CRANFIELD UNIVERSITY

Xu Jiang

Integrating Enterprise Resource Planning (ERP) Systems to Symbiotic Simulation

School of Applied Sciences

MSc by Research Thesis

Academic Year: 2013 -2014

Supervisor: Dr Benny Tjahjono and Dr Peter Ball

September 2014

CRANFIELD UNIVERSITY

School of Applied Sciences

MSc by Research Thesis

Academic Year 2013 -2014

Xu Jiang

Integrating Enterprise Resource Planning (ERP) Systems to Symbiotic Simulation

Supervisors: Dr Benny Tjahjono and Dr Peter Ball
September 2014

This thesis is submitted in partial fulfilment of the requirements for the degree of Master of Science

ABSTRACT

Symbiotic simulation is inspired by symbiosis in biology and is defined as a close association between a simulation system and a physical system, which is usually beneficial to at least one of them. Many applications have been proposed in recent years to implement symbiotic simulation to physical systems. However, little research has been carried out on implementing computer-based information systems to symbiotic simulation.

In this research, symbiotic simulation is developed with the integration of Enterprise Resource Planning (ERP) systems. On the one hand, simulation models benefit from the relevant manufacturing data which are provided by ERP systems. On the other hand, ERP systems and physical systems benefit from the feedback offered by simulation models.

A tube manufacturing shop floor has been selected as a case to demonstrate how the symbiotic simulation can be practically implemented. In this case example, a simulation model has been built using Anylogic 6 to mutually interact with a SAP R/3 system. Experimentation has also been carried out to evaluate the extent to which the symbiotic simulation can effectively address uncertainties in manufacturing environments and ultimately control the ERP system and the tube manufacturing shop floor.

Keywords: Symbiotic simulation, Enterprise resource planning, simulation, integration, optimisation, manufacturing uncertainty.

ACKNOWLEDGEMENTS

Firstly, I would like to thank Commercial Aircraft Corporation of China (COMAC) and China Scholarship Council for sponsoring and giving me the precious opportunity to attend this well-structured research programme.

In particular, I would like to express my gratitude to my supervisors, Dr Benny Tjahjono and Dr Peter Ball. Their knowledge and guidance throughout the whole research period has been extremely valuable. Without their continuous supervision and encouragement, I would not have been able to finish the research programme successfully.

I would like to thank one and all who supported and helped me during my research programme. I appreciate Prof Mark Tibbett and Dr Fan Ip-Shing for their valuable advice and help in each research phase. In addition, I would also like to thank all the staff and research students in SAS who helped me a lot during this academic year.

I also extend my sincere gratitude towards my family and friends in China, especially to my better half, Miss Wang for her unconditional understanding and love during the whole research programme.

Last but not least, I extend my deepest thanks to all my COMAC mates here who have made this one year so special and unforgettable in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ERP	Enterprise Resource Planning
MRP	Material Requirements Planning
SSCS	Symbiotic Simulation Control System
SSDSS	Symbiotic Simulation Decision Support System
SSFS	Symbiotic Simulation Forecasting System
SSMVS	Symbiotic Simulation Model Validation System
SSADS	Symbiotic Simulation Anomaly Detection System
DDDAS	Dynamic Data Driven Applications Systems
OT	Operator Trigger
AT	Anomaly Trigger
PT	Period trigger
DCO	Data Collecting Object
OEL	Object Linking and Embedding
DFO	Data Fusion Object
MMO	Model Management Object
OptO	Optimisation Object
CRES	Corrosion Resistant Steel
Ti	Titanium
Al	Aluminium
COTS	Commercial off-the-shelf

1 Introduction

The concept of symbiotic simulation systems was motivated by symbiosis in biology. The biological definition of symbiosis is broader and contains several subcategories, such as mutualism and parasitism. However, for simulation related work, subcategories other than mutualism are often ignored. Thus for symbiotic simulation systems, mutualism is considered as the only form of symbiosis (Aydt et al., 2008b). Symbiotic simulation systems indicate mutually beneficial interactions between a simulation model and a physical system (Fujimoto et al., 2002). Meanwhile, ERP systems have already become fundamental tools in enterprise management and have proved to be valuable in many application domains. This research explores the possibility of integrating ERP systems to symbiotic simulation systems. This chapter provides an overview about this research, including the research background, motivation, aim, objectives and the thesis structure.

1.1 Background and Motivation

The concept of symbiotic simulation was first proposed at the Dagstuhl Seminar on Grand Challenges for Modelling and Simulation in 2002. Typically, a symbiotic simulation system is comprised of a simulation model and a physical system. On the one hand, the simulation model continuously acquires real-time or near real-time data from the physical system using real-time sensors. On the other hand, the physical system benefits from the simulation results (Fujimoto et al., 2002). Many applications of symbiotic simulation have been proposed to physical systems in recent years, for example, semiconductor manufacturing, unmanned aerial vehicles, and radiation detection (Aydt et al., 2008a; Aydt et al., 2009b). However, in these symbiotic simulation systems, little work has been revealed about integrating information systems, such as ERP systems, to symbiotic simulation systems.

ERP systems are information systems applied to manage and integrate operation activities of an enterprise including almost all aspects, such as human resource, manufacturing, project management, and financial accounting

(Davenport, 1998). In the manufacturing environment, most operational data are stored in ERP systems, such as circle times, bill of materials and so on. A survey carried out by Robertson and Perera (2002) showed that the majority of sources of data for simulation tools are stored in ERP systems, especially for manufacturing companies (Skoogh et al., 2012b). In addition, as ERP systems are developed from Material Requirements Planning (MRP) systems, they inevitably inherit some drawbacks from MRP systems (Moon and Phatak, 2005). They lack the function of prognostic and the ability of dealing with uncertainties, which are crucial factors to support making business and manufacturing decisions (Addo-Tenkorang and Helo, 2011; Battista et al., 2011). The various benefits of using simulation tools in coping with realistic problems have been identified, such as supporting decision making, time compression and expansion, exploring possibilities, diagnosing problems, identifying constraints, developing understandings, visualisation, preparing for changes, and specifying requirements (Banks, 1999; Babulak and Wang, 2008; Jovanoski et al., 2013).

Some work has already been carried out with regard to integrate ERP systems and simulation tools. For instance, a discrete event simulation has been proposed to enhance the functionality of ERP systems in obtaining more accurate lead time (Moon and Phatak, 2005). However, almost all previous applications were used to accomplish some specific purposes and have some limitations. They used traditional simulation methods to link ERP systems and simulation tools instead of symbiotic simulation. Meanwhile the mutual beneficial interactions of ERP systems and simulation tools are extremely suitable to integrate ERP systems to symbiotic simulation systems. Motivated by it, this research explores the probability of integrating ERP systems to symbiotic simulation systems.

1.2 Aim and Objectives

The research aims to investigate the extent to which EPR systems can be integrated to symbiotic simulation systems. This kind of symbiotic simulation systems can be used to deal with uncertainties and random effects in the

manufacturing environment. On the one hand, simulation models benefit from the relevant manufacturing data which are provided by ERP systems and physical systems. On the other hand, simulation models generate optimised feedback to improve the performance of ERP systems and physical systems.

In order to achieve the aim, this research focuses on the following objectives:

- Developing a generic framework for integrating ERP systems to symbiotic simulation systems
- Demonstrating how the framework can be practically implemented
- Validating the functionalities of the symbiotic simulation system

1.3 Thesis Structure

Chapter 1 - Introduction: Research background and motivation are briefly illustrated in this section. This section also suggests the aim and objectives of this research.

Chapter 2 - Literature Review: Related work and state-of-the-art technologies involving symbiotic simulation, ERP systems, and discrete event simulation are presented. In addition, the research gap is identified.

Chapter 3 - Methodology: This section shows the methodology developed for the research.

Chapter 4 - A generic framework for integrating ERP systems to symbiotic simulation systems including the mathematical concept, different objects and subsystems, function blocks and the workflow are illustrated in detail.

Chapter 5 - Case example development: this section provides an overview of the selected ERP-based tube manufacturing floor shop, the simulation tool and the ERP system. Moreover, the process of building the symbiotic simulation system is presented in detail.

Chapter 6 - Experimentation: In order to validate the functions of the symbiotic simulation system, the results and analysis of the experimentation are illustrated.

Chapter 7 - Discussion and conclusions: The overall results and outcomes of the research are illustrated and discussed. In addition, the research contributions, limitations and future work are discussed and prescribed.

2 Literature Review

A comprehensive literature review has been carried out regarding ERP systems and symbiotic simulation. Firstly, symbiotic simulation is reviewed to understand the state-of-the-art work and applications of symbiotic simulation. Secondly, ERP systems and their data management and interactions with simulation tools are then reviewed to illustrate the benefits of integrating ERP systems to symbiotic simulation systems. Thirdly, other simulation methods are reviewed and compared with symbiotic simulation to illustrate the advantages of integrating ERP systems to symbiotic simulation systems. Finally, the research gap is identified and illustrated.

2.1 Symbiotic Simulation

Symbiosis, which has its origins in biology, is a close interaction between two or more different species. The concept of symbiotic simulation systems was motivated by symbiosis and first proposed at the Dagstuhl Seminar on Grand Challenges for Modelling and Simulation in 2002. The paradigm focuses on a close relationship between a simulation model and a physical system which are mutually beneficial to each other (Aydt et al., 2008b). Typically, the simulation model acquires real-time or near real-time data from the physical system using sensors. By running some pre-defined "what-if" scenarios, simulation systems can predict, influence, optimise or control the performance of physical systems. Symbiotic simulation systems continuously execute simulation models and interact with physical systems in real-time (Fujimoto et al., 2002).

Symbiotic simulation systems have many potential application areas such as transportation, communication networks, and manufacturing. For the manufacturing environment, a quick what-if scenario is required to react to unexpected changes in a shop floor. Relevant manufacturing system data are constantly acquired and updated to an online simulation, thus simulation models can continuously improve the performance of the physical system and ultimately improve the competitiveness of the company.

2.1.1 Different Classes of Symbiotic Simulation Systems

Aydt (2008) extended the definition of symbiotic simulation to cover a wider aspect and designated five different types of symbiotic simulation systems: symbiotic simulation control system (SSCS), symbiotic simulation decision support system (SSDSS), symbiotic simulation forecasting system (SSFS), symbiotic simulation model validation system (SSMVS), and symbiotic simulation anomaly detection system (SSADS). Each type of symbiotic simulation system can be implemented separately. Meanwhile, various symbiotic simulation systems can be integrated to model some complicated cases, such as in the semiconductor manufacturing factory. The integration of different symbiotic simulation systems is further referred to as the hybrid symbiotic simulation system (Aydt et al., 2008b). Table 2-1 shows various purposes, loop types, what-if scenarios, and symbiosis types of the five symbiotic simulation systems.

Table 2-1 Different classes of symbiotic simulation systems (Aydt et al., 2009b)

Class	Purpose	Loop	What-if Scenarios	Symbiosis Type
SSCS	Control of a physical system	Closed	Control alternatives	Mutualism/Parasitism
SSDSS	Support of an external decision maker	Closed	Decision alternatives	Mutualism/Parasitism
SSFS	Forecasting of a physical system	Open	Different assumptions for environmental conditions	Commensalism
SSMVS	Validation of a simulation system	Open	Alternative models or different parameters	Commensalism
SSADS	Detection of anomalies either in the physical system or in the simulation model	Open	Reference model only	Commensalism

Closed loop means that a control feedback can be generated to influence or control the physical system. SSCS and SSDSS are both closed loop symbiotic

simulation systems. After measuring data, running what-if scenarios, and analysing results, an SSCS uses internal actuators to control the physical system directly based on simulated results. For instance, it can activate actuation mechanisms to control manufacturing directly. Compared with SSCS, an SSDSS influences physical systems indirectly depending on some external decision makers, such as manufacturing managers. For instance, an SSDSS can suggest how many machines to purchase as an investment as the reference for a manufacturing manager (Aydt et al., 2008b). Figure 2-1 and Figure 2-2 show the overview of an SSCS and an SSDSS respectively.

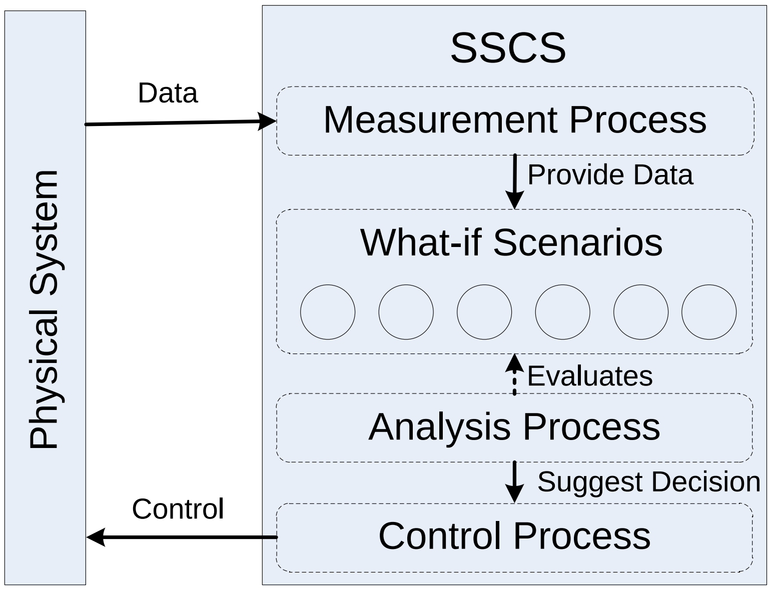


Figure 2-1 Overview of an SSCS (Aydt et al., 2008b)

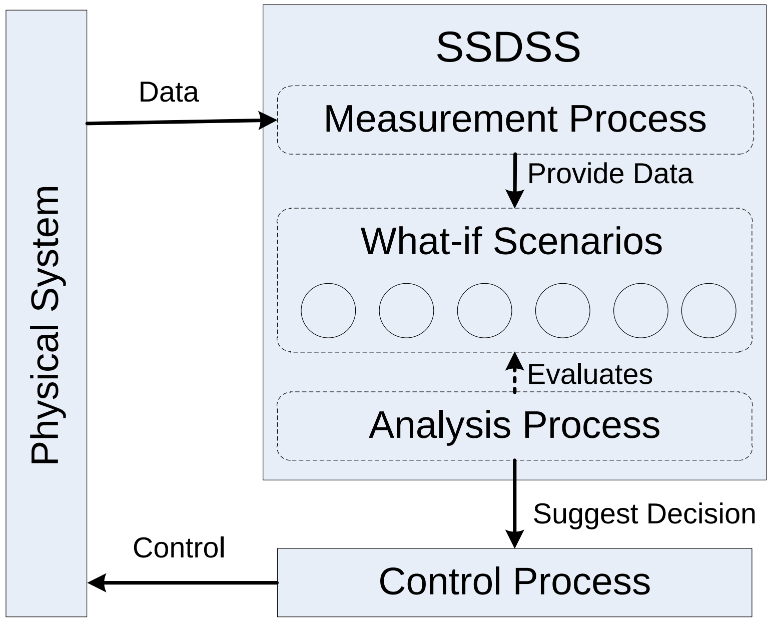


Figure 2-2 Overview of an SSDSS (Aydt et al., 2008b)

Open loop means that no feedback can be generated to physical systems. The other three symbiotic simulation systems are all open-loop systems and do not have the ability to control physical systems. An SSFS is typically applied to predict future behaviours of physical systems. For instance, it is normally used to weather forecast. An SSMVS is normally used for the model validation purpose and aims to determine a model which describes the current behaviour of the physical system with the most accuracy. An SSADS simulates the behaviour of physical systems continuously. It is generally used to detect anomalies by comparing simulation results or physical systems with a reference simulation. An anomaly can be detected if the discrepancy is beyond a certain threshold. In this way, the possible source of the anomaly can be determined (Aydt et al., 2008b). Figure 2-3, Figure 2-4, and Figure 2-5 show the overview of an SSFS, an SSMVS and an SSADS respectively.

