training	.02206	.45305	1.000	-1.3140	1.3582
		Poka Yoke	.00000	.45305	1.000	-1.3361	1.3361
		Pull	4.86029*	.45305	.000	3.5242	6.1964
		SMED	-.01471	.45305	1.000	-1.3508	1.3214
		Ucell	.01471	.45305	1.000	-1.3214	1.3508
	Cross training	5S model	-.02206	.45305	1.000	-1.3582	1.3140
		Actual model	-.02206	.45305	1.000	-1.3582	1.3140
		Poka Yoke	-.02206	.45305	1.000	-1.3582	1.3140
		Pull	4.83824*	.45305	.000	3.5021	6.1743
		SMED	-.03676	.45305	1.000	-1.3729	1.2993
		Ucell	-.00735	.45305	1.000	-1.3434	1.3287
	Poka Yoke	5S model	.00000	.45305	1.000	-1.3361	1.3361
		Actual model	.00000	.45305	1.000	-1.3361	1.3361
		Cross training	.02206	.45305	1.000	-1.3140	1.3582
		Pull	4.86029*	.45305	.000	3.5242	6.1964
		SMED	-.01471	.45305	1.000	-1.3508	1.3214
		Ucell	.01471	.45305	1.000	-1.3214	1.3508
	Pull	5S model	-4.86029*	.45305	.000	-6.1964	-3.5242
		Actual model	-4.86029*	.45305	.000	-6.1964	-3.5242
		Cross training	-4.83824*	.45305	.000	-6.1743	-3.5021
		Poka Yoke	-4.86029*	.45305	.000	-6.1964	-3.5242
		SMED	-4.87500*	.45305	.000	-6.2111	-3.5389
		Ucell	-4.84559*	.45305	.000	-6.1817	-3.5095
	SMED	5S model	.01471	.45305	1.000	-1.3214	1.3508
		Actual model	.01471	.45305	1.000	-1.3214	1.3508
		Cross training	.03676	.45305	1.000	-1.2993	1.3729
		Poka Yoke	.01471	.45305	1.000	-1.3214	1.3508
		Pull	4.87500*	.45305	.000	3.5389	6.2111
		Ucell	.02941	.45305	1.000	-1.3067	1.3655
	Ucell	5S model	-.01471	.45305	1.000	-1.3508	1.3214
		Actual model	-.01471	.45305	1.000	-1.3508	1.3214
		Cross training	.00735	.45305	1.000	-1.3287	1.3434
		Poka Yoke	-.01471	.45305	1.000	-1.3508	1.3214
		Pull	4.84559*	.45305	.000	3.5095	6.1817
		SMED	-.02941	.45305	1.000	-1.3655	1.3067
***. The mean difference is significant at the 0.05 level.**							

Run 4: Uncertainty of resources

Setup Times																			
2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	2.2[min]	[min]	[min]	[min]	[min]
1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	1.5[min]	[min]	[min]	[min]	[min]
1[min]	1.8[min]	2.5[min]	2.8[min]	1[min]	1.8[min]	2.5[min]	2.8[min]	1[min]	1.8[min]	2.5[min]	2.8[min]	1[min]	1.8[min]	2.5[min]	2.8[min]	[min]	[min]	[min]	[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	[min]	[min]	[min]	[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	[min]	[min]	[min]	[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	[min]	[min]	[min]	[min]
1.2[min]	1.5[min]	2[min]	2.1[min]	1.2[min]	1.5[min]	2[min]	2.1[min]	1.2[min]	1.5[min]	2[min]	2.1[min]	1.2[min]	1.5[min]	2[min]	2.1[min]	[min]	[min]	[min]	[min]
0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	0[min]	[min]	[min]	[min]	[min]

Market demand

168.6666667[h]	337	328	340	337
153.3333333[h]	324	342	330	339
168.6666667[h]	321	322	330	334
153.3333333[h]	322	320	338	320
145.6666667[h]	324	331	325	320
161.0000000[h]	324	326	334	331
168.6666667[h]	330	329	320	330
168.6666667[h]	325	331	335	328
153.3333333[h]	331	320	334	336
176.3333333[h]	322	329	333	328
161.0000000[h]	323	325	330	335
153.3333333[h]	320	334	331	342

At t = 25h one worker (treatment shop) absent for one day
At t = 75h first machine of the assembly down for 2 hours
At t = 100h first machine of the machining shop down for one day (8 hours)

ANOVA Test

Descriptives									
		N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean		Minimum	Maximum
						Lower Bound	Upper Bound		
WIP	5S model	112	20.3750	21.32550	2.01507	16.3820	24.3680	1.00	72.00
	Actual model	112	49.6964	20.04942	1.89449	45.9424	53.4505	5.00	82.00
	Cross training	112	22.1875	23.13242	2.18581	17.8562	26.5188	3.00	73.00
	Poka Yoke	112	32.9107	21.31512	2.01409	28.9197	36.9018	4.00	75.00
	Pull	112	8.7054	5.55016	.52444	7.6661	9.7446	3.00	27.00
	SMED	112	30.5625	21.96549	2.07554	26.4497	34.6753	3.00	73.00
	Ucell	112	48.9911	19.92531	1.88276	45.2602	52.7219	5.00	80.00
	Total	784	30.4898	24.20147	.86434	28.7931	32.1865	1.00	82.00
Lead Time	5S model	112	2.5311	2.34038	.22115	2.0929	2.9693	1.07	9.94
	Actual model	112	5.6659	2.77504	.26222	5.1463	6.1855	1.26	10.34
	Cross training	112	2.9195	2.57484	.24330	2.4374	3.4017	1.22	9.38
	Poka Yoke	112	4.1385	2.77423	.26214	3.6190	4.6579	1.27	9.97
	Pull	112	1.6079	1.28651	.12156	1.3670	1.8487	1.20	9.12
	SMED	112	3.5376	2.57876	.24367	3.0548	4.0205	1.20	9.92
	Ucell	112	5.5115	2.73779	.25870	4.9989	6.0242	1.06	9.94
	Total	784	3.7017	2.84626	.10165	3.5022	3.9013	1.06	10.34
DefectRate	5S model	112	6.7057	3.08008	.29104	6.1290	7.2824	2.99	15.00
	Actual model	112	11.6594	5.07824	.47985	10.7085	12.6103	5.33	24.14
	Cross training	112	11.4400	4.34739	.41079	10.6260	12.2540	5.33	20.69
	Poka Yoke	112	.7901	1.21647	.11495	.5623	1.0179	.00	3.95
	Pull	112	10.5515	5.47435	.51728	9.5265	11.5765	2.78	20.83
	SMED	112	11.2796	3.53531	.33406	10.6177	11.9416	5.95	18.18
	Ucell	112	11.6564	5.17268	.48877	10.6879	12.6250	5.26	25.00
	Total	784	9.1547	5.65982	.20214	8.7579	9.5515	.00	25.00

Throughput	5S model	112	63.8571	21.71015	2.05142	59.7921	67.9222	24.00	100.00
	Actual model	112	59.8571	18.59907	1.75745	56.3746	63.3396	24.00	75.00
	Cross training	112	60.7143	13.27426	1.25430	58.2288	63.1998	29.00	75.00
	Poka Yoke	112	61.5714	18.68743	1.76580	58.0724	65.0705	26.00	78.00
	Pull	112	54.0000	15.81424	1.49430	51.0389	56.9611	24.00	74.00
	SMED	112	62.0000	20.18652	1.90745	58.2203	65.7797	24.00	84.00
	Ucell	112	59.9286	18.65172	1.76242	56.4362	63.4209	23.00	76.00
	Total	784	60.2755	18.46845	.65959	58.9807	61.5703	23.00	100.00

Multiple Comparisons

Tukey HSD

Dependent Variable	(I) Models	(J) Models	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
						Lower Bound	Upper Bound
WIP	5S model	Actual model	-29.32143*	2.65177	.000	-37.1603	-21.4826
		Cross training	-1.81250	2.65177	.993	-9.6514	6.0264
		Poka Yoke	-12.53571*	2.65177	.000	-20.3746	-4.6969
		Pull	11.66964*	2.65177	.000	3.8308	19.5085
		SMED	-10.18750*	2.65177	.003	-18.0264	-2.3486
		Ucell	-28.61607*	2.65177	.000	-36.4549	-20.7772
	Actual model	5S model	29.32143*	2.65177	.000	21.4826	37.1603
		Cross training	27.50893*	2.65177	.000	19.6701	35.3478
		Poka Yoke	16.78571*	2.65177	.000	8.9469	24.6246
		Pull	40.99107*	2.65177	.000	33.1522	48.8299
		SMED	19.13393*	2.65177	.000	11.2951	26.9728
		Ucell	.70536	2.65177	1.000	-7.1335	8.5442
	Cross training	5S model	1.81250	2.65177	.993	-6.0264	9.6514
		Actual model	-27.50893*	2.65177	.000	-35.3478	-19.6701
		Poka Yoke	-10.72321*	2.65177	.001	-18.5621	-2.8844
		Pull	13.48214*	2.65177	.000	5.6433	21.3210
		SMED	-8.37500*	2.65177	.027	-16.2139	-.5361
		Ucell	-26.80357*	2.65177	.000	-34.6424	-18.9647
	Poka Yoke	5S model	12.53571*	2.65177	.000	4.6969	20.3746
		Actual model	-16.78571*	2.65177	.000	-24.6246	-8.9469
		Cross training	10.72321*	2.65177	.001	2.8844	18.5621
		Pull	24.20536*	2.65177	.000	16.3665	32.0442
		SMED	2.34821	2.65177	.975	-5.4906	10.1871
		Ucell	-16.08036*	2.65177	.000	-23.9192	-8.2415
	Pull	5S model	-11.66964*	2.65177	.000	-19.5085	-3.8308
		Actual model	-40.99107*	2.65177	.000	-48.8299	-33.1522
		Cross training	-13.48214*	2.65177	.000	-21.3210	-5.6433
		Poka Yoke	-24.20536*	2.65177	.000	-32.0442	-16.3665
		SMED	-21.85714*	2.65177	.000	-29.6960	-14.0183
		Ucell	-40.28571*	2.65177	.000	-48.1246	-32.4469
	SMED	5S model	10.18750*	2.65177	.003	2.3486	18.0264
		Actual model	-19.13393*	2.65177	.000	-26.9728	-11.2951
		Cross training	8.37500*	2.65177	.027	.5361	16.2139
		Poka Yoke	-2.34821	2.65177	.975	-10.1871	5.4906