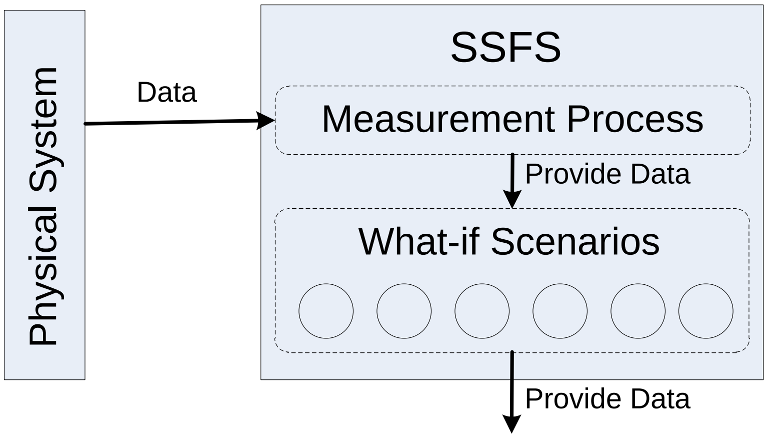


Figure 2-3 Overview of an SSFS (Aydt et al., 2008b)

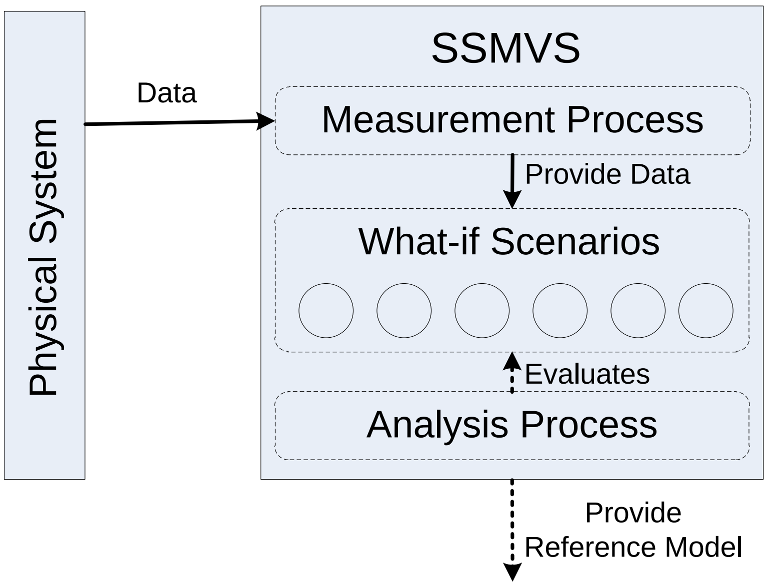


Figure 2-4 Overview of an SSMVS (Aydt et al., 2008b)

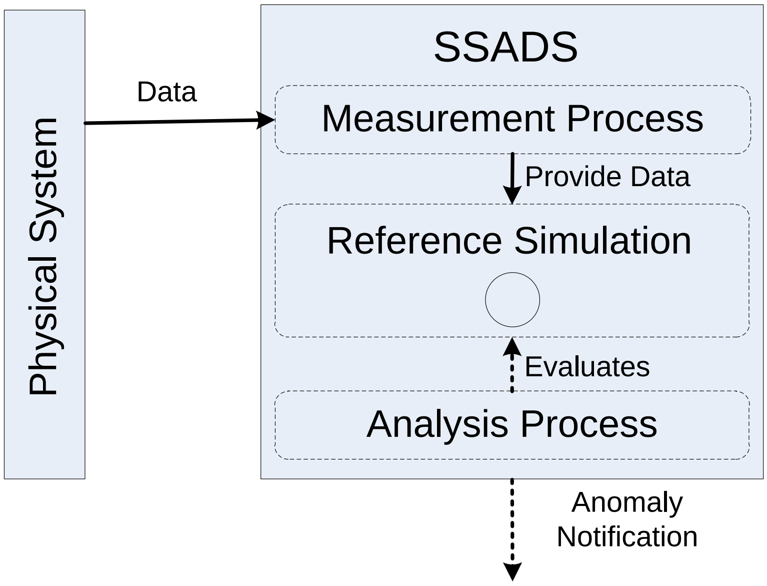


Figure 2-5 Overview of an SSADS (Aydt et al., 2008b)

2.1.2 Various Applications of the Symbiotic Simulation System

Applications of symbiotic simulation vary from industry to industry with different purposes. Symbiotic simulation systems automatically evaluate what-if scenarios and suggest solutions to practical problems (Aydt et al., 2009b).

With the purpose of improving the manufacturing process of semiconductor manufacturing, a proof-of-concept symbiotic simulation system was developed to monitor, optimise and control the assembly and test operation of semiconductor backend. Experiments showed that the symbiotic simulation system has functions to effectively accomplish the monitoring, optimisation and controlling tasks. For a shorter simulation time, the symbiotic simulation system can respond rapidly to abrupt changes in the physical system (Low et al., 2005). Another application in semiconductor manufacturing was carried out by Aydt (2008). The proposed symbiotic control system uses reactive what-if analysis to obtain a stable configuration of a wet bench tool set in near real-time. This symbiotic simulation system was validated to have an outstanding performance for finding the stable configuration. It proves that symbiotic simulation can be applied and integrated in an automation manufacturing environment (Aydt et al., 2008a). After three years, Aydt projected a symbiotic simulation-based problem

solver to automatically resolve decision making problems regarding the operations of the various tools in an entire semiconductor manufacturing fab. The problem solver agent detects the physical system and executes what-if scenarios to identify and solve some manufacturing problems (Aydt et al., 2011).

In order to overcome the effects of new information or sensor observations of unmanned aerial vehicles (UAVs), a symbiotic simulation system was applied in the process of path planning to deal with these uncertainties (Kamrani and Ayani, 2007). Another decision making and controlling application of symbiotic simulation was carried out in 2009. An SSCS was developed to improve the performance of inventory management in the lubricant industry (Fanchao et al., 2009).

2.1.3 Generic Framework of Symbiotic Simulation

There are various symbiotic simulation applications as symbiotic simulation has the potential to extend to a larger scale. All these applications have some common principles and processes, for example, what-if analyses. It will be quite easy for developers to implement symbiotic simulation to their own domains if a generic symbiotic simulation framework is developed. Therefore, in 2009, an agent-based generic symbiotic simulation framework was developed with the consideration of applicability, extensibility, and scalability (Aydt et al., 2009a).

The architecture of the framework is comprised of three layers: a perception layer, a process layer, and an actuation layer. Different layers contain various capabilities with specific functions. The generic framework uses agents to present a specific function. For instance, sensors have the ability to acquire data from a physical system. The development of the generic framework focused on integrating simulation tools and physical systems using symbiotic simulation technologies.

2.1.4 Use Discrete Event Simulation to Build Symbiotic Simulation Models in Manufacturing Environments

Discrete event simulation is defined as imitation of a real-world process over time and applied to search out solutions of real-world problems (Banks, 1999). Among various applications of discrete event simulation, one major application area is the manufacturing system, with the first usage dating back to at least the early 1960's. Manufacturing issues where discrete event simulation is used can be classified in three areas (Law and McComas, 1998):

- It can be used to identify the requirements of equipment or personnel, such as number and type of machines, location and size of workshops, number of shifts and so on.
- It can be used to evaluate the performance of manufacturing issues, such as time-in-system analysis, bottleneck analysis and so on.
- It can be used to evaluate the operational procedures, such as production scheduling, inventory policies, quality-control policies and so on.

Discrete event simulation technology has been widely used to manage various industrial problems because it has the ability to simulate the dynamics of a real system on an event-by-event basis. Meanwhile, its analysis systems can investigate complicated issues by using the computational and mathematical techniques (Babulak and Wang, 2008). Software suppliers have developed many simulation packages, such as Arena, Anylogic and Witness to deal with manufacturing issues. Based on different functions, different tools can be used to achieve specific purposes (Abu-Taieh and El-Sheikh, 2007).

2.2 ERP Systems and Their Interactions with Simulation Tools in Manufacturing Environments

ERP systems were initially developed in the late 1980s with the power of enterprise-wide inter-functional coordination and integration. After that, more modules and functions were developed along with the advancement of computer hardware and software systems (Rashid et al., 2002). ERP systems are information systems that can be applied to manage and integrate the

business activities of an enterprise including almost all aspects, such as human resource, manufacturing, project management and financial accounting (Davenport, 1998). The primary ERP system vendors are referred to as BOPSE, which are BAAN, Oracle, PeopleSoft, SAP and J.D. Edwards. Among them, SAP holds the largest market share (O'Leary, 2000).

ERP systems have already become the dominant management software in manufacturing and distribution systems in today's competitive business environment (Ho and Ireland, 2012). However, they lack the function of prognostic and the ability of dealing with uncertainties, which are crucial factors to support making business and manufacturing decisions (Addo-Tenkorang and Helo, 2011; Battista et al., 2011). However, simulation tools have the ability to cope with these realistic problems, such as supports with decision making, time compression and expansion, exploring possibilities, diagnosing problems, identifying constraints, developing understanding, visualisation, preparing for changes, and specifying requirements (Banks, 1999; Babulak and Wang, 2008; Jovanoski et al., 2013). Recent work has been carried out to integrate ERP systems and simulation tools in the manufacturing environment. Efforts have been made mainly in the following two areas:

2.2.1 Data Management between ERP Systems and Simulation Tools

Input data management is a crucial and time-consuming process for both ERP systems and simulation tools (Skoogh et al., 2012a). At the heart of an ERP system is the central database that feeds data into a series of applications supporting diverse company functions (Davenport, 1998). Therefore, lots of work has been reviewed focusing on the linkage and data transmission between ERP systems and simulation tools.

ERP systems normally host most of the operational data, such as circle times, set-up times, and bill-of-materials. As ERP systems contain the information required by simulation models, ERP systems are assumed as prime sources for simulation data. It is also suggested that better integration between simulation tools and ERP systems is needed in order to facilitate automation of data transmission between them (Robertson and Perera, 2002). A survey was

handed out to investigate input data of simulation tools in 2012. For manufacturing companies, 40% of the main and 77% of the common sources of data for simulation tools are stored in ERP systems. It also revealed that around 80% of users still rely on highly manual work procedures in input data management. Additionally, results also showed that 77% of participants expect to implement a higher level of automation of input data for simulation tools (Skoogh et al., 2012b).

Robertson and Perera (2002) proposed a concept of automated data transmission between ERP systems and simulation tools. In addition, an illustrated framework has been proposed regarding how the intermediary database was used to automatically extract and then store the data between ERP systems and simulation tools as shown in Figure 2-6 (Robertson and Perera, 2002). Since then, the need for an interface or specific tools for automatically acquiring data from ERP systems for simulation tools has been raised and many methods and products have been developed through the years. For instance, a concept design of the software tool called the Generic Data Management Tool has been proposed to automatically collect critical and time-consuming data from ERP systems (Skoogh et al., 2012a).

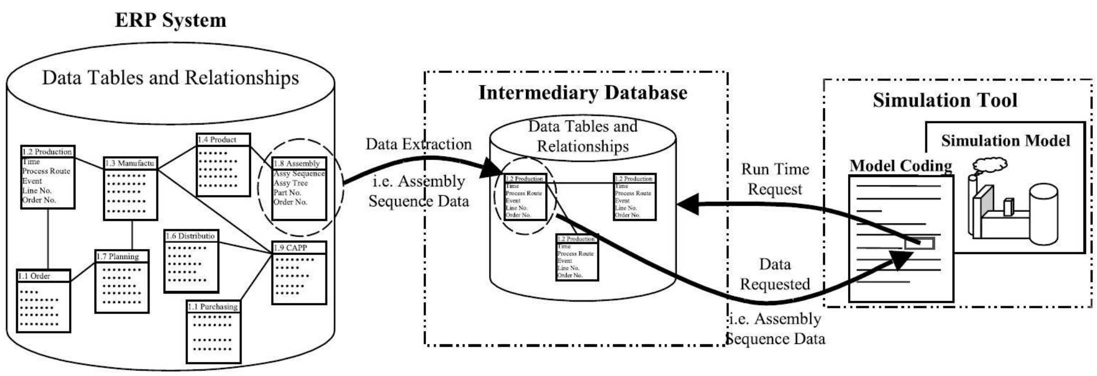


Figure 2-6 A concept of automated data transmission between ERP systems and simulation tools (Robertson and Perera, 2002)

2.2.2 Interactions between ERP Systems and Simulation Tools

As ERP systems are developed from MRP systems, they inevitably inherit some drawbacks from their origins. Moon and Phatak (2005) summarised the

manufacturing drawbacks of ERP systems as fixed lead-time, infinite resource, fixed routing. Influenced by these shortcomings, this may eventually result in missed deadlines for the manufacturing industry. A method was proposed to connect a discrete simulation model and an ERP system to enhance the functionality of the ERP system with determining realistic production lead time data. Arena 7.0 and SAP R/3 system were used to build the ERP/simulation system. A Germany pump manufacturing factory, as an illustrated case, was simulated and the lead-time data can be determined with more accuracy (Moon and Phatak, 2005). Figure 2-7 shows the proposed ERP/simulation system. After manually triggering this ERP/Simulation system, the simulation tool acquires relevant manufacturing data from the SAP R/3 system and output simulated lead time for a manufacturing manager. Comparing the simulated results with the actual due date, the manufacturing manager changes data by adjusting overtime and executes the simulation model again. This process is repeated until the manufacturing manager is satisfied with the simulation results. The ERP/simulation system proposed by Moon and Phatak can be seen as an uncompleted symbiotic simulation system. However, it uses the traditional off-line simulation methods and lacks the ability of automatic validation. The way this system optimises and influences the physical system depends on the production manager instead of being directly controlled. .