		Pull	21.85714*	2.65177	.000	14.0183	29.6960
		Ucell	-18.42857*	2.65177	.000	-26.2674	-10.5897
	Ucell	5S model	28.61607*	2.65177	.000	20.7772	36.4549
		Actual model	-.70536	2.65177	1.000	-8.5442	7.1335
		Cross training	26.80357*	2.65177	.000	18.9647	34.6424
		Poka Yoke	16.08036*	2.65177	.000	8.2415	23.9192
		Pull	40.28571*	2.65177	.000	32.4469	48.1246
		SMED	18.42857*	2.65177	.000	10.5897	26.2674
Lead Time	5S model	Actual model	-3.13481*	.33237	.000	-4.1173	-2.1523
		Cross training	-.38844	.33237	.906	-1.3710	.5941
		Poka Yoke	-1.60738*	.33237	.000	-2.5899	-.6249
		Pull	.92323	.33237	.082	-.0593	1.9058
		SMED	-1.00654*	.33237	.041	-1.9891	-.0240
		Ucell	-2.98046*	.33237	.000	-3.9630	-1.9979
	Actual model	5S model	3.13481*	.33237	.000	2.1523	4.1173
		Cross training	2.74636*	.33237	.000	1.7638	3.7289
		Poka Yoke	1.52742*	.33237	.000	.5449	2.5099
		Pull	4.05804*	.33237	.000	3.0755	5.0406
		SMED	2.12827*	.33237	.000	1.1457	3.1108
		Ucell	.15435	.33237	.999	-.8282	1.1369
	Cross training	5S model	.38844	.33237	.906	-.5941	1.3710
		Actual model	-2.74636*	.33237	.000	-3.7289	-1.7638
		Poka Yoke	-1.21894*	.33237	.005	-2.2015	-.2364
		Pull	1.31168*	.33237	.002	.3292	2.2942
		SMED	-.61809	.33237	.508	-1.6006	.3644
		Ucell	-2.59201*	.33237	.000	-3.5745	-1.6095
	Poka Yoke	5S model	1.60738*	.33237	.000	.6249	2.5899
		Actual model	-1.52742*	.33237	.000	-2.5099	-.5449
		Cross training	1.21894*	.33237	.005	.2364	2.2015
		Pull	2.53061*	.33237	.000	1.5481	3.5131
		SMED	.60085	.33237	.543	-.3817	1.5834
		Ucell	-1.37307*	.33237	.001	-2.3556	-.3905
	Pull	5S model	-.92323	.33237	.082	-1.9058	.0593
		Actual model	-4.05804*	.33237	.000	-5.0406	-3.0755
		Cross training	-1.31168*	.33237	.002	-2.2942	-.3292
		Poka Yoke	-2.53061*	.33237	.000	-3.5131	-1.5481
		SMED	-1.92977*	.33237	.000	-2.9123	-.9472
		Ucell	-3.90369*	.33237	.000	-4.8862	-2.9212
	SMED	5S model	1.00654*	.33237	.041	.0240	1.9891
		Actual model	-2.12827*	.33237	.000	-3.1108	-1.1457
		Cross training	.61809	.33237	.508	-.3644	1.6006
		Poka Yoke	-.60085	.33237	.543	-1.5834	.3817
		Pull	1.92977*	.33237	.000	.9472	2.9123
		Ucell	-1.97392*	.33237	.000	-2.9564	-.9914
	Ucell	5S model	2.98046*	.33237	.000	1.9979	3.9630
		Actual model	-.15435	.33237	.999	-1.1369	.8282
		Cross training	2.59201*	.33237	.000	1.6095	3.5745
		Poka Yoke	1.37307*	.33237	.001	.3905	2.3556
		Pull	3.90369*	.33237	.000	2.9212	4.8862
		SMED	1.97392*	.33237	.000	.9914	2.9564
DefectRate	5S model	Actual model	-4.95371*	.56442	.000	-6.6222	-3.2852
		Cross training	-4.73432*	.56442	.000	-6.4028	-3.0659
		Poka Yoke	5.91561*	.56442	.000	4.2471	7.5841

		Pull	-3.84583*	.56442	.000	-5.5143	-2.1774
		SMED	-4.57391*	.56442	.000	-6.2424	-2.9054
		Ucell	-4.95075*	.56442	.000	-6.6192	-3.2823
	Actual model	5S model	4.95371*	.56442	.000	3.2852	6.6222
		Cross training	.21939	.56442	1.000	-1.4491	1.8879
		Poka Yoke	10.86931*	.56442	.000	9.2008	12.5378
		Pull	1.10788	.56442	.439	-.5606	2.7763
		SMED	.37980	.56442	.994	-1.2887	2.0483
		Ucell	.00296	.56442	1.000	-1.6655	1.6714
	Cross training	5S model	4.73432*	.56442	.000	3.0659	6.4028
		Actual model	-.21939	.56442	1.000	-1.8879	1.4491
		Poka Yoke	10.64992*	.56442	.000	8.9815	12.3184
		Pull	.88849	.56442	.699	-.7800	2.5570
		SMED	.16041	.56442	1.000	-1.5081	1.8289
		Ucell	-.21643	.56442	1.000	-1.8849	1.4520
	Poka Yoke	5S model	-5.91561*	.56442	.000	-7.5841	-4.2471
		Actual model	-10.86931*	.56442	.000	-12.5378	-9.2008
		Cross training	-10.64992*	.56442	.000	-12.3184	-8.9815
		Pull	-9.76144*	.56442	.000	-11.4299	-8.0930
		SMED	-10.48952*	.56442	.000	-12.1580	-8.8211
		Ucell	-10.86636*	.56442	.000	-12.5348	-9.1979
	Pull	5S model	3.84583*	.56442	.000	2.1774	5.5143
		Actual model	-1.10788	.56442	.439	-2.7763	.5606
		Cross training	-.88849	.56442	.699	-2.5570	.7800
		Poka Yoke	9.76144*	.56442	.000	8.0930	11.4299
		SMED	-.72808	.56442	.857	-2.3965	.9404
		Ucell	-1.10492	.56442	.443	-2.7734	.5635
	SMED	5S model	4.57391*	.56442	.000	2.9054	6.2424
		Actual model	-.37980	.56442	.994	-2.0483	1.2887
		Cross training	-.16041	.56442	1.000	-1.8289	1.5081
		Poka Yoke	10.48952*	.56442	.000	8.8211	12.1580
		Pull	.72808	.56442	.857	-.9404	2.3965
		Ucell	-.37684	.56442	.994	-2.0453	1.2916
	Ucell	5S model	4.95075*	.56442	.000	3.2823	6.6192
		Actual model	-.00296	.56442	1.000	-1.6714	1.6655
		Cross training	.21643	.56442	1.000	-1.4520	1.8849
		Poka Yoke	10.86636*	.56442	.000	9.1979	12.5348
		Pull	1.10492	.56442	.443	-.5635	2.7734
		SMED	.37684	.56442	.994	-1.2916	2.0453
Throughput	5S model	Actual model	4.00000	2.44749	.660	-3.2350	11.2350
		Cross training	3.14286	2.44749	.859	-4.0921	10.3778
		Poka Yoke	2.28571	2.44749	.967	-4.9493	9.5207
		Pull	9.85714*	2.44749	.001	2.6222	17.0921
		SMED	1.85714	2.44749	.989	-5.3778	9.0921
		Ucell	3.92857	2.44749	.679	-3.3064	11.1635
	Actual model	5S model	-4.00000	2.44749	.660	-11.2350	3.2350
		Cross training	-.85714	2.44749	1.000	-8.0921	6.3778
		Poka Yoke	-1.71429	2.44749	.993	-8.9493	5.5207
		Pull	5.85714	2.44749	.203	-1.3778	13.0921
		SMED	-2.14286	2.44749	.976	-9.3778	5.0921
		Ucell	-.07143	2.44749	1.000	-7.3064	7.1635
	Cross training	5S model	-3.14286	2.44749	.859	-10.3778	4.0921
		Actual model	.85714	2.44749	1.000	-6.3778	8.0921

		Poka Yoke	-.85714	2.44749	1.000	-8.0921	6.3778
		Pull	6.71429	2.44749	.089	-.5207	13.9493
		SMED	-1.28571	2.44749	.998	-8.5207	5.9493
		Ucell	.78571	2.44749	1.000	-6.4493	8.0207
	Poka Yoke	5S model	-2.28571	2.44749	.967	-9.5207	4.9493
		Actual model	1.71429	2.44749	.993	-5.5207	8.9493
		Cross training	.85714	2.44749	1.000	-6.3778	8.0921
		Pull	7.57143*	2.44749	.033	.3365	14.8064
		SMED	-.42857	2.44749	1.000	-7.6635	6.8064
		Ucell	1.64286	2.44749	.994	-5.5921	8.8778
	Pull	5S model	-9.85714*	2.44749	.001	-17.0921	-2.6222
		Actual model	-5.85714	2.44749	.203	-13.0921	1.3778
		Cross training	-6.71429	2.44749	.089	-13.9493	.5207
		Poka Yoke	-7.57143*	2.44749	.033	-14.8064	-.3365
		SMED	-8.00000*	2.44749	.019	-15.2350	-.7650
		Ucell	-5.92857	2.44749	.191	-13.1635	1.3064
	SMED	5S model	-1.85714	2.44749	.989	-9.0921	5.3778
		Actual model	2.14286	2.44749	.976	-5.0921	9.3778
		Cross training	1.28571	2.44749	.998	-5.9493	8.5207
		Poka Yoke	.42857	2.44749	1.000	-6.8064	7.6635
		Pull	8.00000*	2.44749	.019	.7650	15.2350
		Ucell	2.07143	2.44749	.980	-5.1635	9.3064
	Ucell	5S model	-3.92857	2.44749	.679	-11.1635	3.3064
		Actual model	.07143	2.44749	1.000	-7.1635	7.3064
		Cross training	-.78571	2.44749	1.000	-8.0207	6.4493
		Poka Yoke	-1.64286	2.44749	.994	-8.8778	5.5921
		Pull	5.92857	2.44749	.191	-1.3064	13.1635
		SMED	-2.07143	2.44749	.980	-9.3064	5.1635

***. The mean difference is significant at the 0.05 level.**

Appendix B

All Lean models and JaamSim configuration files are uploaded to the following GitHub repository.

https://github.com/jalalpossik/Models

Below is the Java code of the HLA Master Federate and the Lean tools federate.