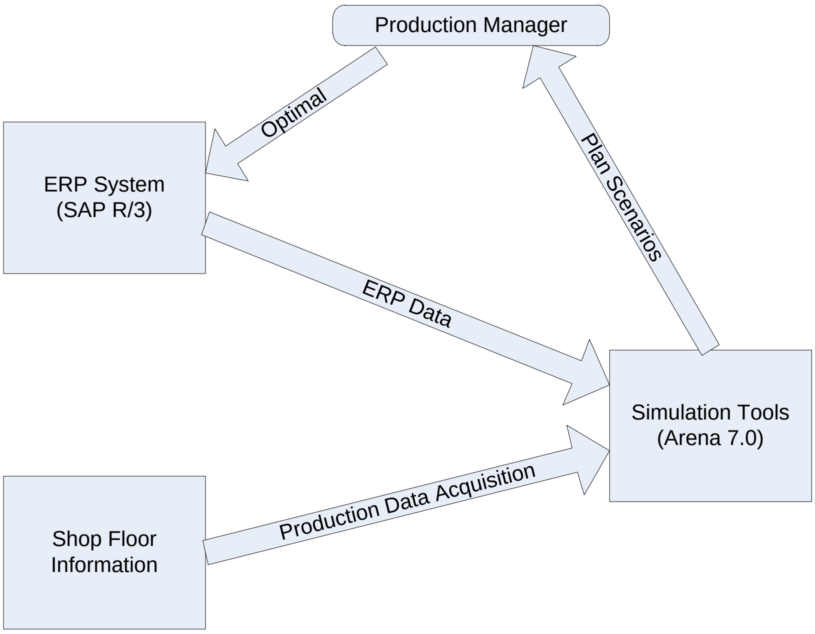


Figure 2-7 The ERP/simulation system (Moon and Phatak, 2005)

In 2007, a simulation model was developed to capture the effects of uncertainty on an EPR-controlled manufacturing supply chain. MRP release logic and an ERP-controlled manufacturing supply chain were simulated by ARENA/SIMAN. The results showed that late deliveries from suppliers, machine breakdown, unexpected/urgent changes to machine assignments, and customer design changes significantly affected the performance of ERP in many ways (Koh and Gunasekaran, 2007).

In 2011, MRP procedure was integrated into a simulation tool which was conceived to natively embed look-back material management logic. A solution was found to keep MRP algorithm execution and share input data with the simulation tool simultaneously. Simulation tools are used to predict the future state of the MRP system (Battista et al., 2011).

An ERP-controlled manufacturing system was simulated to assess the influence of forecasting errors on the performance of ERP systems in 2012. The simulation results suggested that environmental factors, such as lot-sizing rule, demand lumpiness, cost ratio, and lead time uncertainty, influence the performance of ERP systems. In this case, a validation simulation was also carried out to evaluate the effectiveness of using lot-sizing rules to cope with

forecast errors under these uncertainties in an ERP system (Ho and Ireland, 2012).

2.3 Other Simulation Concepts Related to Symbiotic Simulation

Among the wide variety of simulation methods, on-line simulation and the dynamic data driven applications system (DDDAS) are two simulation concepts which are closely related to symbiotic simulation systems.

2.3.1 On-line Simulation

This concept has various definitions and can be found in many areas. For instance, Manivannan and Banks (1992) defined it as a computerised system capable of performing both deterministic and stochastic simulations in real time (or near real time) to monitor, control, and schedule parts and resources in a discrete-part manufacturing environment (Manivannan and Banks, 1992). Another slightly different definition, which was given by Kamrani (2007), is a simulation that runs in real-time and in parallel with a physical system and does not necessarily include a feedback to the physical system (Kamrani and Ayani, 2007). The term on-line simulation has been used in various ways, usually referring to a simulation which is initialised and driven by real-time sensor data (Aydt et al., 2008b). However, on-line simulation does not reflect the close relationship between simulation models and physical systems as symbiotic simulation does.

2.3.2 DDDAS

DDDAS is a paradigm whereby applications (or simulations) and measurements become a symbiotic feedback control system. It entails the ability to dynamically incorporate additional data into an executing application (or a simulation), and in reverse, the ability of an application (or a simulation) to dynamically steer the measurement process. Different from symbiotic simulation, DDDAS emphasises the ability of the application (or simulation) to control and guide the measurement processes. Symbiotic simulation, on the other hand, focuses on the control and influence of physical systems. Although very similar, the primary focus of these paradigms is different (Aydt et al., 2008b).

Online simulation and DDDAS are simulation concepts related to symbiotic simulation. Online simulation focuses on acquiring real-time data while the latter one emphasises the ability of the application (or simulation) to control and guide the measurement processes. None of them reveal the interactions and mutual beneficial relationship like symbiotic simulation does. A comparison of symbiotic simulation and its related paradigms (on-line simulation and DDDAS) has been illustrated in table 2-2.

Table 2-2 Comparison of symbiotic simulation, DDDAS, and on-line simulation
(Aydt et al., 2009a)

Paradigm	Steering of the Measurement Proc.	Control Feedback	Data-Driven App./Sim.	What-if Analysis
Symbiotic simulation	Optional	Optional	Simulation only	Yes
DDDAS	Mandatory	Optional	Both	Optional
On-line Simulation	Optional	No	Simulation only	Optional

2.4 Research Gap

A comprehensive literature review has been carried out relating to state-of-the art technologies and applications of symbiotic simulation. In addition, ERP systems and their data management and interactions with simulation tools have been reviewed as well.

Previous applications of symbiotic simulation mainly focused on physical systems. However, this close relationship addressed by symbiotic simulation is extremely suitable for integrating ERP systems to symbiotic simulation systems. In this way, the performance of existing applications can be improved and symbiotic simulation can be easily applied to ERP-based manufacturing shop floors. Therefore, an additional step of integrating ERP systems to symbiotic simulation systems is identified. This research gap, compared with how the previously mentioned works, is identified below:

- Little research has been done by implementing ERP systems to symbiotic simulation systems. In addition, there is a lack of generic framework for integrating ERP systems to symbiotic simulation systems

3 Methodology

The foundation of the research is to integrate ERP systems to symbiotic simulation systems. Therefore, the methodology focuses on development and validation of a generic framework for integrating ERP systems to symbiotic simulation systems. The research methodology to a large extent follows the three main stages.

3.1 Stage 1: Understanding the Interactions between ERP Systems and Symbiotic Simulation Systems and Designing the Function Blocks

The interactions between ERP systems and simulation tools were summarised based on the related works. After that, a generic framework for symbiotic simulation systems for ERP systems was developed following three steps. Firstly, a mathematical concept was designed as the guidance. Secondly, various objects and subsystems with specific capabilities were defined and designed. Finally, the function blocks and workflow of the generic framework were developed and illustrated.

3.2 Stage 2: Developing a Case Example of an ERP-based Symbiotic Simulation System

An ERP-based tube manufacturing workshop was chosen as a case example to build the symbiotic simulation system. Data were collected from this real tube manufacturing company in China. After acquiring all the information, data were summarised to make a concise tube manufacturing shop floor. The SAP R/3 system was chosen as the ERP system for its wide range usage in manufacturing industries. Anylogic 6 was chosen to build the discrete event simulation model. Based on the generic framework and collected data, a symbiotic simulation system was developed to reflect the ERP-based tube manufacturing shop floor.

3.3 Stage 3: Testing the Case Example and Analysing the Experimental Results

Two sets of experiments were designed and carried out to validate the functions of the symbiotic simulation system compared with traditional simulation methods. For raw materials experiments, they were carried out to validate that the symbiotic simulation system can detect anomalies and respond in real-time. For customer order experiments, they were carried out to validate that the symbiotic simulation system has the function of generating 2D (or 3D) animation of real-time state and future prediction. Additionally, the symbiotic simulation system can control the SAP R/3 system directly and suggest decision parameters for external actuators.

4 A Generic Framework for integrating ERP Systems to symbiotic simulation systems

The development of the generic framework is illustrated in detail in this chapter. Firstly, the mathematical concept is proposed as the mathematical guidance to cover common activities and functions. Secondly, different components and subsystems of the generic framework are defined. Finally, function blocks and workflow of the generic framework are outlined.

4.1 Mathematical Concept

The mathematical concept is used as the mathematical guidance to develop the generic framework. Based on the activities and functions of the ERP-based symbiotic simulation system, mathematical processes can be divided into the following three phases: simulation model development, experiment scenarios, optimisation and decisions.

4.1.1 Simulation Model Development

A symbiotic simulation system is based on a simulation model which interacts with ERP systems and physical systems. Firstly, an appropriate simulation model, further indicated as m, represents where an ERP-based physical system is needed. f_M represents the function of creating the simulation models for ERP systems and physical system. For simulation models, they contain a set of objects and parameters to represent the manufacturing environment. Objects, further denoted by obj, represent the basic components of simulation models and may represent very diverse things such as machines, products, ideas, organisations, vehicles and even people in different roles. Each object has its own associated capabilities, for example, a machine may have its specific use of drilling or a technician can repair a machine. Parameters, further denoted by p, represent different characteristics of objects or simulation models. For instance, a machine object may contain parameters such as circle time, setup time, capacity, and breakdown rate. Otherwise, parameters may only be used to store data such as simulated results. Thus various models can be built based on their objectives with different objects and parameters. The mathematical

expression of creating a simulation model for an ERP-based physical system by function M_{ERP} can be indicated as below:

$$m = f_M(obj_1, obj_2, \dots, obj_n;\ p_1,\ p_2, \dots, p_n); \tag{4-1}$$

By defining different objects and parameters, different model instances can be created. For instance, if objects represent machines and parameters represent capacities of each machine, this can be expressed by the following two expressions:

$$m_1 = f_M(obj_1, obj_2;\ 1,\ \ 2); \tag{4-2}$$

$$m_2 = f_M(obj_1,\ obj_2;\ 2,\ \ 2); \tag{4-3}$$

Equation 4-2 indicates that m_1 contains two types of machines and capacities of each type and these are 1 and 2 respectively. While equation 4-3 indicates the number of the first type, machine is 2 in m_2. By changing values of objects or parameters, countless different model instances can be created as a result. Among the infinite model instances, only one can be considered to represent the practical situations with the best accuracy. This instance is defined as a reference model, further denoted by m_R. The discrepancy between the simulation model and the practical situation is indicated as ∇. The requirement expression of the reference model is indicated as below:

$$\nabla m_2 < \nabla m_i \qquad \forall\ i\ \in\ (1,2, \dots, n), i\ \neq\ R; \tag{4-4}$$

Because uncertainties and complexities inevitably exist in a practical situation, sometimes the most accurate values of objects and parameters may change over time. Therefore, a reference model can only be valid for a period of time. After a period of time, there will be a model, which can represent the real world with higher accuracy, to replace the previous reference model (Aydt et al., 2009a). The mathematical expression of this process is indicated as below:

$$set\ \nabla m_j = min\{\nabla m_i\} \qquad \forall\ i\ \in\ (1,2, \dots, n); \tag{4-5}$$

$$if\ \nabla m_R > \nabla m_j\ \ then\ \ m_R = m_j;$$

4.1.2 Experiment Scenarios

Normally, experiment scenarios, further denoted by s, are designed for a specific objective to simulate the behaviour of physical systems (Aydt et al., 2009a). In the previous section, m_R is created to represent a practical situation with the most accuracy. However, only a simulation model itself is insufficient to create what-if scenarios and execute the simulation model.

Data in ERP systems, further denoted by d_{ERP}, are another crucial input as the initial state to execute the simulation model. For instance, for a customer-oriented company, orders in ERP systems from customers are treated as input to pull the manufacturing process. The quantity of jobs will definitely influence the initialisation of a simulation model. Therefore, key data in ERP systems need to be transferred to simulation models as the initial state in real time or near real time.

Besides simulation models and initial states, manufacturing factors are also important and should be considered, as the behaviour of a physical system is influenced by them, further denoted by mf. For instance, if a machine breakdown happens while the simulation is running, in order to provide accurate results, this machine breakdown state needs to be updated to the simulation model immediately. Various methods have been used to obtain these real-time data. In the manufacturing environment, real-time sensors are widely used to obtain manufacturing data, such as the current working process of machines, the number of products being produced and machine breakdowns. Similarly, previous states can also be achieved to predict future tendencies by accessing historical sensor data.

All objects, parameters, and manufacturing factors are changeable throughout the model execution. Thus, by controlling values of them, different experiment scenarios can be created. In this way, each experiment scenario contains different assumptive parameters to achieve different objectives and results. For instance, in order to reduce the tardy jobs, various sequences of jobs are scheduled and simulated to explore the best sequence of jobs. Therefore, experiment scenarios can be generated to achieve various objectives by the

function f_S with a specific model m_i, an initial state d_{ERPi}, and manufacturing factor mf_i.

$$s_i = f_S\ (\ m_i, d_{ERPi}, mf_i\ ); \qquad \textbf{(4-6)}$$

4.1.3 Optimisation and Decisions

Experiment scenarios are executed with optimisation which is used to analyse and compare results to generate the ultimate optimum decisions. Optimisation, further denoted by objective function f_{OPT}, is the process of finding the combination of conditions resulting in the best possible solution. In this case, it aims to automatically find the best decision parameters while remaining subject to certain constraints and requirements, further denoted by c and r respectively. Constraint is a condition defined upon the parameters. The values of parameters must satisfy the defined constraints. On the other hand, requirement is an additional restriction, also known as an indicator, imposed on the solution found by the optimisation process. Requirements are checked at the end of each scenario to see if the optimised parameters or solutions are feasible.

After optimising and comparing with indicators, simulation models can generate some decision parameters, further denoted by p_d, to influence or control ERP systems and physical systems. For example, in order to reduce tardy jobs, the simulation model can generate the best sequence of jobs. The mathematical expression of optimisation is indicated as below:

$$p_{di} = f_{OPT}\ (\ s_i,\ c_i, r_i\ ); \qquad \textbf{(4-7)}$$

4.1.4 Mathematical Process

The mathematical model of the generic symbiotic simulation system then can be developed step by step:

$set\ obj_1, obj_2, \dots, obj_n\ and\ p_1, p_2, \dots, p_n;$ Define objects and parameters

$$\begin{cases} m_1 = f_M(\, obj_1, obj_2, \ldots, obj_n;\; p_1, p_2, \ldots, p_n); \\ m_2 = f_M(\, obj_1, obj_2, \ldots, obj_n;\; p_1, p_2, \ldots, p_n); \\ \ldots \\ m_n = f_M(\, obj_1, obj_2, \ldots, obj_n;\; p_1, p_2, \ldots, p_n); \end{cases}$$

Create different simulation models by model development function

$$set\ \nabla m_j = min\{\nabla m_i\} \quad \forall\ i\ \in\ (1,2,\ldots,n);$$

$$if\ \nabla m_R > \nabla m_j\ \ then\ \ m_R = m_j;$$

Continuously search the reference model

$$set\ d_{ERP}, mf;$$

Acquire data from ERP systems and physical system

$$\begin{cases} s_1 = f_S\ (\, m_1, d_{ERP1}, mf_1\,); \\ s_2 = f_S\ (\, m_2, d_{ERP2}, mf_2\,); \\ \ldots \\ s_n = f_S\ (\, m_n, d_{ERPn}, mf_n\,); \end{cases}$$

Create different experiment scenarios for various objectives by scenario development function

$$set\ c, r;$$

Define constraints and requirements

$$\begin{cases} p_{d1} = f_{OPT}\ (\, s_1,\ c_1, r_1\,); \\ p_{d2} = f_{OPT}\ (\, s_2,\ c_2, r_2\,); \\ \ldots \\ p_{dn} = f_{OPT}\ (\, s_n,\ c_n, r_n\,); \end{cases}$$

Execute each experiment scenario with optimisation and generate decision parameters to influence or control ERP systems

4.2 Objects and Subsystems

Considering the functions and capabilities of ERP-based symbiotic simulation systems, a hybrid generic framework is proposed based on the mathematical concept. It consists of four subsystems SSFS, SSADS, SSDSS, and SSCS to achieve various purposes. Besides the four subsystems, it also contains triggers and objects with different capabilities. All these subsystems, triggers and objects work together to access the data inside ERP systems, evaluate trigger conditions, create and run what-if scenarios, optimise and analyse simulated results, visualise real-time states, forecast the future, recommend solutions, and control ERP systems directly. The concept design of the generic framework is outlined in Figure 4-1. A detailed introduction of each subsystem and object is illustrated in the following sections of this chapter.

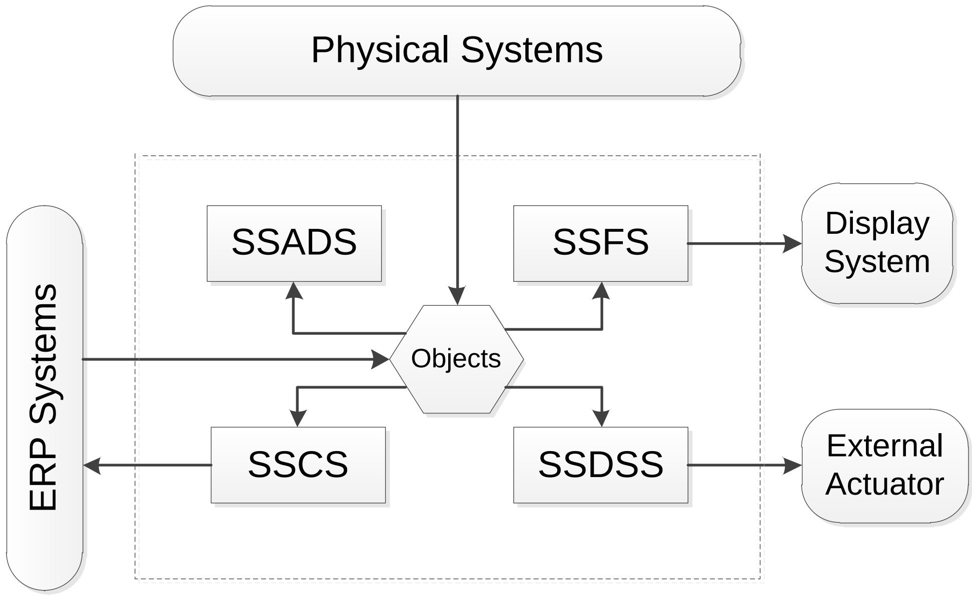


Figure 4-1 Concept design of the symbiotic simulation system

4.2.1 Trigger

Generally, there are three types of triggering mechanisms: reactive, preventive, and pro-active triggers. Reactive triggers mean simulation models are executed when triggering conditions are met. Preventive triggers are similar to reactive triggers as they need to observe triggering conditions as well. However, the difference is that preventive triggers rely on the forecast of physical systems. When simulation models identify that triggering conditions will happen in a period of time, the simulation model is launched. The pro-active trigger is launched periodically to invoke simulation models. Its purpose is to continuously improve the performance of simulation models (Aydt et al., 2009a).

In this generic framework, two reactive triggers and one pro-active trigger are defined.

- Operator trigger (OT) and anomaly trigger (AT)

Both OT and AT are reactive triggers and work in real time when triggering conditions are met.

OT is data operations in ERP systems as this may influence the results of simulation models. Symbiotic simulation systems continuously detect data changes inside ERP systems. If key data are changed, then the triggering condition is met and simulation models are invoked.

AT is the anomaly detection of physical systems. As mentioned in mathematical concepts, manufacturing factors may influence simulation results in many ways. Factors which cannot be predicted are called anomalies and they affect simulation results significantly. When an anomaly is detected, the triggering condition is met and simulation models are invoked. For instance, if a machine breakdown is detected, simulation models are then executed to find a new lead time. The new result data will be transferred to ERP systems to notify the manager and customer that the delivery date might be delayed because of the machine breakdown.

- Period trigger (PT)

The aim of the period trigger is to maintain the data inside ERP systems to keep them up-to-date. A timer is set as PT to launch simulation models periodically, e.g. hourly or daily. The length of the period of time depends on the model executing time and objectives of symbiotic simulation systems. When PT is triggered, symbiotic simulation systems access data in ERP systems and generate new simulation results accordingly.

4.2.2 Data Collection Object (DCO)

ERP systems focus on information integration of different business processes and provide relevant manufacturing data to simulation models (Li, 2011). Therefore, the essential function of integrating ERP systems is accessing and collecting the data inside ERP systems. DCO is used to automatically extract raw data from ERP systems, transform data to simulation models and present them in an accessible format for simulation models. Various DCO can be applied depending on the selected ERP systems and simulation tools. For instance, for SAP R/3 systems and Anylogic simulation models, Object Linking and Embedding (OEL) technology is a commonly used method to achieve this

purpose. Besides collecting data from ERP systems, it is also crucial to acquire real-time data from physical systems. In the manufacturing environment, manufacturing state data, such as machine working state or machine breakdown, are normally acquired by applying real-time sensors.

4.2.3 Data Fusion Object (DFO)

After acquiring data from both ERP systems and physical systems, an interface is needed to aggregate, store, and analyse all the data for simulation models. DFO is used to achieve this purpose and has the ability to carry out data categorisation, conversion of data format, identification and removal of duplicate or erroneous data, and data calculation. Normally, the initial state data and dynamic information obtained by DCO are aggregated and stored in DFO.

4.2.4 Model Management Object (MMO)

When triggering conditions are met, triggering notifications are sent to MMO. MMO then manage what-if scenarios and invoke the subsystems. At the start-up stage of simulation models, MMO is used to assign data stored in DFO to simulation parameters and update simulation models to the current or defined state. In simulation running stage, MMO delivers dynamic data from DFO to simulation models continuously.

4.2.5 Optimisation Object (OptO)

OptO aims to find the best possible sets of model specifications, such as input parameters, leading to optimum performance (April et al., 2003). In order to seek optimum decision parameters, SSCS and SSDSS subsystems need to request OptO to execute what-if scenarios and generate optimum decision parameters.

The goal of the optimisation process is to find the parameter values that result in a maximum or minimum of the objective function while respecting certain constraints and requirements. Objective function is a mathematical expression describing a relationship of the optimisation parameters or the result of a simulation that uses the optimisation parameters as inputs. It is normally pre-

defined in what-if scenarios based on the aim of the simulation models. Constraint is a condition defined upon the parameters. The values of input parameters must satisfy the defined constraints. Meanwhile, requirement is an additional restriction imposed on the solution found by the optimisation process. The constraints and requirements are evaluated using the suggested decisions to determine whether the optimum parameters are feasible.

Many methods have already been developed to enhance the performance of optimisation such as Gradient Based Search, Stochastic Optimisation, Heuristic methods, and Statistical methods. Because the development of optimisation methods, simulation models can now generate optimised decision parameters in near real-time. Some add-on optimisation engines have already been developed by software vendors such as proModel, LayOPT, OptQuest (Carson and Maria, 1997).

4.2.6 SSADS Subsystem

The SSADS subsystem aims to detect anomalies from both physical systems and simulation models by comparing with simulation models and the actual behaviours by accessing data in DFO. For physical systems, anomalies can be an unexpected event or abnormal behaviour, for example, machine breakdown or error operation. For simulation models, when discrepancy between the simulation model and measured behaviour is beyond a certain tolerance, it is considered as an anomaly. The SSADS subsystem is comprised of a reference model and an anomaly evaluating system. The reference model is used to compare with ERP systems and physical systems to find any discrepancy. The anomaly evaluating system is used to evaluate whether the discrepancy is beyond certain tolerance or criteria.

The basic structure and workflow are shown in Figure 4-2.The SSADS subsystem continuously accesses the data in DFO and compares them with the reference model. If any discrepancy is found, the anomaly evaluating system will analyse the discrepancy to confirm whether it is beyond pre-defined thresholds. If an anomaly is confirmed, the SSADS subsystem will send an

anomaly notification to MMO which will invoke SSFS, SSDSS, and SSCS subsystems in order to provide more accurate results.

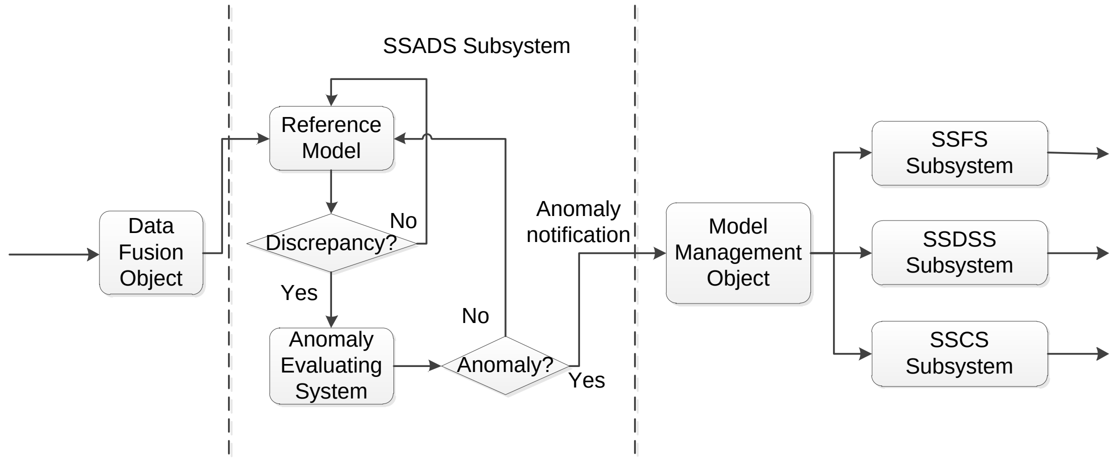


Figure 4-2 Structure and workflow of SSADS subsystem

4.2.7 SSFS Subsystem

The SSFS subsystem aims to visualise simulation results and generate prediction data. The essential part of the SSFS subsystem is what-if scenarios which can forecast future activities. Sometimes, it may contain animation systems to generate 2D or 3D animation for better visualisation purposes.

The basic structure and workflow is shown in Figure 4-3. MMO delivers static and dynamic data to the SSFS subsystem and invokes the what-if scenarios. The SSFS subsystem generates visualisation and prediction data to a display system, such as a monitor.

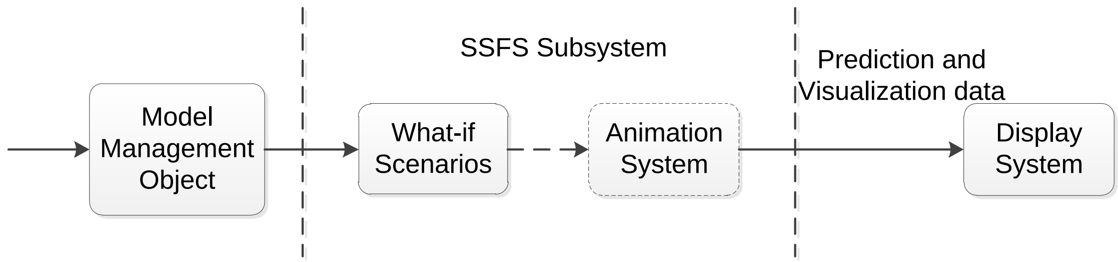


Figure 4-3 Structure and workflow of SSFS subsystem

4.2.8 SSDSS Subsystem

The SSDSS subsystem is used to optimise, analyse, and generate suggested decisions. However, the SSDSS subsystem does not contain actuators to execute the suggested decisions. Therefore, the aim of the SSDSS subsystem is to support external actuators rather than implementing decisions to ERP systems and physical systems directly. The reason why the symbiotic simulation system needs external actuators is that decisions generated by the SSDSS subsystem are just recommended and need external decision makers to consider other requirements. For instance, after running simulation models, the SSDSS subsystem suggests a workshop needs to work overtime to meet a due date. In this situation, the symbiotic simulation system cannot execute the decision directly because it regards many other issues, such as labour available. For this case, the external actuator could be the workshop manager who receives the suggested decisions as references. Base on the suggested decisions, the manufacturing manager either arranges another shift or contacts customers to delay the delivery date.

The SSDSS subsystem is comprised of what-if scenarios and an analysis system. The what-if scenarios along with the OptO are used to find the optimum decisions for certain problems. An analysis system is used to decide whether these decisions are feasible by comparing with certain indicators. The basic structure and workflow of the SSDSS subsystem is shown in Figure 4-4. MMO delivers initial state and dynamic information to the SSDSS subsystem and invokes it. The SSDSS subsystem then requests OptO to execute the what-if scenarios to find the optimum decisions. Decisions are then investigated and compared by the analysis system to decide whether the decision is feasible for implementation. If the decisions are feasible, they will be transferred to an external actuator to be executed. Otherwise, the process returns to the what-if scenarios to generate other decisions.

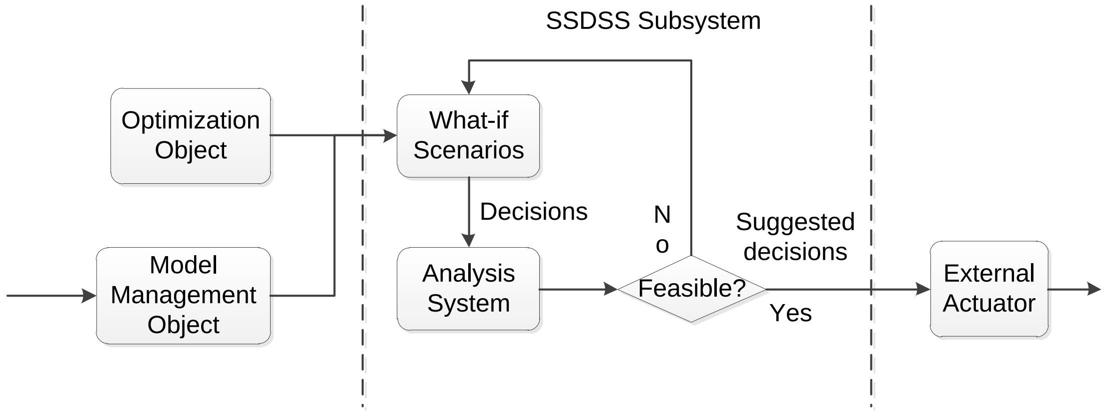


Figure 4-4 Structure and workflow of SSDSS subsystem

4.2.9 SSCS Subsystem

The only difference between the SSCS subsystem and the SSDSS subsystem is that the former contains the actuator itself, which has the ability to control ERP systems directly. For certain decisions, the symbiotic simulation system can control ERP systems directly instead of depending on external actuators. For instance, the estimated lead time generated by the symbiotic simulation system can input to ERP systems directly.

The basic structure and workflow of the SSCS subsystem is shown in Figure 4-5. MMO delivers initial state and dynamic information to the SSCS subsystem and invokes it. The SSCS subsystem then requests the optimisation object to optimise the what-if scenarios to find the optimum decisions. Decisions are then investigated and compared by the analysis system to decide whether the decision is feasible to influence ERP systems. If the decisions are feasible, they will be transferred to an actuator to execute it. Otherwise, the process returns to what-if scenarios to generate other decisions.