Java Code of the Master Federate

```
import java.io.BufferedReader;
import java.io.File;
import java.io.InputStreamReader;
import java.net.URL;

import hla.rti1516e.AttributeHandle;
import hla.rti1516e.AttributeHandleSet;
import hla.rti1516e.AttributeHandleValueMap;
import hla.rti1516e.CallbackModel;
import hla.rti1516e.InteractionClassHandle;
import hla.rti1516e.LogicalTimeFactoryFactory;
import hla.rti1516e.NullFederateAmbassador;
import hla.rti1516e.ObjectClassHandle;
import hla.rti1516e.ObjectInstanceHandle;
import hla.rti1516e.OrderType;
import hla.rti1516e.ParameterHandle;
import hla.rti1516e.ParameterHandleValueMap;
import hla.rti1516e.RTIambassador;
import hla.rti1516e.ResignAction;
import hla.rti1516e.RtiFactory;
import hla.rti1516e.RtiFactoryFactory;
import hla.rti1516e.TransportationTypeHandle;
import hla.rti1516e.encoding.DecoderException;
import hla.rti1516e.encoding.EncoderFactory;
import hla.rti1516e.encoding.HLAfloat32LE;
import hla.rti1516e.encoding.HLAinteger32LE;
import hla.rti1516e.encoding.HLAunicodeString;
import hla.rti1516e.exceptions.FederateInternalError;
import hla.rti1516e.exceptions.FederateNotExecutionMember;
import hla.rti1516e.exceptions.FederatesCurrentlyJoined;
import hla.rti1516e.exceptions.FederationExecutionAlreadyExists;
import hla.rti1516e.exceptions.FederationExecutionDoesNotExist;
import hla.rti1516e.exceptions.IllegalName;
import hla.rti1516e.exceptions.InteractionClassNotDefined;
import hla.rti1516e.exceptions.InteractionClassNotPublished;
import hla.rti1516e.exceptions.InteractionParameterNotDefined;
import hla.rti1516e.exceptions.NotConnected;
import hla.rti1516e.exceptions.RTIexception;
import hla.rti1516e.exceptions.RTIinternalError;
import hla.rti1516e.exceptions.RestoreInProgress;
import hla.rti1516e.exceptions.SaveInProgress;
import hla.rti1516e.time.HLAfloat64Time;
import hla.rti1516e.time.HLAfloat64TimeFactory;

public class MasterFederate extends NullFederateAmbassador {

  public static MasterFederate instance;
```

```
private RTIambassador _rtiAmbassador;
private EncoderFactory _encoderFactory;
private static final String FEDERATION_NAME = "HLA_Lean";
private static String RTI_HOST = "localhost";
File xmlFile = new File("C:/Users/academic1/Desktop/RTIEclipse/HLA_LM/HLA_Lean.xml");

private AttributeHandle Name;
private AttributeHandle SimTime;
private AttributeHandle MaterialBuffer;
private AttributeHandle SKU;
private AttributeHandle WIP;
private AttributeHandle DefectRate;
private AttributeHandle ProductionThroughput;
private AttributeHandle LeadTime;
private AttributeHandle SetupTime;
private AttributeHandle ProcessingTime;
private AttributeHandle TravelTime;
private AttributeHandle PlannedDownTime;
private AttributeHandle UnplannedDownTime;
private AttributeHandle DefectiveProbability;
private AttributeHandle MarketDemand;
private AttributeHandle NumberOfWorkers;

private volatile boolean reservationCompleted;
private volatile boolean reservationSucceeded;
private Object reservation = new Object();
private ObjectInstanceHandle regObjInstName;
String objectInstanceName = "Master";

private InteractionClassHandle ScenarioLoad;
private InteractionClassHandle ScenarioLoaded;
private InteractionClassHandle ScenarioError;
private InteractionClassHandle SimulationControl;
private InteractionClassHandle SMEDInteraction;
private InteractionClassHandle POKAYOKEInteraction;
private InteractionClassHandle FiveSInteraction;
private InteractionClassHandle UCELLInteraction;
private InteractionClassHandle PULLInteraction;

private ParameterHandle ScName;
private ParameterHandle FederateNameLoaded;
private ParameterHandle FederateNameError;
private ParameterHandle Action;
private ParameterHandle RealTimeFactor;
private ParameterHandle SetupTimeReduction;
private ParameterHandle DefectsReduction;
private ParameterHandle ProcessingAndDefectsReduction;
private ParameterHandle TravelTimeReduction;
private ParameterHandle PULLValues;

private ParameterHandleValueMap scenarioLoadParameters;
private HLAunicodeString scenarioLoadEncoder;
private ParameterHandleValueMap simulationControlParameters;
private HLAunicodeString simulationControlEncoder;
private ParameterHandleValueMap SMEDInteractionParameters;
private HLAunicodeString SMEDInteractionEncoder;
private ParameterHandleValueMap POKAYOKEInteractionParameters;
private HLAunicodeString POKAYOKEInteractionEncoder;
private ParameterHandleValueMap FiveSInteractionParameters;
private HLAunicodeString FiveSInteractionEncoder;
private ParameterHandleValueMap UCELLInteractionParameters;
```

```java
private HLAunicodeString UCELLInteractionEncoder;
private ParameterHandleValueMap PULLInteractionParameters;
private HLAunicodeString PULLInteractionEncoder;
private ParameterHandleValueMap realTimeFactorParameter;
private HLAunicodeString realTimeFactorEncoder;

double simTime = 0;
double leadTime = 0;
int productionThroughput = 0;
double defectRate = 0;
int wWIP = 0;
int x = 0;

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
RealTimeChart LeadTimeChart;
RealTimeChart WIPChart;
RealTimeChart ProductionThroughputChart;
RealTimeChart DefectRateChart;

//////////
// Main //
//////////

public static void main(String[] args) throws Exception {
  instance = new MasterFederate();
  instance.execute();

}

/////////////
// Execute //
/////////////

private void execute() throws Exception {

  try {

    //////////////////////////////////////
    // Get RTI Ambassador Host and port //
    //////////////////////////////////////

    try {
      RtiFactory rtiFactory = RtiFactoryFactory.getRtiFactory();
      _rtiAmbassador = rtiFactory.getRtiAmbassador();
      _encoderFactory = rtiFactory.getEncoderFactory();
    } catch (Exception e) {
      System.out.println("Unable to create RTI ambassador.");
      return;
    }

    _rtiAmbassador.connect(this, CallbackModel.HLA_IMMEDIATE, RTI_HOST);

    ///////////////////////////////////////////////////////////////////////
    // Create the Federation using the Federation Name and the XML File.  //
    ///////////////////////////////////////////////////////////////////////


    try {

      // Clean up old federation
```

```java
          _rtiAmbassador.destroyFederationExecution(FEDERATION_NAME);
        } catch (FederatesCurrentlyJoined ignored) {
        } catch (FederationExecutionDoesNotExist ignored) {
        }

        try {
          _rtiAmbassador.createFederationExecution(FEDERATION_NAME, new URL[] { xmlFile.toURL()
},
              "HLAfloat64Time");
        } catch (FederationExecutionAlreadyExists ignored) {
        }

        ////////////////////////////////////////////////////////////////
        // Joining any existing Federation, Argument1 is the Federate //
        ////////////////////////////////////////////////////////////////

        _rtiAmbassador.joinFederationExecution("Master",    FEDERATION_NAME,    new    URL[]    {
xmlFile.toURL() });

        ////////////////////////////////////
        // Objects/Attributes Declaration //
        ////////////////////////////////////

        ObjectClassHandle Scenario = _rtiAmbassador.getObjectClassHandle("Scenario");
        AttributeHandleSet attributeSet = _rtiAmbassador.getAttributeHandleSetFactory().create();

        Name = _rtiAmbassador.getAttributeHandle(Scenario, "Name");
        SimTime = _rtiAmbassador.getAttributeHandle(Scenario, "SimTime");
        MaterialBuffer = _rtiAmbassador.getAttributeHandle(Scenario, "MaterialBuffer");
        SKU = _rtiAmbassador.getAttributeHandle(Scenario, "SKU");
        WIP = _rtiAmbassador.getAttributeHandle(Scenario, "WIP");
        ProductionThroughput            =            _rtiAmbassador.getAttributeHandle(Scenario,
"ProductionThroughput");
        DefectRate = _rtiAmbassador.getAttributeHandle(Scenario, "DefectRate");
        LeadTime = _rtiAmbassador.getAttributeHandle(Scenario, "LeadTime");
        SetupTime = _rtiAmbassador.getAttributeHandle(Scenario, "SetupTime");
        ProcessingTime = _rtiAmbassador.getAttributeHandle(Scenario, "ProcessingTime");
        TravelTime = _rtiAmbassador.getAttributeHandle(Scenario, "TravelTime");
        PlannedDownTime = _rtiAmbassador.getAttributeHandle(Scenario, "PlannedDownTime");
        UnplannedDownTime = _rtiAmbassador.getAttributeHandle(Scenario, "UnplannedDownTime");
        DefectiveProbability            =            _rtiAmbassador.getAttributeHandle(Scenario,
"DefectiveProbability");
        MarketDemand = _rtiAmbassador.getAttributeHandle(Scenario, "MarketDemand");
        NumberOfWorkers = _rtiAmbassador.getAttributeHandle(Scenario, "NumberOfWorkers");

        attributeSet.add(Name);
        attributeSet.add(SimTime);
        attributeSet.add(MaterialBuffer);
        attributeSet.add(SKU);
        attributeSet.add(WIP);
        attributeSet.add(ProductionThroughput);
        attributeSet.add(DefectRate);
        attributeSet.add(LeadTime);
        attributeSet.add(SetupTime);
        attributeSet.add(ProcessingTime);
        attributeSet.add(TravelTime);
        attributeSet.add(PlannedDownTime);
        attributeSet.add(UnplannedDownTime);
        attributeSet.add(DefectiveProbability);
        attributeSet.add(MarketDemand);
        attributeSet.add(NumberOfWorkers);
```

```
	// Subscribe and publish objects
	_rtiAmbassador.subscribeObjectClassAttributes(Scenario, attributeSet);
	_rtiAmbassador.publishObjectClassAttributes(Scenario, attributeSet);



	/////////////////////////////////////////
	// Interactions/Parameters Declaration //
	/////////////////////////////////////////

	ScenarioLoad = _rtiAmbassador.getInteractionClassHandle("ScenarioLoad");
	ScName = _rtiAmbassador.getParameterHandle(ScenarioLoad, "ScName");

	ScenarioLoaded = _rtiAmbassador.getInteractionClassHandle("ScenarioLoaded");
	FederateNameLoaded = _rtiAmbassador.getParameterHandle(ScenarioLoaded, "FederateName");

	ScenarioError = _rtiAmbassador.getInteractionClassHandle("ScenarioError");
	FederateNameError = _rtiAmbassador.getParameterHandle(ScenarioError, "FederateName");

	SimulationControl = _rtiAmbassador.getInteractionClassHandle("SimulationControl");
	Action = _rtiAmbassador.getParameterHandle(SimulationControl, "Action");
	RealTimeFactor = _rtiAmbassador.getParameterHandle(SimulationControl, "RealTimeFactor");

	SMEDInteraction = _rtiAmbassador.getInteractionClassHandle("SMEDInteraction");
	SetupTimeReduction = _rtiAmbassador.getParameterHandle(SMEDInteraction,
"SetupTimeReduction");

	POKAYOKEInteraction = _rtiAmbassador.getInteractionClassHandle("POKAYOKEInteraction");
	DefectsReduction = _rtiAmbassador.getParameterHandle(POKAYOKEInteraction,
"DefectsReduction");

	FiveSInteraction = _rtiAmbassador.getInteractionClassHandle("FiveSInteraction");
	ProcessingAndDefectsReduction = _rtiAmbassador.getParameterHandle(FiveSInteraction,
		"ProcessingAndDefectsReduction");

	UCELLInteraction = _rtiAmbassador.getInteractionClassHandle("UCELLInteraction");
	TravelTimeReduction = _rtiAmbassador.getParameterHandle(UCELLInteraction,
"TravelTimeReduction");

	PULLInteraction = _rtiAmbassador.getInteractionClassHandle("PULLInteraction");
	PULLValues = _rtiAmbassador.getParameterHandle(PULLInteraction, "PULLValues");

	// Subscribe and publish interactions

	_rtiAmbassador.publishInteractionClass(ScenarioLoad);
	_rtiAmbassador.publishInteractionClass(SimulationControl);
	_rtiAmbassador.publishInteractionClass(SMEDInteraction);
	_rtiAmbassador.publishInteractionClass(POKAYOKEInteraction);
	_rtiAmbassador.publishInteractionClass(FiveSInteraction);
	_rtiAmbassador.publishInteractionClass(UCELLInteraction);
	_rtiAmbassador.publishInteractionClass(PULLInteraction);

	_rtiAmbassador.subscribeInteractionClass(ScenarioLoaded);
	_rtiAmbassador.subscribeInteractionClass(ScenarioError);

	HLAfloat64TimeFactory _logicalTimeFactory = (HLAfloat64TimeFactory)
LogicalTimeFactoryFactory
		.getLogicalTimeFactory(HLAfloat64TimeFactory.NAME);
	HLAfloat64Time _logicalTime = _logicalTimeFactory.makeInitial();
```

```
    ////////////////////
    // Parameters Data //
    ////////////////////

    scenarioLoadParameters = _rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    scenarioLoadEncoder = _encoderFactory.createHLAunicodeString();

    simulationControlParameters =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    simulationControlEncoder = _encoderFactory.createHLAunicodeString();

    realTimeFactorParameter = _rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    realTimeFactorEncoder = _encoderFactory.createHLAunicodeString();

    SMEDInteractionParameters =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    SMEDInteractionEncoder = _encoderFactory.createHLAunicodeString();