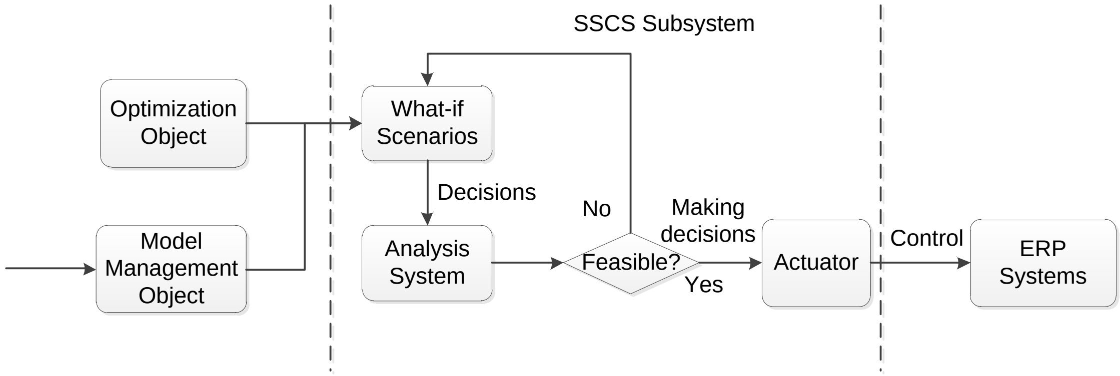


Figure 4-5 Structure and workflow of SSCS subsystem

4.3 Function Blocks

Figure 4-6 shows the function blocks of the generic framework. There are three triggers to launch the symbiotic simulation: OT, AT and PT. DCO continuously transfers real-time data from ERP systems and physical systems to DFO. The SSADS subsystem constantly monitors the information in DFO and compares it with the reference model to detect anomalies. MMO receives triggering notifications and uses the information in DFO to update the SSFS, SSDSS, and SSCS what-if scenarios and invokes the three subsystems. The SSFS subsystem generates future prediction and visualisation to display systems. The SSCS and the SSDSS subsystems request OptO to generate optimum decision parameters, which are used to control ERP systems directly or pass the decision parameters to an external actuator respectively.

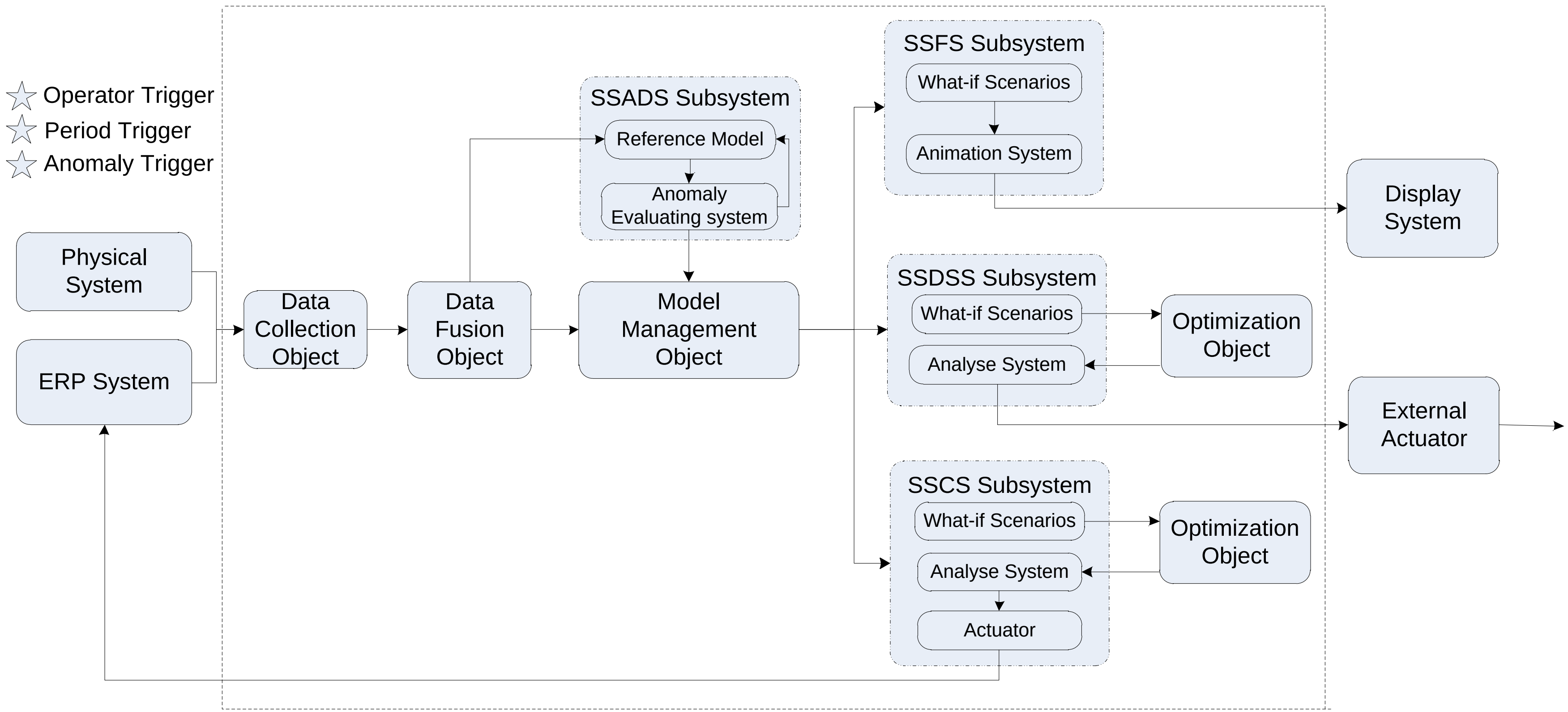


Figure 4-6 Function blocks of the generic framework

4.4 Workflow

The workflow contains two phases: the preparation phase and the simulation phase. Figure 4-7 shows the workflow of the generic framework.

- Preparation phase

DCO continuously delivers data from ERP systems and external environments to DFO which is used to aggregate, store, and analyse these data. Triggers are used to launch the whole simulation workflow. The first triggering condition is that operators change the key data of ERP systems. If the triggering condition is met, then a triggering notification will be sent to MMO. In addition, the SSADS subsystem continuously detects and evaluates the discrepancy. If the discrepancy exceeds certain tolerance, an anomaly notification will be sent to MMO. Otherwise, the process returns to detect the discrepancy. Additionally, PT sends a triggering notification to MMO periodically.

When MMO receive a triggering notification, it requests the information in DFO to update three subsystems to the current state. After that, MMO will invoke SSFS, SSDSS, and SSCS subsystems. In the running time, MMO will continuously update the simulation model with dynamic data.

- Simulation phase

When three subsystems are invoked by MMO:

For the SSFS subsystem, pre-defined what-if scenarios are executed and prediction data is generated. The animation system creates 2D or 3D animations and exports them to a display system, normally a monitor. The visualisation and prediction information of a manufacturing line can be used as an important reference for manufacturing planners.

For the SSDSS subsystem, pre-defined scenarios are executed and OptO is requested to explore optimum decision parameters. After that, the decision parameters are evaluated and compared with certain inductors. If the parameters are feasible, they will be exported to an external actuator. If not, the process returns to run the what-if scenarios to find other decision parameters.

For the SSCS subsystem, pre-defined scenarios are executed and optimised by OptO to get the optimum decision parameters. After that, the decision parameters are evaluated and compared with certain inductors. If the parameters are feasible, they will be executed by an internal actuator to control the ERP system directly. If not, the process returns to run the what-if scenarios to find other decision parameters.

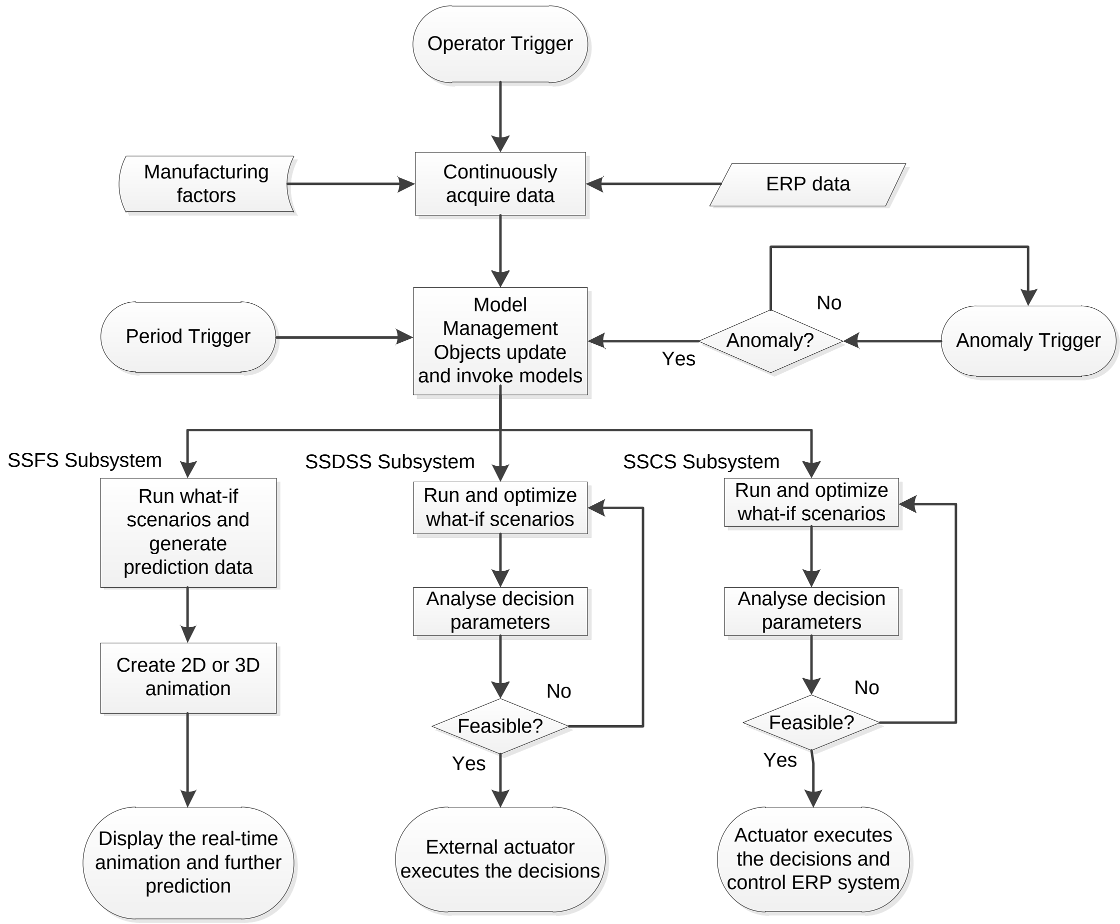


Figure 4-7 Workflow of the symbiotic simulation system

5 Case Example of the Generic Framework

The generic framework for integrating ERP systems to symbiotic simulation systems has already been developed in the previous chapter. In order to validate the applicability of the generic framework, a specific ERP-based symbiotic simulation system has been built as a case example based on a tube manufacturing shop floor. The SAP R/3 system as a widely applied information system is used as the ERP system and the simulation model is built by using Anylogic 6.

The main reason of using the SAP R/3 system is that it is recognised as the leader of ERP systems with 24% of the market share. It has a very high level of integration among different business processes which guarantees consistency of data through the system (Shaul and Tauber, 2013). Meanwhile, Anylogic 6 allows both discrete and continuous approaches to be implemented in one model and as such, more applications, such as computer performance evaluation and complex system design evaluation can be developed (Abu-Taieh and El-Sheikh, 2007). In addition, the native computer language environment of Anylogic 6 is Java, thus simulation models can be exported as standalone Java application. They also support Java codes to collaborate with external libraries and data sources (Anylogic, 2014).

In order to use this symbiotic simulation system to deal with practical manufacturing problems, an ERP-based tube manufacturing shop floor is chosen as a case study. All relevant manufacturing data of this tube shop floor are stored in a SAP R/3 system. In addition, a simulation model is developed based on the tube manufacturing shop floor by using Anylogic 6. Figure 5-1 shows the concept design of the symbiotic simulation system. The simulation model acquires relevant manufacturing data from the SAP R/3 system and the tube manufacturing shop floor. After executing simulation models, visualisation and prediction animations, suggested decision parameters and direct controlling parameters are generated to improve the performance of the tube manufacturing shop floor. In this way, the SAP R/3 system is integrated to a

symbiotic simulation system and can be mutually beneficial with the simulation model.

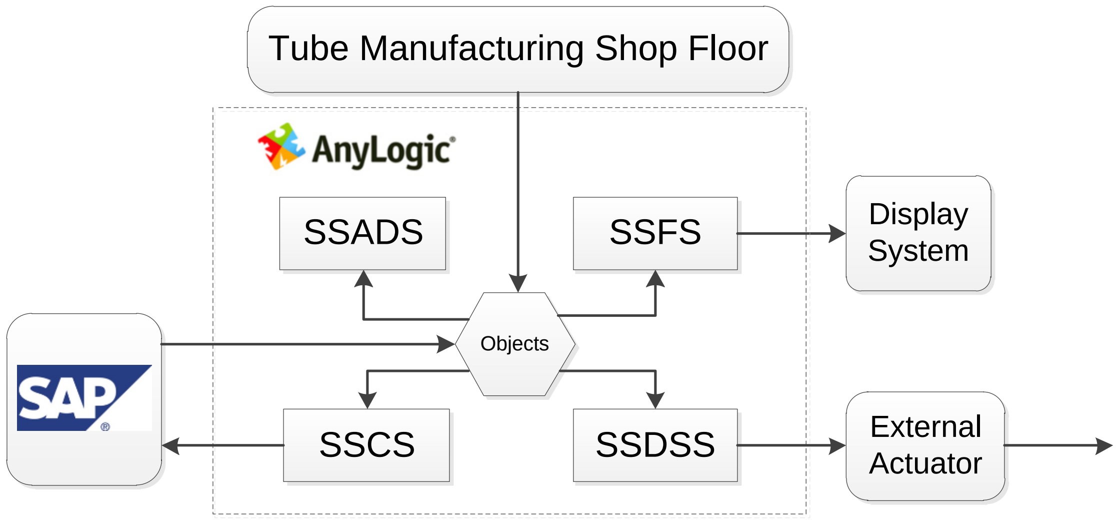


Figure 5-1 Concept design of the symbiotic simulation system

5.1 Introduction of the Tube Manufacturing Shop Floor

This tube manufacturing company is a real company based in China, which has been designing and manufacturing ducting and tubing components for many years, especially in the designing, manufacturing and testing aircraft ducting systems involving fuel, anti-ice and environmental control systems.

In this study, data are collected and summarised to form a concise tube manufacturing shop floor. The shop floor produces 9 different tube variations from raw materials to products. Three different raw materials are used in this workshop: corrosion resistant steel (CRES), titanium (Ti), and aluminium (Al). CRES and Ti are used to produce high-temperature and high-pressure tubes. Meanwhile, Al is used to produce lower temperature tubes.

The shop floor contains two main working areas: a preparing area and an assembly area. The preparing area contains cleaning tanks, job distribution conveyors and cutting machines. The main working content in the preparing area is cleaning raw materials and cutting them to standard parts. The shop floor normally receives batches of raw materials. After cleaning, raw materials

are delivered by job distribution conveyors based on a set of probabilities to different cutting stations. Each cutting station is operated with different efficiencies. After cutting, standard parts are produced and stored in the storage rack.

The assembly area contains a tube manufacturing line which is comprised of six distinct working centres. Each working centre contains two or more machines and performs certain working content. The shop floor is customer-oriented which means that when customer orders are received, workers pick up standard parts in the storage rack and start producing in the manufacturing line. There are 9 types of tube. Various tubes have different routings based on their variations. Figure 5-2 shows the layout of the tube manufacturing shop floor. Table 5-1 shows the main working contents of the six working centres. Table 5-2 shows different types of tube and routes with a brief description.

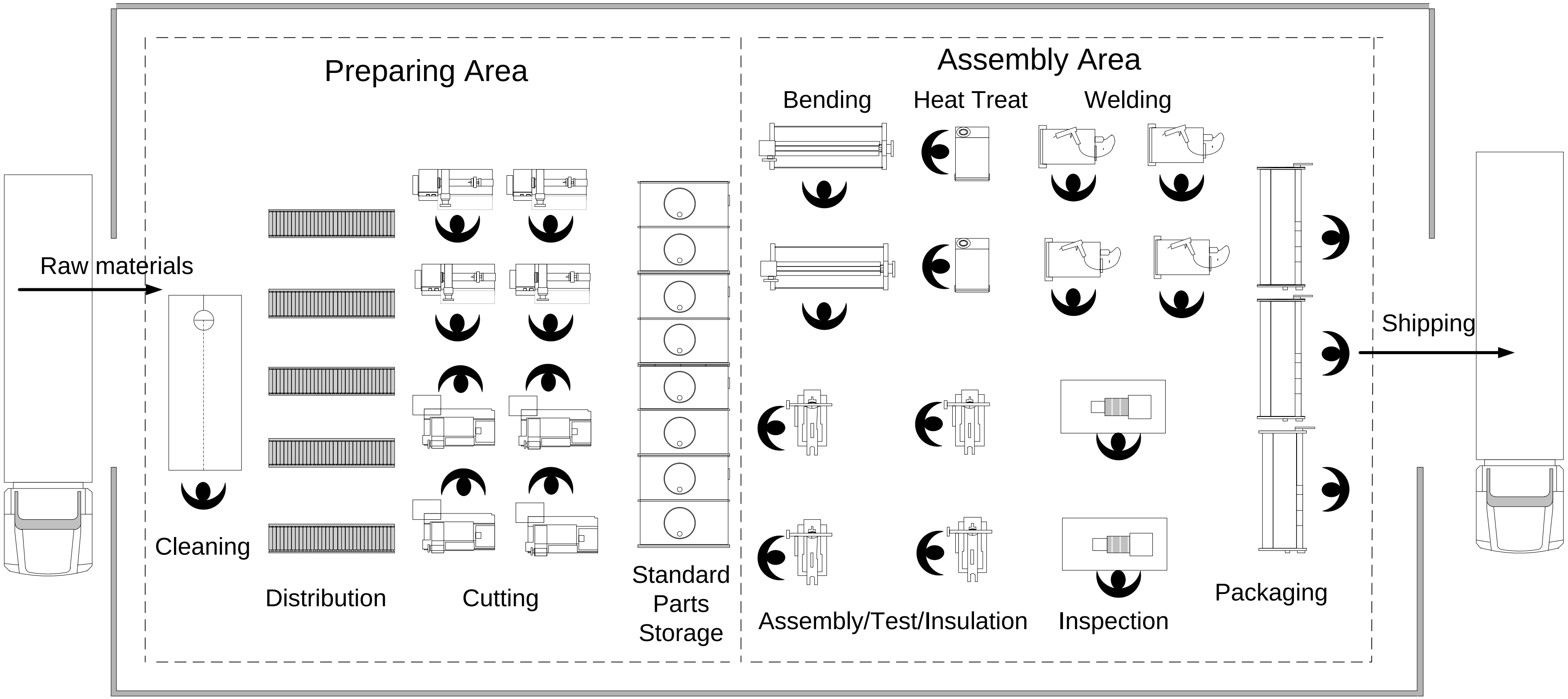


Figure 5-2 Layout of the tube manufacturing shop floor

Table 5-1 Contents of each working centre in the manufacturing line

Serial number	Working contents
WC 1310	Bending
WC 1410	Heat treat
WC 1420	Welding
WC 1510	Tube assembly / Test / Insulation
WC 1520	Inspection
WC 1610	Packaging / Shipping

Table 5-2 Routes of different tube variations

Type	Description	Route
C100	CRES without fitting	1310 – 1410 – 1420 – 1510 – 1520 – 1610
C101	CRES with1 Swage fitting	1310 – 1410 – 1520 – 1420 – 1510 – 1610
C102	CRES with 2 Swage fittings	1310 – 1410 – 1520 – 1510 – 1610
A100	Al without fitting	1310 – 1410 – 1420 – 1510 – 1610
A101	Al with 1 Swage fitting	1310 – 1410 – 1420 – 1510 – 1610
A103	Al with 2 Swage fittings	1310 – 1410 – 1420 – 1510 – 1610
T100	Ti without fitting	1310 – 1410 – 1420 – 1510 – 1610
T101	Ti with 1 Swage fitting	1310 – 1410 – 1520 – 1510 – 1610
T102	Ti with 2 Swage fittings	1310 – 1410 – 1420 – 1510 – 1610

5.2 Relevant Manufacturing Data in the SAP R/3 system

In this thesis, the SAP R/3 system is emulated by Microsoft Excel spreadsheets which store raw material delivery information and customer orders in detail. Through some techniques, key data in the SAP R/3 system can be mapped to Microsoft Excel spreadsheets. When an operator changes data in the SAP R/3 system, certain fields of spreadsheets will be adjusted accordingly. When data inside spreadsheets are changed by the simulation model, data inside the SAP R/3 system will be adjusted accordingly too. How to achieve this function will be illustrated in detail later.

The SAP R/3 system contains three kinds of data: raw material delivery information, exit probabilities of job distribution conveyors, and customer order.

The raw material delivery information consists of raw material types, delivery time, and quantity. At the delivery time, the corresponding quantities of certain raw materials are received by the tube manufacturing work floor. For the simulation model, raw material delivery information was loaded and at the specific delivery time, the simulation model injects corresponding quantities of certain raw material entities. Table 5-3 shows an example of raw material delivery information.

Table 5-3 Example of raw material delivery information

Raw material type	Delivery time	Quantity
CRES	9.00 AM 20/09/2014	80
Al	13.00 PM 25/09/2014	100
Ti	13.00 PM 26/09/2014	80
CRES	9.00 AM 28/09/2014	100
Ti	13.00 PM 28/09/2014	100

The jobs distribution conveyors transfer raw materials to different cutting stations based on their exit distribution probabilities. Table 5-4 shows an example of exit probability of the jobs distribution conveyors. In this case, the jobs distribution conveyor dynamically exports 34%, 18%, 22%, 14%, 12% of raw materials to according exit ports (cutting stations) respectively.

Table 5-4 Example of exit probability of conveyor system

Exit port	Exit port 1	Exit port 2	Exit port 3	Exit port 4	Exit port 5
Probability	0.34	0.18	0.22	0.14	0.12

Customer order contains tube type, ERP lead time, and due date. It indicates which type and when the customer wants to receive products. ERP lead time is experiential data pre-defined by operators based on experience. When the customer order is created, the workshop starts producing. The goal of the tube manufacturing shop floor is to deliver products before the due date. Table 5-5 shows an example of a customer order. There are two columns left for inputting simulation results.

Table 5-5 Example of customer order

Type	ERP lead time		Due date		Simulation results	
Type	Hours	Days	Hours	Days		
C100	48	2	72	3		
C101	48	2	48	3		
C102	72	3	72	4		
A100	24	1	96	4		
A101	48	2	72	3		
A102	48	2	72	3		
T100	48	2	72	3		

5.3 Data Collection

In order to acquire manufacturing relevant data and load simulated results in the SAP R/3 system, Microsoft Excel is chosen as a bridge. Integration between the SAP R/3 system and other data software is nothing new. One crucial reason for choosing Microsoft Excel is that it is more advanced in summarising and analysing data (Ignatiadis and Nandhakumar, 2009). Meanwhile, Anylogic 6 has a connectivity tool called Excel File which can provide easy platform-independent access to Microsoft Excel files within simulation models. Therefore, Microsoft Excel, which is considered as the industry standard of spreadsheets and has the advantages of widespread usage and compatibility for cross-platform, is chosen as the intermediary to exchange data between the SAP R/3 system and the simulation model. For manufacturing factors, the simulation model normally acquires real-time or near real-time data by applying sensors. This technology is well developed and has been used for many years.

Therefore, Microsoft Excel spreadsheets, Connectivity tool in Anylogic 6 and real-time sensors work together as the DCO to acquire data from the SAP R/3 system and the tube manufacturing shop floor. An emulated connection between the SAP R/3 system and Microsoft Excel spreadsheets is proposed instead of actually developing it.

5.3.1 Connection between the SAP R/3 System and Microsoft Excel Spreadsheets

SAP and Microsoft have worked together to use Object Linking and Embedding (OLE) to transfer and share information between SAP and Microsoft's products. OLE is the technology for transferring and sharing information among different applications. It is often used to integrate other applications with SAP systems, thus extending the functionality beyond its own essential capabilities (Anderson and Larocca, 2005). The SAP Assistant is the OLE interface for calling SAP functions for other applications and exposes both ActiveX controls and OLE object classes for managing and transferring data. In addition, almost all compiling languages support OLE, such as C++, Java, Microsoft .NET Visual Basic and so on. Therefore, it gives developers enough space to access and input data in the SAP R/3 system (Anderson, 2011). Figure 5-3 shows the overview of using OLE to connect them.

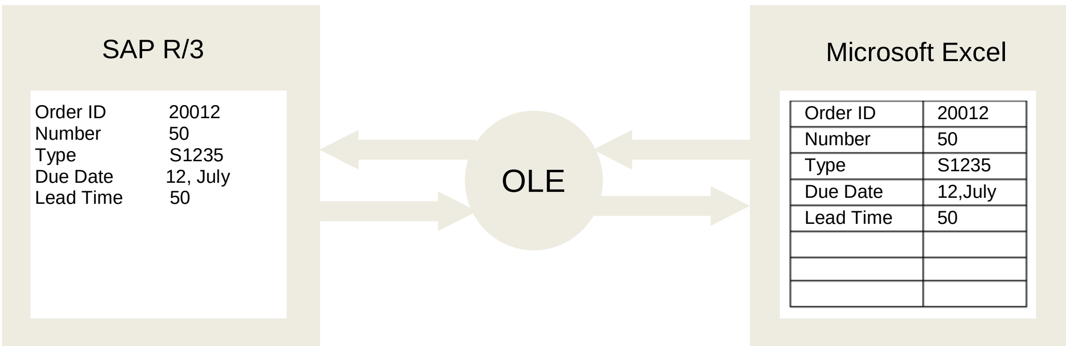


Figure 5-3 Overview of using OEL to connect SAP R/3 systems and Microsoft Excel spreadsheets

There are many tutorial books explaining how to export data from SAP R/3 systems to Microsoft Excel spreadsheets by using OLE. For instance, front-door export function can be used directly to convert manufacturing relevant data into Microsoft Excel spreadsheets (Mazzullo, 2006). Also the Query tool can be used to export data from SAP R/3 systems to Microsoft Excel spreadsheets (Anderson and Larocca, 2005). Meanwhile, based on OLE, many software vendors have developed tools to deal with the data transaction between SAP

R/3 systems and Microsoft Excel spreadsheets. For instance, Winshuttle's SAP usability products and software enables users to work with SAP directly from Microsoft Excel without programming. The Winshuttle's built-in products Transaction and Query are automated tools to transfer required data between SAP R/3 systems and Microsoft Excel spreadsheets in real time (Winshuttle, 2014). Figure 5-4 shows the overview working progress of Winshuttle. A screenshot of Winshuttle Transaction is shown in Figure 5-5. It shows the captured data from a SAP R/3 system are mapped to a Microsoft Excel spreadsheet.

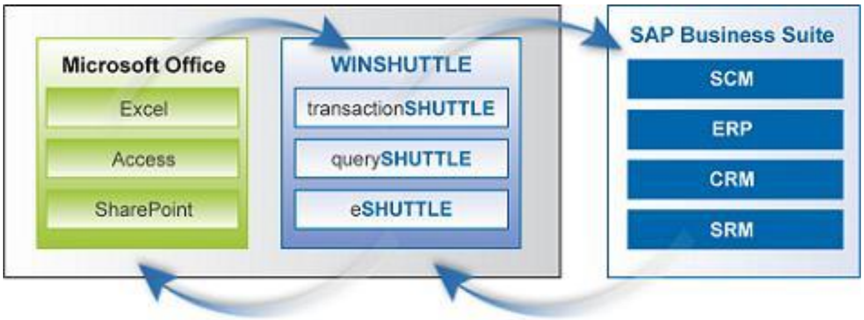


Figure 5-4 Overview of Winshuttle (Winshuttle, 2014)

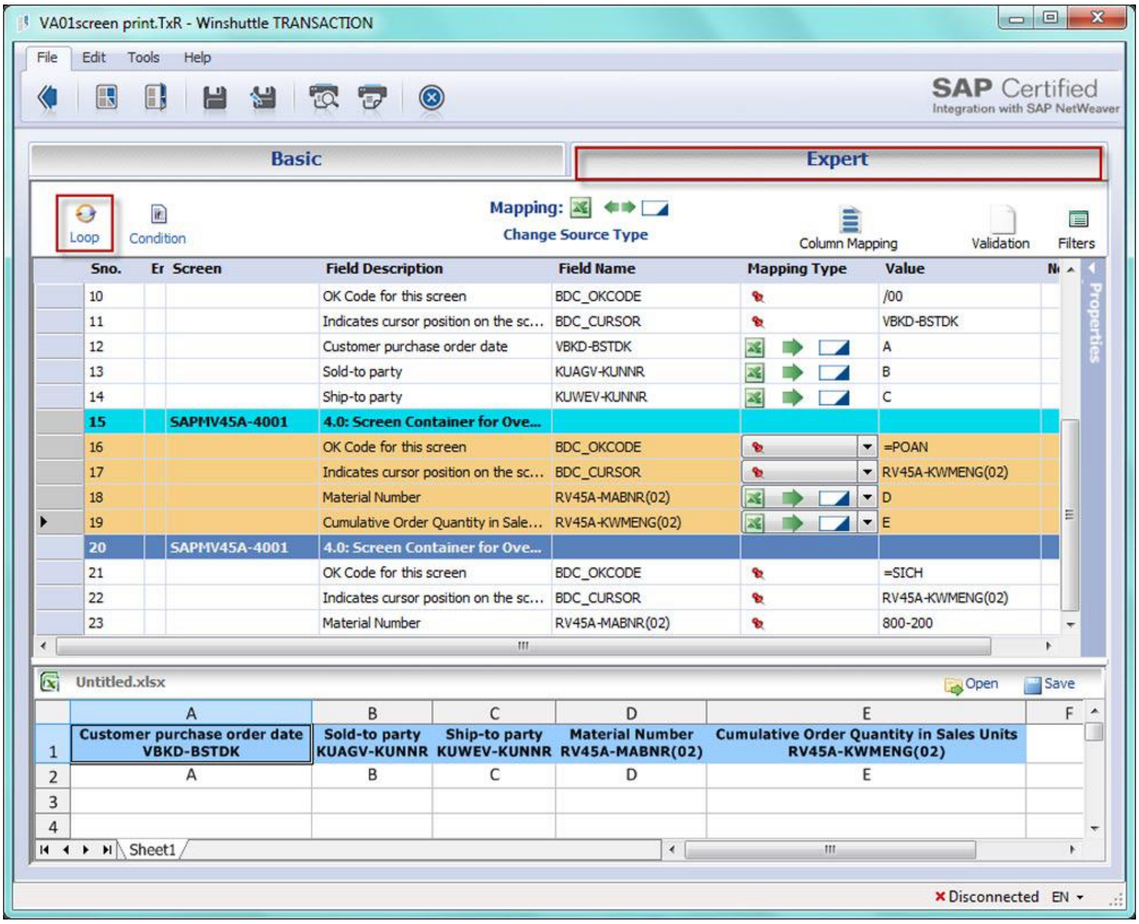


Figure 5-5 Screenshot of Winshuttle Transaction (Winshuttle, 2014)

Furthermore, these techniques have already been applied in other researchers' works. For instance, a Microsoft Excel user interface is proposed by Jänicke. It not only reads key data from SAP R/3 systems, but also writes changes back into SAP R/3 systems (Jänicke, 2001). The data management between SAP and Microsoft Excel is fully developed and mature. Therefore, this research uses Microsoft Excel spreadsheets to emulate the SAP R/3 system.