    POKAYOKEInteractionParameters =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    POKAYOKEInteractionEncoder = _encoderFactory.createHLAunicodeString();

    UCELLInteractionParameters =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    UCELLInteractionEncoder = _encoderFactory.createHLAunicodeString();

    PULLInteractionParameters =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    PULLInteractionEncoder = _encoderFactory.createHLAunicodeString();

    FiveSInteractionParameters =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    FiveSInteractionEncoder = _encoderFactory.createHLAunicodeString();

    ///////////////////////////////////////
    // Object reservation and registration //
    ///////////////////////////////////////

    do {

      try {
        reservationCompleted = false;
        _rtiAmbassador.reserveObjectInstanceName(objectInstanceName);
        // Thread.sleep(3000);
        synchronized (reservation) {
          while (!reservationCompleted) {
            reservation.wait();
          }
        }
      } catch (IllegalName e) {
        System.out.println("Illegal name. Try again.");
      } catch (RTIexception e) {
        System.out.println("RTI exception when reserving name: " + e.getMessage());
        return;
      }
    } while (!reservationSucceeded);

    regObjInstName = _rtiAmbassador.registerObjectInstance(Scenario, objectInstanceName);
    LeadTimeChart = new RealTimeChart("Lead Time", "Simulation Hours", "Hours");
    WIPChart = new RealTimeChart("WIP", "Simulation Hours", "Total WIP");
    DefectRateChart = new RealTimeChart("Defect Rate", "Simulation Hours", "Defects per day");
```

```
      ProductionThroughputChart = new RealTimeChart("Production Throughput", "Simulation Hours",
"Products per day");

      MasterInterface.execute();

    } catch (Exception e) {
      throw new RuntimeException(e);
    }

  }

  @Override
  public final void objectInstanceNameReservationSucceeded(String objectName) {
    synchronized (reservation) {
      reservationCompleted = true;
      reservationSucceeded = true;
      reservation.notifyAll();
    }
  }

  @Override
  public final void objectInstanceNameReservationFailed(String objectName) {
    synchronized (reservation) {
      reservationCompleted = true;
      reservationSucceeded = false;
      reservation.notifyAll();
    }
  }

  @Override
  public void removeObjectInstance(ObjectInstanceHandle theObject, byte[] userSuppliedTag,
OrderType sentOrdering,
      SupplementalRemoveInfo removeInfo) {
      }

  public void loadScenarios(String scenarioName)
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {
    System.out.println(scenarioName);
    scenarioLoadEncoder.setValue(scenarioName);
    scenarioLoadParameters.put(ScName, scenarioLoadEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(ScenarioLoad, scenarioLoadParameters, null);

  }

  public void setupTime(String scenarioName)
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {
    System.out.println(scenarioName);
    scenarioLoadEncoder.setValue(scenarioName);
    scenarioLoadParameters.put(ScName, scenarioLoadEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(ScenarioLoad, scenarioLoadParameters, null);

  }

  public void startSimulation()
```

```
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {
    realTimeFactorEncoder.setValue(Double.toString(MasterController.RealTimeFactor));
    realTimeFactorParameter.put(RealTimeFactor, realTimeFactorEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(SimulationControl, realTimeFactorParameter, null);

    simulationControlEncoder.setValue("START");
    simulationControlParameters.put(Action, simulationControlEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(SimulationControl, simulationControlParameters, null);

  }

  public void pauseSimulation()
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {
    simulationControlEncoder.setValue("PAUSE");
    simulationControlParameters.put(Action, simulationControlEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(SimulationControl, simulationControlParameters, null);

  }

  public void stopSimulation()
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {
    simulationControlEncoder.setValue("STOP");
    simulationControlParameters.put(Action, simulationControlEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(SimulationControl, simulationControlParameters, null);

  }

  public void simulationSpeed()
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {

    System.out.println(MasterController.RealTimeFactor);

    realTimeFactorEncoder.setValue(Double.toString(MasterController.RealTimeFactor));
    realTimeFactorParameter.put(RealTimeFactor, realTimeFactorEncoder.toByteArray());
    _rtiAmbassador.sendInteraction(SimulationControl, realTimeFactorParameter, null);

  }

  public void sendLeanToolsInteractions(String s)
      throws InteractionClassNotPublished, InteractionParameterNotDefined,
InteractionClassNotDefined,
      SaveInProgress, RestoreInProgress, FederateNotExecutionMember, NotConnected,
RTIinternalError {

    if (s.equals("SMED")) {
      SMEDInteractionEncoder.setValue(MasterController.SetupTimeReduction);
      SMEDInteractionParameters.put(SetupTimeReduction, SMEDInteractionEncoder.toByteArray());
      _rtiAmbassador.sendInteraction(SMEDInteraction, SMEDInteractionParameters, null);
    }
```

```
    if (s.equals("POKAYOKE")) {

      POKAYOKEInteractionEncoder.setValue(MasterController.DefectsReduction);
      POKAYOKEInteractionParameters.put(DefectsReduction,
POKAYOKEInteractionEncoder.toByteArray());
      _rtiAmbassador.sendInteraction(POKAYOKEInteraction, POKAYOKEInteractionParameters, null);
    }

    if (s.equals("UCELL")) {
      UCELLInteractionEncoder.setValue(MasterController.TravelTimeReduction);
      UCELLInteractionParameters.put(TravelTimeReduction,
UCELLInteractionEncoder.toByteArray());
      _rtiAmbassador.sendInteraction(UCELLInteraction, UCELLInteractionParameters, null);
    }

    if (s.equals("5S")) {
      FiveSInteractionEncoder.setValue(MasterController.ProcessingAndDefectsReduction);
      FiveSInteractionParameters.put(ProcessingAndDefectsReduction,
FiveSInteractionEncoder.toByteArray());
      _rtiAmbassador.sendInteraction(FiveSInteraction, FiveSInteractionParameters, null);
    }

    if (s.equals("PULL")) {
      PULLInteractionEncoder.setValue(MasterController.PULLValues);
      PULLInteractionParameters.put(PULLValues, PULLInteractionEncoder.toByteArray());
      _rtiAmbassador.sendInteraction(PULLInteraction, PULLInteractionParameters, null);
    }
  }

  void updateAttributes(String att) throws Exception {

    String scSetupTime = MasterController.SetupTime;
    String scProcessingTime = MasterController.ProcessingTime;
    String scTravelTime = MasterController.TravelTime;
    String scPlannedDownTime = MasterController.PlannedDownTime;
    String scUnplannedDownTime = MasterController.UnplannedDownTime;
    String scDefectiveProbability = MasterController.DefectiveProbability;
    String scMarketDemand = MasterController.MarketDemand;
    String scNumberOfWorkers = MasterController.NumberOfWorkers;

    AttributeHandleValueMap attributeValues =
_rtiAmbassador.getAttributeHandleValueMapFactory().create(1);

    HLAunicodeString scSetupTimeEncoder = _encoderFactory.createHLAunicodeString(scSetupTime);
    HLAunicodeString scProcessingTimeEncoder =
_encoderFactory.createHLAunicodeString(scProcessingTime);
    HLAunicodeString scTravelTimeEncoder =
_encoderFactory.createHLAunicodeString(scTravelTime);
    HLAunicodeString scPlannedDownTimeEncoder =
_encoderFactory.createHLAunicodeString(scPlannedDownTime);
    HLAunicodeString scUnplannedDownTimeEncoder =
_encoderFactory.createHLAunicodeString(scUnplannedDownTime);
    HLAunicodeString scDefectiveProbabilityEncoder =
_encoderFactory.createHLAunicodeString(scDefectiveProbability);
    HLAunicodeString scMarketDemandEncoder =
_encoderFactory.createHLAunicodeString(scMarketDemand);
    HLAunicodeString scNumberOfWorkersEncoder =
_encoderFactory.createHLAunicodeString(scNumberOfWorkers);
    //
```

```
    if (att.equals("SetupTime"))
      attributeValues.put(SetupTime, scSetupTimeEncoder.toByteArray());
    if (att.equals("ProcessingTime"))
      attributeValues.put(ProcessingTime, scProcessingTimeEncoder.toByteArray());
    if (att.equals("TravelTime"))
      attributeValues.put(TravelTime, scTravelTimeEncoder.toByteArray());
    if (att.equals("PlannedDownTime"))
      attributeValues.put(PlannedDownTime, scPlannedDownTimeEncoder.toByteArray());
    if (att.equals("UnplannedDownTime"))
      attributeValues.put(UnplannedDownTime, scUnplannedDownTimeEncoder.toByteArray());
    if (att.equals("DefectiveProbability"))
      attributeValues.put(DefectiveProbability, scDefectiveProbabilityEncoder.toByteArray());
    if (att.equals("MarketDemand"))
      attributeValues.put(MarketDemand, scMarketDemandEncoder.toByteArray());
    if (att.equals("NumberOfWorkers"))
      attributeValues.put(NumberOfWorkers, scNumberOfWorkersEncoder.toByteArray());

    _rtiAmbassador.updateAttributeValues(regObjInstName, attributeValues, null);

  }

  void disconnect() throws Exception {
    _rtiAmbassador.resignFederationExecution(ResignAction.DELETE_OBJECTS_THEN_DIVEST);
    _rtiAmbassador.destroyFederationExecution(FEDERATION_NAME);
    _rtiAmbassador.disconnect();
    _rtiAmbassador = null;
  }

  @Override
  public void discoverObjectInstance(ObjectInstanceHandle theObject, ObjectClassHandle
theObjectClass,
      String objectName) throws FederateInternalError {
  }

  @Override
  public void reflectAttributeValues(ObjectInstanceHandle theObject, AttributeHandleValueMap
theAttributes,
      byte[] userSuppliedTag, OrderType sentOrdering, TransportationTypeHandle theTransport,
      SupplementalReflectInfo reflectInfo) {
    try {
      final HLAunicodeString stringDecoder = _encoderFactory.createHLAunicodeString();
      final HLAfloat32LE floatDecoder = _encoderFactory.createHLAfloat32LE();
      final HLAinteger32LE intDecoder = _encoderFactory.createHLAinteger32LE();

      stringDecoder.decode(theAttributes.get(Name));

      if (theAttributes.containsKey(SimTime)) {
        floatDecoder.decode(theAttributes.get(SimTime));
        simTime = floatDecoder.getValue();

      }

      if (theAttributes.containsKey(WIP)) {
        intDecoder.decode(theAttributes.get(WIP));
        wWIP = intDecoder.getValue();
        WIPChart.DataAdd(stringDecoder.getValue(), simTime, wWIP);
      }

      if (theAttributes.containsKey(DefectRate)) {
        floatDecoder.decode(theAttributes.get(DefectRate));
        defectRate = floatDecoder.getValue();
```

```
        DefectRateChart.DataAdd(stringDecoder.getValue(), simTime, defectRate);
      }

      if (theAttributes.containsKey(ProductionThroughput)) {
        intDecoder.decode(theAttributes.get(ProductionThroughput));
        productionThroughput = intDecoder.getValue();
        ProductionThroughputChart.DataAdd(stringDecoder.getValue(),                    simTime,
productionThroughput);
      }

      if (theAttributes.containsKey(LeadTime)) {
        floatDecoder.decode(theAttributes.get(LeadTime));
        leadTime = floatDecoder.getValue() / 3600;
        LeadTimeChart.DataAdd(stringDecoder.getValue(), simTime, leadTime);
      }

      if (theAttributes.containsKey(MaterialBuffer)) {
        intDecoder.decode(theAttributes.get(MaterialBuffer));
      }

      if (theAttributes.containsKey(SKU)) {
        intDecoder.decode(theAttributes.get(SKU));
      }

    } catch (DecoderException | InterruptedException e) {

      e.printStackTrace();
    } catch (Exception e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    }
  }

  @Override
  public final void provideAttributeValueUpdate(ObjectInstanceHandle theObject,
AttributeHandleSet theAttributes,
      byte[] userSuppliedTag) {
      }

  @Override
  public void receiveInteraction(InteractionClassHandle interactionClass,
ParameterHandleValueMap theParameters,
      byte[] userSuppliedTag, OrderType sentOrdering, TransportationTypeHandle theTransport,
      SupplementalReceiveInfo receiveInfo) throws FederateInternalError {
    final HLAunicodeString stringDecoder = _encoderFactory.createHLAunicodeString();

    try {

      if (interactionClass.equals(ScenarioLoaded)) {
        stringDecoder.decode(theParameters.get(FederateNameLoaded));
              } else if (interactionClass.equals(ScenarioError)) {
        stringDecoder.decode(theParameters.get(FederateNameError));
              }

    } catch (DecoderException e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    }
  }
}
```

Lean Tools Federates

```
package com.jaamsim.ui;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.URL;

import com.jaamsim.Graphics.DisplayEntity;
import com.jaamsim.basicsim.Simulation;

import hla.rti1516e.AttributeHandle;
import hla.rti1516e.AttributeHandleSet;
import hla.rti1516e.AttributeHandleValueMap;
import hla.rti1516e.CallbackModel;
import hla.rti1516e.InteractionClassHandle;
import hla.rti1516e.LogicalTime;
import hla.rti1516e.LogicalTimeFactoryFactory;
import hla.rti1516e.NullFederateAmbassador;
import hla.rti1516e.ObjectClassHandle;
import hla.rti1516e.ObjectInstanceHandle;
import hla.rti1516e.OrderType;
import hla.rti1516e.ParameterHandle;
import hla.rti1516e.ParameterHandleValueMap;
import hla.rti1516e.RTIambassador;
import hla.rti1516e.ResignAction;
import hla.rti1516e.RtiFactory;
import hla.rti1516e.RtiFactoryFactory;
import hla.rti1516e.TransportationTypeHandle;
import hla.rti1516e.encoding.DecoderException;
import hla.rti1516e.encoding.EncoderFactory;
import hla.rti1516e.encoding.HLAfloat32LE;
import hla.rti1516e.encoding.HLAinteger32LE;
import hla.rti1516e.encoding.HLAunicodeString;
import hla.rti1516e.exceptions.FederateInternalError;
import hla.rti1516e.exceptions.FederateNotExecutionMember;
import hla.rti1516e.exceptions.IllegalName;
import hla.rti1516e.exceptions.InTimeAdvancingState;
import hla.rti1516e.exceptions.InteractionClassNotDefined;
import hla.rti1516e.exceptions.InteractionClassNotPublished;
import hla.rti1516e.exceptions.InteractionParameterNotDefined;
import hla.rti1516e.exceptions.InvalidLogicalTime;
import hla.rti1516e.exceptions.LogicalTimeAlreadyPassed;
import hla.rti1516e.exceptions.NotConnected;
import hla.rti1516e.exceptions.RTIexception;
import hla.rti1516e.exceptions.RTIinternalError;
import hla.rti1516e.exceptions.RequestForTimeConstrainedPending;
import hla.rti1516e.exceptions.RequestForTimeRegulationPending;
import hla.rti1516e.exceptions.RestoreInProgress;
import hla.rti1516e.exceptions.SaveInProgress;
import hla.rti1516e.time.HLAfloat64Interval;
import hla.rti1516e.time.HLAfloat64Time;
import hla.rti1516e.time.HLAfloat64TimeFactory;
```

```
public class Federate extends NullFederateAmbassador {
  private GUIFrame gui;
  public static Federate instance;

  private RTIambassador _rtiAmbassador;
  private EncoderFactory _encoderFactory;
  private static final String FEDERATION_NAME = "HLA_Lean";
  private static String RTI_HOST = "localhost";
  File xmlFile = new File("C:/Users/academic1/Desktop/RTIEclipse/HLA_LM/HLA_Lean.xml");
  private AttributeHandle Name;
  private AttributeHandle SimTime;
  private AttributeHandle MaterialBuffer;
  private AttributeHandle SKU;
  private AttributeHandle WIP;
  private AttributeHandle DefectRate;
  private AttributeHandle ProductionThroughput;
  private AttributeHandle LeadTime;
  private AttributeHandle SetupTime;
  private AttributeHandle ProcessingTime;
  private AttributeHandle TravelTime;
  private AttributeHandle PlannedDownTime;
  private AttributeHandle UnplannedDownTime;
  private AttributeHandle DefectiveProbability;
  private AttributeHandle MarketDemand;
  private AttributeHandle NumberOfWorkers;

  private volatile boolean reservationCompleted;
  private volatile boolean reservationSucceeded;
  private Object reservation = new Object();
  private ObjectInstanceHandle regObjInstName;
  String objectInstanceName;
  static BufferedWriter writer = null;
  private static String[] guiArgs;
  public int jaamsimPort = 0;
  private int scWIP = 0;
  private float scLeadTime = 0;
  private float scDefectRate = 0;
  private int scProductionThroughput = 0;

  private boolean trEnabled = false;
  private boolean tcEnabled = false;

  BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

  private InteractionClassHandle ScenarioLoad;
  private InteractionClassHandle ScenarioLoaded;
  private InteractionClassHandle ScenarioError;
  private InteractionClassHandle SimulationControl;
  private InteractionClassHandle SMEDInteraction;
  private InteractionClassHandle POKAYOKEInteraction;
  private InteractionClassHandle FiveSInteraction;
  private InteractionClassHandle UCELLInteraction;
  private InteractionClassHandle PULLInteraction;

  private ParameterHandle ScName;
  private ParameterHandle FederateNameLoaded;
  private ParameterHandle Action;
  private ParameterHandle RealTimeFactor;
  private ParameterHandle SetupTimeReduction;
  private ParameterHandle DefectsReduction;
  private ParameterHandle ProcessingAndDefectsReduction;
```

```
  private ParameterHandle TravelTimeReduction;
  private ParameterHandle PULLValues;

  private HLAfloat64TimeFactory _logicalTimeFactory;
  private HLAfloat64Time _logicalTime;
  private HLAfloat64Interval _lookahead;
  private String federateName;

  String setupTime;
  String processingTime;
  String travelTime;
  String plannedDownTime;
  String unplannedDownTime;
  String defectiveProbability;
  String marketDemand;
  String numberOfWorkers;

  public Federate(String federateName) throws Exception {
    instance = this;
    this.federateName = federateName;
    this.objectInstanceName = "Senario_" + federateName;

    //////////////////////////////////////
    // Get RTI Ambassador Host and port //
    //////////////////////////////////////

    RtiFactory rtiFactory = RtiFactoryFactory.getRtiFactory();
    _rtiAmbassador = rtiFactory.getRtiAmbassador();
    _encoderFactory = rtiFactory.getEncoderFactory();
    _rtiAmbassador.connect(this, CallbackModel.HLA_IMMEDIATE, RTI_HOST);

    ////////////////////////////////////////////////////////////////
    // Joining any existing Federation, Argument1 is the Federate //
    ////////////////////////////////////////////////////////////////


    _rtiAmbassador.joinFederationExecution(federateName,   FEDERATION_NAME,   new   URL[]   {
xmlFile.toURL() });

    ////////////////////////////////////
    // Objects/Attributes Declaration //
    ////////////////////////////////////

    ObjectClassHandle Scenario = _rtiAmbassador.getObjectClassHandle("Scenario");
    AttributeHandleSet attributeSet = _rtiAmbassador.getAttributeHandleSetFactory().create();

    Name = _rtiAmbassador.getAttributeHandle(Scenario, "Name");
    SimTime = _rtiAmbassador.getAttributeHandle(Scenario, "SimTime");
    MaterialBuffer = _rtiAmbassador.getAttributeHandle(Scenario, "MaterialBuffer");
    SKU = _rtiAmbassador.getAttributeHandle(Scenario, "SKU");
    WIP = _rtiAmbassador.getAttributeHandle(Scenario, "WIP");
    ProductionThroughput = _rtiAmbassador.getAttributeHandle(Scenario,
"ProductionThroughput");
    DefectRate = _rtiAmbassador.getAttributeHandle(Scenario, "DefectRate");
    LeadTime = _rtiAmbassador.getAttributeHandle(Scenario, "LeadTime");
    SetupTime = _rtiAmbassador.getAttributeHandle(Scenario, "SetupTime");
    ProcessingTime = _rtiAmbassador.getAttributeHandle(Scenario, "ProcessingTime");
    TravelTime = _rtiAmbassador.getAttributeHandle(Scenario, "TravelTime");
    PlannedDownTime = _rtiAmbassador.getAttributeHandle(Scenario, "PlannedDownTime");
    UnplannedDownTime = _rtiAmbassador.getAttributeHandle(Scenario, "UnplannedDownTime");
```

```
    DefectiveProbability = _rtiAmbassador.getAttributeHandle(Scenario,
"DefectiveProbability");
    MarketDemand = _rtiAmbassador.getAttributeHandle(Scenario, "MarketDemand");
    NumberOfWorkers = _rtiAmbassador.getAttributeHandle(Scenario, "NumberOfWorkers");

    attributeSet.add(Name);
    attributeSet.add(SimTime);
    attributeSet.add(MaterialBuffer);
    attributeSet.add(SKU);
    attributeSet.add(WIP);
    attributeSet.add(ProductionThroughput);
    attributeSet.add(DefectRate);
    attributeSet.add(LeadTime);
    attributeSet.add(SetupTime);
    attributeSet.add(ProcessingTime);
    attributeSet.add(TravelTime);
    attributeSet.add(PlannedDownTime);
    attributeSet.add(UnplannedDownTime);
    attributeSet.add(DefectiveProbability);
    attributeSet.add(MarketDemand);
    attributeSet.add(NumberOfWorkers);

    // Subscribe and publish objects

    _rtiAmbassador.subscribeObjectClassAttributes(Scenario, attributeSet);
    _rtiAmbassador.publishObjectClassAttributes(Scenario, attributeSet);

    ////////////////////////////////////////
    // Interactions/Parameters Declaration //
    ////////////////////////////////////////

    ScenarioLoad = _rtiAmbassador.getInteractionClassHandle("ScenarioLoad");
    ScName = _rtiAmbassador.getParameterHandle(ScenarioLoad, "ScName");

    ScenarioLoaded = _rtiAmbassador.getInteractionClassHandle("ScenarioLoaded");
    FederateNameLoaded = _rtiAmbassador.getParameterHandle(ScenarioLoaded, "FederateName");

    ScenarioError = _rtiAmbassador.getInteractionClassHandle("ScenarioError");
    _rtiAmbassador.getParameterHandle(ScenarioError, "FederateName");

    SimulationControl = _rtiAmbassador.getInteractionClassHandle("SimulationControl");
    Action = _rtiAmbassador.getParameterHandle(SimulationControl, "Action");
    RealTimeFactor = _rtiAmbassador.getParameterHandle(SimulationControl, "RealTimeFactor");

    SMEDInteraction = _rtiAmbassador.getInteractionClassHandle("SMEDInteraction");
    SetupTimeReduction = _rtiAmbassador.getParameterHandle(SMEDInteraction,