5.3.2 Connection between the Simulation Model and Microsoft Excel Spreadsheets

A built-in connectivity tool Excel File of Anylogic 6 is used to exchange data between the simulation model and Microsoft Excel files. Through programming, it allows the simulation model to read and load Microsoft Excel files automatically. In this way, data of raw material delivery information and the customer order can be transferred to the simulation model automatically. Figure

5-6 shows the overview connection between the simulation model and Microsoft Excel files.

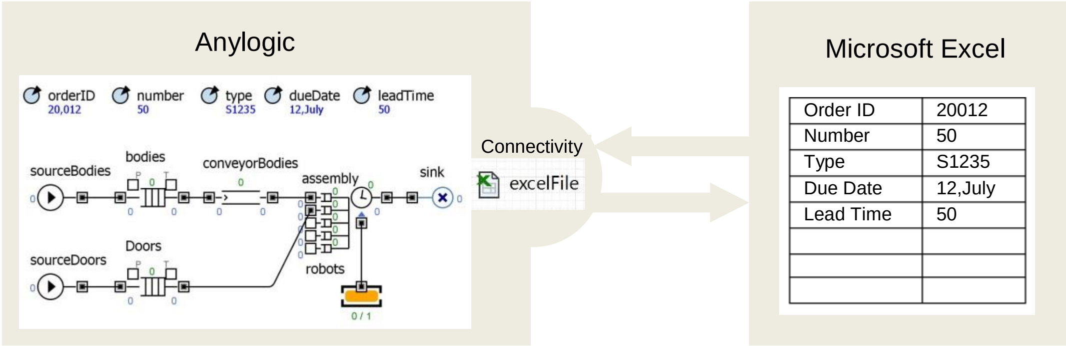


Figure 5-6 Overview connection of the simulation model and Microsoft Excel spreadsheets

5.3.2.1 Overall Data Transmission Structure of the Symbiotic Simulation System

In this research, a method was proposed to connect the SAP R/3 system and the simulation model by using Microsoft Excel spreadsheets as an intermediary, while it is not the only way to implement it. There are many other methods that can be used to accomplish this purpose. For example, sharing the same database between the SAP R/3 system and the simulation model or creating a live cache in a user’s PC to exchange the key data between the SAP R/3 system and the simulation model (Moon and Phatak, 2005; Jänicke, 2001). To summarise, the connection structure of the symbiotic simulation system can be described as below:

The simulation model acquires key data from the SAP R/3 system by using Microsoft Excel spreadsheets as a bridge. The key data are set to parameters, variables and collections as inputs to execute the simulation model. After running the simulation, the simulated results are transferred to Microsoft Excel spreadsheets to control and modify the SAP R/3 system. The connection structure of the symbiotic simulation system is shown in Figure 5-7.

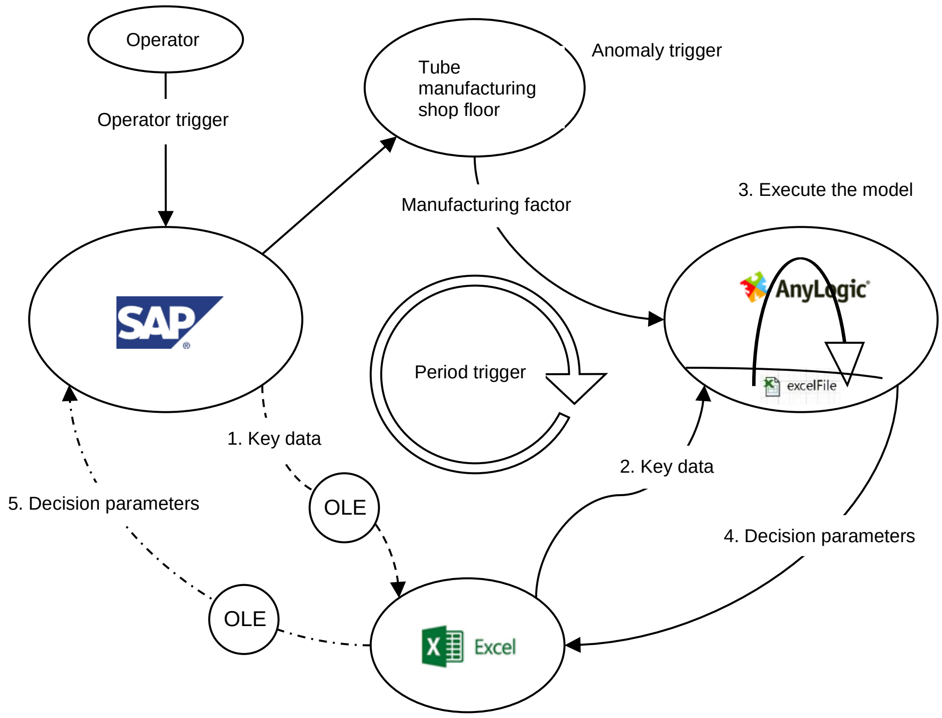


Figure 5-7 Connection structure of the symbiotic simulation system

5.4 Data Fusion

Anylogic 6 provides various kinds of tools to store and analyse data. After acquiring data, Parameters, Variables, and Collections are used to store them. Parameters are generally used to represent some characteristics of the modelled objects, such as cycle time of machines. Variables are frequently used to store simulation results or object characteristics changing over time, such as simulated lead time. Collections represent a group of objects and are used to store, retrieve and manipulate aggregate data, such as queue or sequence. Some java functions are defined to analyse acquired data, such as removal of duplicate data, or calculation. Figure 5-8 is a screenshot of the interface built in the simulation model. Parameters, variables and collections, along with some Java programming functions, are used together as DFO.

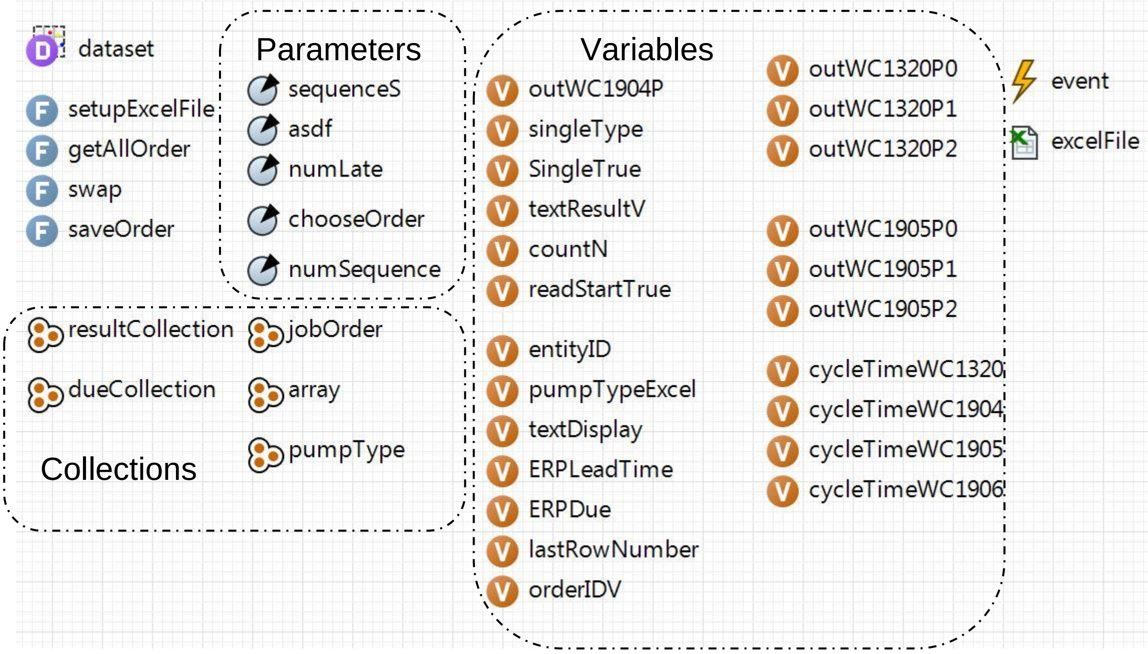


Figure 5-8 Screenshot of the interface in the simulation model

In this simulation model, four functions setupExcelFile, getAllOrder, swap, and saveOrder are defined. Function setupExcelfile is used to load specific Excel spreadsheets and assign certain values of the spreadsheets to parameters and variables. It is also used to convert formation of data, such as convert integer to double. Function getAllOrder and function swap are used to calculate all the possible sequences of jobs (customer order). Function saveOrder is used to store all the possible sequences of jobs to according collection. Figure 5-9 shows the partial screenshot of the function codes.

▾ Function body

```
//use this code to read Excel
excelFile.readFile();
lastRowNumber=excelFile.getLas
readStartTrue=true;
SingleTrue=false;
for (int i =0;i<lastRowNumber-
resultCollection.add(i,0);
dueCollection.add(i,0);
array.add(i,i+1+"");
}
getAllOrder(0, array.size()-1);
sequenceS=jobOrder.get(chooseOrder);
char[] jobOrderChar=sequenceS.toCharArray();
for(int i=3;i<=lastRowNumber;i++){
orderIDV=2+Integer.parseInt(jobOrderChar[i-3]+"");

pumpTypeExcel = excelFile.getCellStringValue( "A"+orderIDV );
ERPLeadTime=(int)excelFile.getCellNumericValue("B"+orderIDV);
ERPDue=(int)excelFile.getCellNumericValue("D"+orderIDV);
//end code
```

▾ Function body

```
//loop function to find new sequence
if (from == to) {
      return;
   }
   String tmp = array.get(from);
   array.set(from,array.get(to));
   array.set(to,tmp);
```

▾ Function body

```
//calculate all the possible seque
 if (begin == end) {
      saveOrder(end);
   } else {
      for (int i = begin; i <= end
        swap(begin, i);
        getAllOrder(begin + 1, end
        swap(i, begin);
      }
   }
```

▾ Function body

```
String orderTemp="";
for(int i=0;i<=outNum;i++){
orderTemp+=array.get(i);
}
jobOrder.add(countN,orderTemp);
countN++;
```

Figure 5-9 Partial screenshot of the function codes

Initialised data such as routes of different tubes, cycle time of machines, customer orders, and raw material delivery information are stored by parameters and variables. Collections are used to store all the possible job sequences and simulated lead time of each job. Figure 5-10 shows the screenshot of parameters, variables and collections with values in simulation running time.

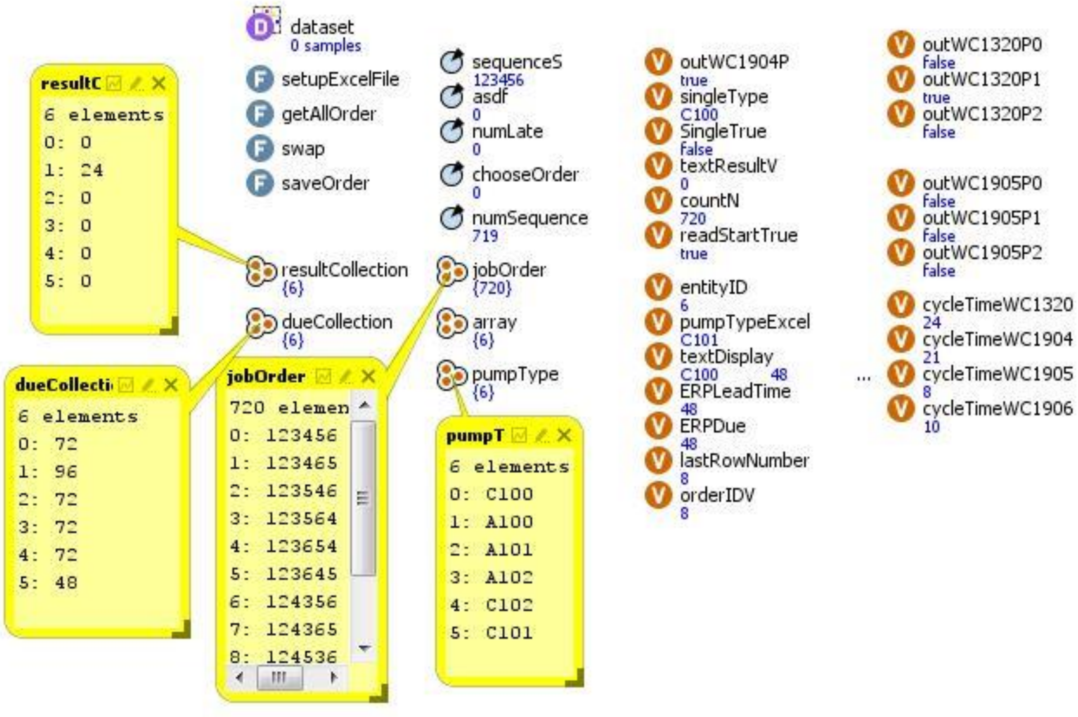


Figure 5-10 Screenshot of parameters, variables, and collections with values

5.5 Trigger Condition

According to the generic framework, three triggers are defined in the symbiotic simulation system as below:

- Operator trigger

Through OLE technics, key data in the SAP R/3 system can be mapped to Microsoft Excel spreadsheets. When the operators change the data in the SAP R/3 system, certain fields of Microsoft Excel spreadsheets will be adjusted accordingly. Therefore, in this research, the operator triggering condition is changing raw material delivery information or customer orders in the Microsoft Excel spreadsheets.

- Anomaly trigger

In this research, machine breakdown is recognised as the anomaly. In practical cases, a real-time sensor can be simply applied to detect machine breakdown

and send notifications to the simulation model. However, because of resource limitation, in this research, an emulate button is used to present machine breakdowns in the tube manufacturing shop floor. When the button is pressed, one random cutting machine breaks down and the triggering condition is met.

- Period trigger

Anylogic 6 contains an Event tool which is normally used to schedule some action in simulation models. In this research, a timeout triggered event is built as PT which sends a triggering notification to MMO periodically. The period of time of this event can be set by users.

5.6 Model Management

In the start-up period of simulation model, Java codes are programmed as MMO to update the simulation model with data stored in parameters, variables, and collections. Based on the triggering condition, different what-if scenarios are invoked by programming. Figure 5-11 shows the screenshot of partial Java codes that are used to update the simulation model and invoke different subsystems.