"SetupTimeReduction");

    POKAYOKEInteraction = _rtiAmbassador.getInteractionClassHandle("POKAYOKEInteraction");
    DefectsReduction = _rtiAmbassador.getParameterHandle(POKAYOKEInteraction,
"DefectsReduction");

    FiveSInteraction = _rtiAmbassador.getInteractionClassHandle("FiveSInteraction");
    ProcessingAndDefectsReduction = _rtiAmbassador.getParameterHandle(FiveSInteraction,
        "ProcessingAndDefectsReduction");

    UCELLInteraction = _rtiAmbassador.getInteractionClassHandle("UCELLInteraction");
    TravelTimeReduction = _rtiAmbassador.getParameterHandle(UCELLInteraction,
"TravelTimeReduction");

    PULLInteraction = _rtiAmbassador.getInteractionClassHandle("PULLInteraction");
```

```
  PULLValues = _rtiAmbassador.getParameterHandle(PULLInteraction, "PULLValues");

  // Subscribe and publish interactions

  _rtiAmbassador.subscribeInteractionClass(ScenarioLoad);
  _rtiAmbassador.subscribeInteractionClass(SimulationControl);
  _rtiAmbassador.subscribeInteractionClass(SMEDInteraction);
  _rtiAmbassador.subscribeInteractionClass(POKAYOKEInteraction);
  _rtiAmbassador.subscribeInteractionClass(FiveSInteraction);
  _rtiAmbassador.subscribeInteractionClass(UCELLInteraction);
  _rtiAmbassador.subscribeInteractionClass(PULLInteraction);

  _rtiAmbassador.publishInteractionClass(ScenarioLoaded);
  _rtiAmbassador.publishInteractionClass(ScenarioError);

  ////////////////////////////////////////
  // Object reservation and registration //
  ////////////////////////////////////////

  do {

    try {
      reservationCompleted = false;
      _rtiAmbassador.reserveObjectInstanceName(objectInstanceName);
      // Thread.sleep(3000);
      synchronized (reservation) {
        while (!reservationCompleted) {
          reservation.wait();
        }
      }
    } catch (IllegalName e) {
      System.out.println("Illegal name. Try again.");
    } catch (RTIexception e) {
      System.out.println("RTI exception when reserving name: " + e.getMessage());
      return;
    }
  } while (!reservationSucceeded);

  regObjInstName = _rtiAmbassador.registerObjectInstance(Scenario, objectInstanceName);

  /////////////////////
  // Attributes Data //
  /////////////////////

  System.out.print("My Scenario Name is " + federateName + "\r");
}

@Override
public final void objectInstanceNameReservationSucceeded(String objectName) {
  synchronized (reservation) {
    reservationCompleted = true;
    reservationSucceeded = true;
    reservation.notifyAll();
  }
}

@Override
public final void objectInstanceNameReservationFailed(String objectName) {
  synchronized (reservation) {
    reservationCompleted = true;
    reservationSucceeded = false;
```

```
      reservation.notifyAll();
    }
  }

  @Override
  public void removeObjectInstance(ObjectInstanceHandle theObject, byte[] userSuppliedTag,
OrderType sentOrdering,
      SupplementalRemoveInfo removeInfo) {
  }

  @Override
  public void discoverObjectInstance(ObjectInstanceHandle theObject, ObjectClassHandle
theObjectClass,
      String objectName) throws FederateInternalError {
  }

  @Override
  public void turnUpdatesOnForObjectInstance(ObjectInstanceHandle theObject,
AttributeHandleSet theAttributes)
      throws FederateInternalError {
  }

  @Override
  public void reflectAttributeValues(ObjectInstanceHandle theObject, AttributeHandleValueMap
theAttributes,
      byte[] userSuppliedTag, OrderType sentOrdering, TransportationTypeHandle theTransport,
      SupplementalReflectInfo reflectInfo)

  {
    try {

      final HLAunicodeString stringDecoder = _encoderFactory.createHLAunicodeString();

      if (theAttributes.containsKey(SetupTime)) {
        stringDecoder.decode(theAttributes.get(SetupTime));
        setupTime = stringDecoder.getValue();

        File SetupTimeFile = new File(
            "C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/" + federateName +
"/SetupTime.txt");
        FileWriter fw = new FileWriter(SetupTimeFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(setupTime);
        bw.close();

      }

      if (theAttributes.containsKey(ProcessingTime)) {
        stringDecoder.decode(theAttributes.get(ProcessingTime));
        processingTime = stringDecoder.getValue();

        File ProcessingTimeFile = new File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-
May19/"
            + federateName + "/ProcessingTime.txt");
        FileWriter fw = new FileWriter(ProcessingTimeFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(processingTime);
        bw.close();
```

```
      }

      if (theAttributes.containsKey(TravelTime)) {
        stringDecoder.decode(theAttributes.get(TravelTime));
        travelTime = stringDecoder.getValue();

        File TravelTimeFile = new File(
            "C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/" + federateName +
"/TravelTime.txt");
        FileWriter fw = new FileWriter(TravelTimeFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(travelTime);
        bw.close();

      }

      if (theAttributes.containsKey(PlannedDownTime)) {
        stringDecoder.decode(theAttributes.get(PlannedDownTime));
        plannedDownTime = stringDecoder.getValue();

        File PlannedDownTimeFile = new File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-
May19/"
            + federateName + "/PlannedDownTime.txt");
        FileWriter fw = new FileWriter(PlannedDownTimeFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(plannedDownTime);
        bw.close();

      }

      if (theAttributes.containsKey(UnplannedDownTime)) {
        stringDecoder.decode(theAttributes.get(UnplannedDownTime));
        unplannedDownTime = stringDecoder.getValue();

        File UnplannedDownTimeFile = new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
            + federateName + "/UnplannedDownTime.txt");
        FileWriter fw = new FileWriter(UnplannedDownTimeFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(unplannedDownTime);
        bw.close();

      }

      if (theAttributes.containsKey(DefectiveProbability)) {
        stringDecoder.decode(theAttributes.get(DefectiveProbability));
        defectiveProbability = stringDecoder.getValue();

        File DefectiveProbabilityFile = new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
            + federateName + "/DefectiveProbability.txt");
        FileWriter fw = new FileWriter(DefectiveProbabilityFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(defectiveProbability);
        bw.close();

      }
```

```
      if (theAttributes.containsKey(NumberOfWorkers)) {
        stringDecoder.decode(theAttributes.get(NumberOfWorkers));
        numberOfWorkers = stringDecoder.getValue();

        if (federateName.equals("CrossTraining")) {
          String arr[] = numberOfWorkers.split("\t");
          int nbOfWorkers = Integer.parseInt(arr[0]) + Integer.parseInt(arr[1]) +
Integer.parseInt(arr[2])
              + Integer.parseInt(arr[3]);

          File NumberOfWorkersFile = new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
              + federateName + "/NumberOfWorkers.txt");
          FileWriter fw = new FileWriter(NumberOfWorkersFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(Integer.toString(nbOfWorkers));
          bw.close();
        } else {
          File NumberOfWorkersFile = new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
              + federateName + "/NumberOfWorkers.txt");
          FileWriter fw = new FileWriter(NumberOfWorkersFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(numberOfWorkers);
          bw.close();
        }
      }

      if (theAttributes.containsKey(MarketDemand)) {
        stringDecoder.decode(theAttributes.get(MarketDemand));
        marketDemand = stringDecoder.getValue();

        File MarketDemandFile = new File(
            "C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"    +    federateName    +
"/MarketDemand.txt");
        FileWriter fw = new FileWriter(MarketDemandFile, false);
        BufferedWriter bw = new BufferedWriter(fw);

        bw.write(marketDemand);
        bw.close();

      }

    } catch (DecoderException e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    } catch (FileNotFoundException e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    } catch (IOException e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    }
  }

  @Override
  public final void provideAttributeValueUpdate(ObjectInstanceHandle theObject,
AttributeHandleSet theAttributes,
```

```
      byte[] userSuppliedTag) {

  }

  @Override
  public void receiveInteraction(InteractionClassHandle interactionClass,
ParameterHandleValueMap theParameters,
      byte[] userSuppliedTag, OrderType sentOrdering, TransportationTypeHandle theTransport,
      SupplementalReceiveInfo receiveInfo) throws FederateInternalError {

    try {

      final HLAunicodeString stringDecoder = _encoderFactory.createHLAunicodeString();

      if (theParameters.containsKey(ScName)) {
        stringDecoder.decode(theParameters.get(ScName));
        String scenario = stringDecoder.getValue();
        if (scenario.equals(federateName))
          loadScenario();
      }

      else if (theParameters.containsKey(Action)) {
        stringDecoder.decode(theParameters.get(Action));
        String action = stringDecoder.getValue();
        if (action.equals("START")) {
          enableTimeManagement();
          startScenario();
        }
        if (action.equals("PAUSE"))
          pauseScenario();
        disableTimeManagement();
        if (action.equals("STOP"))
          stopScenario();
      }

      else if (theParameters.containsKey(RealTimeFactor)) {

        stringDecoder.decode(theParameters.get(RealTimeFactor));
        gui.setRealTimeFactor(Double.valueOf(stringDecoder.getValue()));
      }

      else if (theParameters.containsKey(SetupTimeReduction)) {

        if (federateName.equals("SMED")) {
          stringDecoder.decode(theParameters.get(SetupTimeReduction));

          File SetupTimeReductionFile = new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
              + federateName + "/SetupTimeReduction.txt");
          FileWriter fw = new FileWriter(SetupTimeReductionFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(stringDecoder.getValue());
          bw.close();
        }
      }

      else if (theParameters.containsKey(ProcessingAndDefectsReduction)) {
        if (federateName.equals("5S")) // make sure about FiveS
        {
          stringDecoder.decode(theParameters.get(ProcessingAndDefectsReduction));
```

```
          File ProcessingAndDefectsReductionFile = new File(
              "C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/" + federateName
                  + "/ProcessingAndDefectsReduction.txt");
          FileWriter fw = new FileWriter(ProcessingAndDefectsReductionFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(stringDecoder.getValue());
          bw.close();
        }
      }

      else if (theParameters.containsKey(DefectsReduction)) {

        if (federateName.equals("POKAYOKE")) {
          stringDecoder.decode(theParameters.get(DefectsReduction));

          File DefectsReductionFile = new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
              + federateName + "/DefectsReduction.txt");
          FileWriter fw = new FileWriter(DefectsReductionFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(stringDecoder.getValue());
          bw.close();
        }
      }

      else if (theParameters.containsKey(TravelTimeReduction)) {

        if (federateName.equals("UCELL"))
        {
          stringDecoder.decode(theParameters.get(TravelTimeReduction));

          File                  TravelTimeReductionFile                  =                  new
File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-May19/"
              + federateName + "/TravelTimeReduction.txt");
          FileWriter fw = new FileWriter(TravelTimeReductionFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(stringDecoder.getValue());
          bw.close();
        }
      }

      else if (theParameters.containsKey(PULLValues)) {

        if (federateName.equals("PULL")) // make sure about FiveS
        {
          stringDecoder.decode(theParameters.get(PULLValues));

          File  PULLValuesFile  =  new  File("C:/Users/academic1/Desktop/Lean_Simulation_Aero-
May19/"
              + federateName + "/PULLValues.txt");
          FileWriter fw = new FileWriter(PULLValuesFile, false);
          BufferedWriter bw = new BufferedWriter(fw);

          bw.write(stringDecoder.getValue());
          bw.close();
        }
      }
```

```
      else

    } catch (FederateNotExecutionMember | NotConnected | InteractionClassNotPublished
        |  InteractionParameterNotDefined  |  InteractionClassNotDefined  |  SaveInProgress  |
RestoreInProgress
        | RTIinternalError e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    } catch (Exception e) {
      // TODO Auto-generated catch block
      e.printStackTrace();
    }

  }

  void enableTimeManagement() throws Exception {

    /////////////////////
    // Time Management //
    /////////////////////

    _logicalTimeFactory = (HLAfloat64TimeFactory) LogicalTimeFactoryFactory
        .getLogicalTimeFactory(HLAfloat64TimeFactory.NAME);
    _logicalTime = _logicalTimeFactory.makeInitial();

    _lookahead = _logicalTimeFactory.makeInterval(5);
    _rtiAmbassador.enableTimeRegulation(_lookahead);
    _rtiAmbassador.enableTimeConstrained();
  }

  void disableTimeManagement() throws Exception {

    _rtiAmbassador.disableTimeRegulation();
    trEnabled = false;
    _rtiAmbassador.disableTimeConstrained();
    tcEnabled = false;
  }

  @Override
  public void timeRegulationEnabled(LogicalTime time) throws FederateInternalError {
    // TODO Auto-generated method stub

    trEnabled = true;
  }

  @Override
  public void timeConstrainedEnabled(LogicalTime time) throws FederateInternalError {
    // TODO Auto-generated method stub

    tcEnabled = true;
  }

  void updateAttributes(DisplayEntity entity) throws Exception {
    AttributeHandleValueMap attributeValues =
_rtiAmbassador.getAttributeHandleValueMapFactory().create(1);
    HLAunicodeString scNameEncoder = _encoderFactory.createHLAunicodeString(federateName);
    //

    float scSimTime = (float) Simulation.getInstance().getSimTime() / 3600;
    HLAfloat32LE scSimTimeEncoder = _encoderFactory.createHLAfloat32LE(scSimTime);
```

```
    HLAinteger32LE scWIPEncoder = _encoderFactory.createHLAinteger32LE(scWIP);
    HLAinteger32LE scProductionThroughputEncoder =
_encoderFactory.createHLAinteger32LE(scProductionThroughput);
    HLAfloat32LE scDefectRateEncoder = _encoderFactory.createHLAfloat32LE(scDefectRate);
    HLAfloat32LE scLeadTimeEncoder = _encoderFactory.createHLAfloat32LE(scLeadTime);

    attributeValues.put(Name, scNameEncoder.toByteArray());
    attributeValues.put(SimTime, scSimTimeEncoder.toByteArray());
    attributeValues.put(WIP, scWIPEncoder.toByteArray());
    attributeValues.put(ProductionThroughput, scProductionThroughputEncoder.toByteArray());
    attributeValues.put(DefectRate, scDefectRateEncoder.toByteArray());
    attributeValues.put(LeadTime, scLeadTimeEncoder.toByteArray());

    //
    _rtiAmbassador.updateAttributeValues(regObjInstName, attributeValues, null);
  }

  void disconnect() throws Exception {
    _rtiAmbassador.resignFederationExecution(ResignAction.DELETE_OBJECTS_THEN_DIVEST);
    _rtiAmbassador.destroyFederationExecution(FEDERATION_NAME);
    _rtiAmbassador.disconnect();
    _rtiAmbassador = null;
  }

  void loadScenario() throws Exception {
    gui = GUIFrame.create(guiArgs);

    ParameterHandleValueMap                     scenarioLoadedParameters                     =
_rtiAmbassador.getParameterHandleValueMapFactory().create(1);
    HLAunicodeString scenarioLoadedEncoder = _encoderFactory.createHLAunicodeString();
    scenarioLoadedEncoder.setValue(federateName);
    scenarioLoadedParameters.put(FederateNameLoaded, scenarioLoadedEncoder.toByteArray());
    //
    boolean scLoaded = true;
    _rtiAmbassador.sendInteraction(scLoaded ? ScenarioLoaded : ScenarioError,
scenarioLoadedParameters, null);
  }

  void startScenario() {

    gui.startSimulation();

  }

  void pauseScenario() {
    gui.pauseSimulation();
  }

  void stopScenario() {
    gui.stopSimulation();
    gui.close();
  }

  private boolean pause = false;

  public synchronized void onAddEntity(DisplayEntity entity) {
    try {
      double simTime = Simulation.getInstance().getSimTime();

      updateAttributes(entity);
```

```
      while (!trEnabled || !tcEnabled) {
        Thread.sleep(100);
      }
      //
      try {

        double dtWIP = entity.getOutputHandle("dtWIP").getValue(simTime, Double.class);
        scWIP = (int) dtWIP;

        double productionThroughput =
entity.getOutputHandle("productionThroughput").getValue(simTime, Double.class);
        scProductionThroughput = (int) productionThroughput;

        double defectRate = entity.getOutputHandle("defectRate").getValue(simTime,
Double.class);
        scDefectRate = (float) defectRate;

        double dtLeadTime = entity.getOutputHandle("dtLeadTime").getValue(simTime,
Double.class);
        scLeadTime = (float) dtLeadTime;

        double month = entity.getOutputHandle("month").getValue(simTime, Double.class);
        //
      } catch (Exception e) {
        e.printStackTrace();
      }
      //
      pause = true;
      _logicalTime = _logicalTimeFactory.makeTime(simTime / 3600);

      try {
        _rtiAmbassador.nextMessageRequest(_logicalTime);
        while (pause)
          wait();

      } catch (LogicalTimeAlreadyPassed | InvalidLogicalTime | InTimeAdvancingState
          | RequestForTimeRegulationPending | RequestForTimeConstrainedPending | SaveInProgress
          | RestoreInProgress | FederateNotExecutionMember | NotConnected | RTIinternalError e)
{
        e.printStackTrace();

      }

    } catch (Exception e) {
      throw new RuntimeException(e);
    }
  }

  @Override
  public synchronized void timeAdvanceGrant(LogicalTime theTime) throws FederateInternalError {
    pause = false;
    notifyAll();
      }

  public static void main(String[] args) throws Exception {
    String federateName = args[0].substring(4);
    instance = new Federate(federateName);
    guiArgs = args;
  }
}
```