▾ Agent actions

On startup:

```
//fill the dataset with some sample data
//updata the simulation model with the data in parameters, variables, and collections.
//invoke different what-if scenarios
excelFile.readFile();
textDisplay="";
lastRowNumber=excelFile.getLastRowNum("Sheet1");
readStartTrue=true;
SingleTrue=false;
for (int i =0;i<lastRowNumber-2;i++){
resultCollection.add(i,0);
dueCollection.add(i,0);
array.add(i,i+1+"");
}
getAllOrder(0, array.size()-1);
sequenceS=jobOrder.get(chooseOrder);
char[] jobOrderChar=sequenceS.toCharArray();
for(int i=3;i<=lastRowNumber;i++){
orderIDV=2+Integer.parseInt(jobOrderChar[i-3]+"");

pumpTypeExcel = excelFile.getCellStringValue( "A"+orderIDV );
ERPLeadTime=(int)excelFile.getCellNumericValue("B"+orderIDV);
ERPDue=(int)excelFile.getCellNumericValue("D"+orderIDV);
textDisplay+= pumpTypeExcel+"            "+ERPLeadTime+
"                         "+ERPDue+"\n";
source.inject(1);
pumpType.add(i-3,pumpTypeExcel);
dueCollection.set(i-3,ERPDue);
}
numSequence=jobOrder.size()-1;
text1.setText(textDisplay);
```

Figure 5-11 Screenshot of the partial Java codes

5.7 Optimisation

Optimisation of Anylogic 6 is built on top of OptQuest Optimisation Engine. OptQuest Engine is an optimisation tool produced by an optimisation software and service vendor OptTek. State-of-the-art procedures and methods are used by OptQuest (OptTek, 2014). OptQuest is used as OptO to help the SSDSS and SSCS subsystems automatically acquire the best parameters of pre-defined scenarios, with respect to the pre-defined constraints and requirements.

5.7.1 Concepts in Optimisation Process

Optimisation parameters are model parameters to be optimised. The goal of requesting OptQuest Engine is to find optimum parameter values. Constraint is a condition defined upon the parameters. The values of parameters must satisfy the defined constraints. While requirement is an additional restriction, it is checked at the end of each scenario to see if the optimised parameters or solutions are feasible. The mathematical expression described a relationship between the optimisation parameters (or results of an operation) and the objective is called objective function. For instance, a simulation model is developed to seek minimised cycle time by purchasing a certain number of new machines, while restricting a certain capital investment. Meanwhile, the daily operation cost is hoped to be kept under a certain amount. In this case, the mathematical expression to minimise cycle time is objective function. The number of new machines is the optimisation parameter. The capital investment is constraint and daily operation cost is requirement.

5.7.2 Working Steps of OptQuest Engine

Optimisation is an iterative process while OptQuest Engine calculates possible solutions for the optimisation parameters in the simulation model. The objective function and constraints are evaluated using the suggested solutions. After iterations, a new set of possible solutions is calculated until the stop condition is met. For the case example, OptO is requested to find the optimised jobs distribution probability parameters in preparing area and optimised sequence of jobs to meet the customers' due date in the manufacturing line.

Figure 5-12 shows the screenshot of the optimisation experiment. Steps of requesting OptQuest engine to optimise the simulation model are shown below:

Step1: Define optimisation parameters.

Step2: Specify the objective function.

Step3: Define constraints and requirements.

Step4: Specify the optimisation stop condition.

Step5: Run the optimisation iterations.

Step6: Generate best feasible optimisation parameters

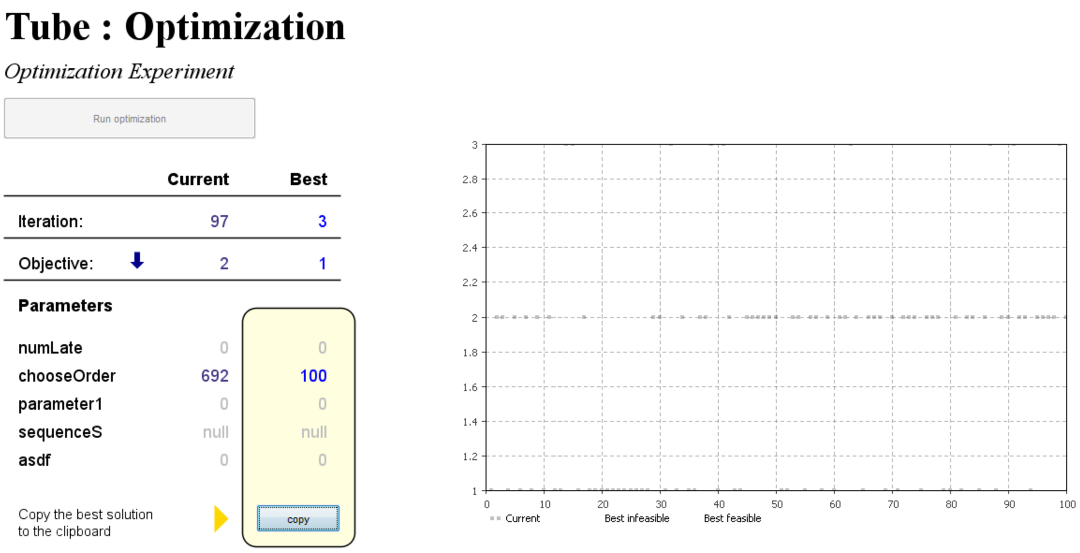


Figure 5-12 Screenshot of the optimisation experiment

5.8 SSADS, SSFS, SSDSS, SSCS Subsystems

For the tube manufacturing shop floor, a discrete event simulation model is built by Anylogic 6. The structural components including entities, enterprise libraries, activities and events, resources, variables, a calendar, control buttons, and statistics collectors and so on, are used to build the simulation model. By changing values of parameters and variables, different scenarios are generated based on their specific purposes.

- SSADS subsystem

For the SSADS subsystem, it contains a reference model and aims to detect the anomaly. However, because of resource limitation, an emulate button is used to present machine breakdown. In reality, a sensor can be simply applied to fulfil this function. When the breakdown button is pressed, one of the cutting machines then breaks down. An anomaly notification is generated to inform MMO invoking other subsystems.

- SSFS subsystem

For the SSFS subsystem, after running what-if scenarios, future working state and lead time will be generated. Anylogic 6 provides lots of 2D or 3D animations which can be used for visualisation. In this simulation model, tube manufacturing states are displayed in 2D and 3D visualisation. In addition, charts and histograms are used to present the working state of each working centre. Current, historical, and future states of the tube manufacturing shop floor can be easily observed through a monitor by people who are in charge. A text box is designed to display a simulated lead-time of each customer order comparing it with the due date.

- SSDSS subsystem

For the SSDSS subsystem, what-if scenarios of acquiring best sequence of jobs in a customer order are defined. After running the simulation, best sequence can be generated to reduce tardy jobs. The manufacturing manager as the external actuator can then deploy the jobs.

The job sequence parameter is set as the optimisation parameter in the SSDSS subsystem. The objective function is defined to achieve a minimum of tardy jobs. 500 optimisation iterations of job sequences are executed in order to get the best sequence. Figure 5-13 shows the screenshot of the SSDSS subsystem.

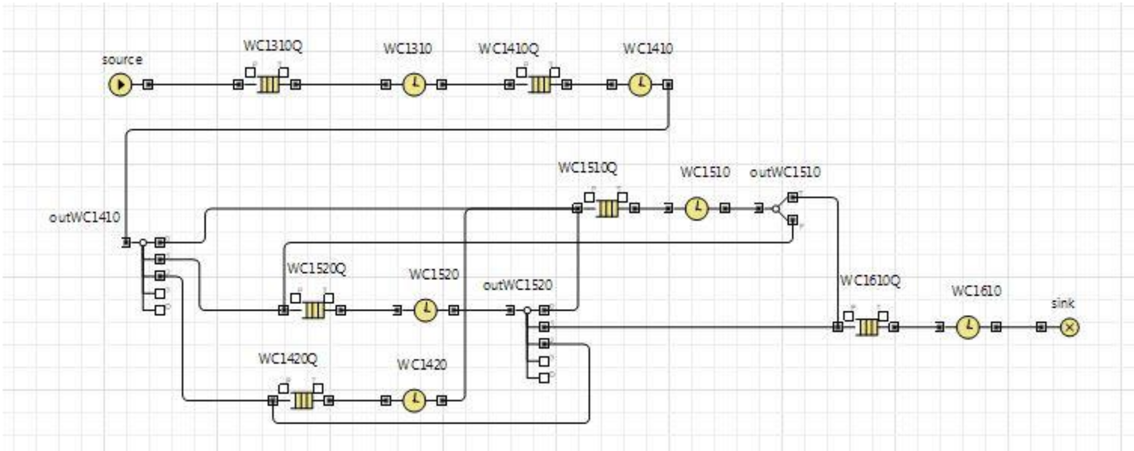


Figure 5-13 Overview of the SSDSS subsystem

- SSCS subsystem

For the SSCS subsystem, after running what-if scenarios, optimised distribution probability parameters are generated and transferred to the SAP R/3 system directly.

The distribution probability parameters are set as optimisation parameters. The objective function is designed to achieve minimum total lead time. Constraint is set as the totality of distribution probability parameters is 100%. 200 optimisation iterations are set to get the best set of distribution probability parameters. Figure 5-14 shows the overview of the SSCS subsystem.

Figure 5-15 shows the overview of the simulation model.

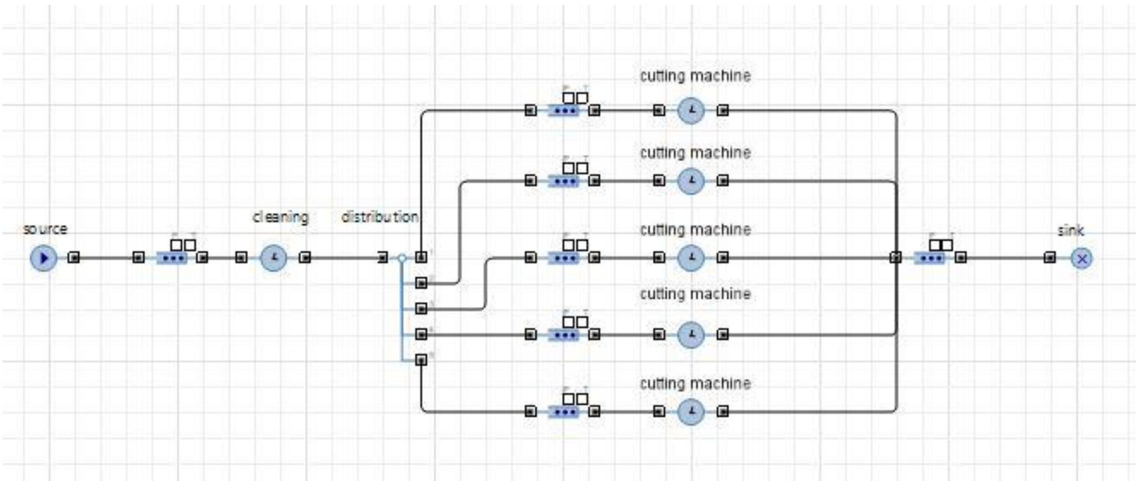


Figure 5-14 Overview of the SSCS subsystem

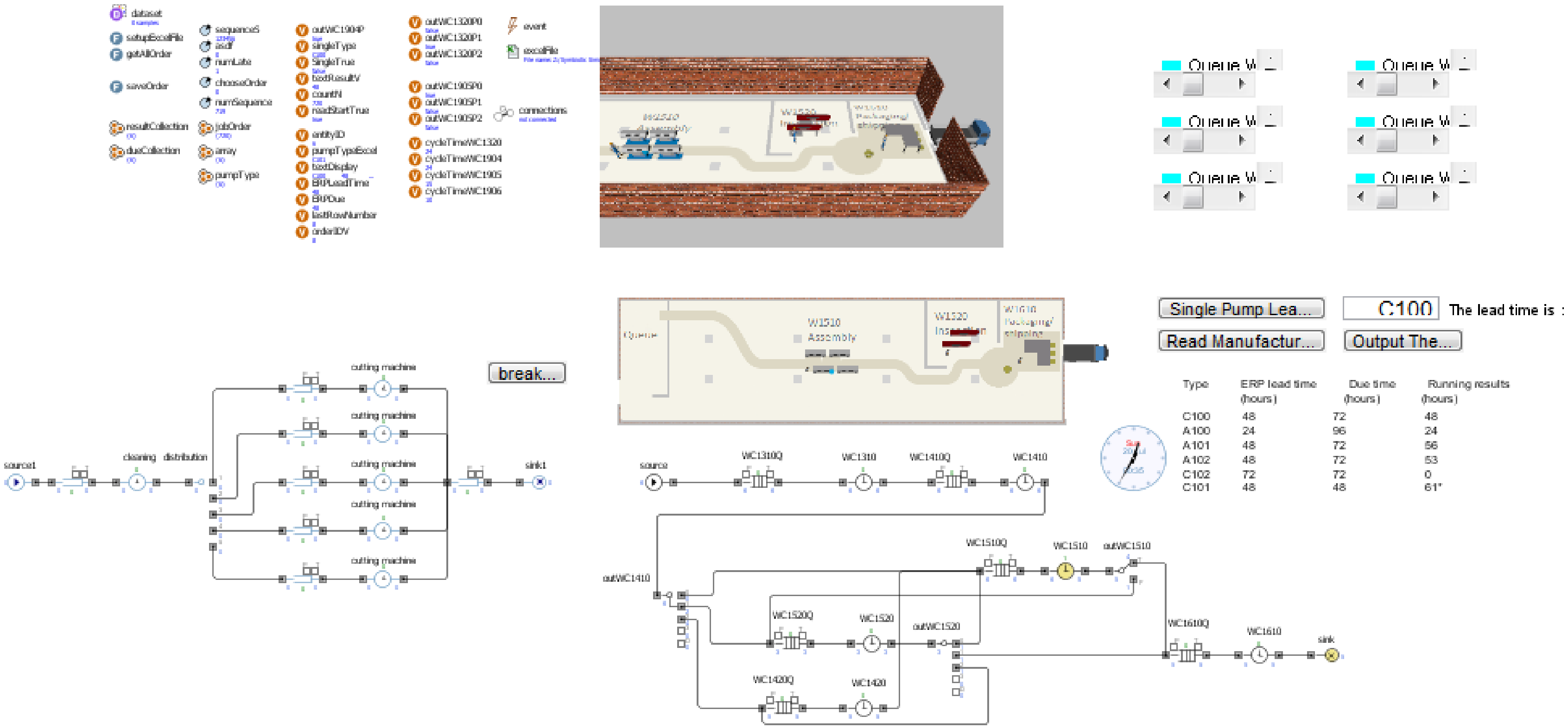


Figure 5-15 Overview of the simulation model

6 Experimentation

Based on the selected case, experiments are carried out to validate the feasibility, applicability and functions of the symbiotic simulation system. The aim is to validate that all the objects and subsystems of the symbiotic simulation system can work effectively comparing them with traditional simulation methods. Experiments focus on validating the following four aspects.

- All the triggers and object can work efficiently in the symbiotic simulation system.
- The symbiotic simulation system can display accurate prediction and visualisation data (the function of the SSFS subsystem).
- Suggested decision parameters can be generated for external actuator by the symbiotic simulation system (the function of SSDSS subsystem).
- The symbiotic simulation system can control the SAP R/3 system directly (The function of the SSCS subsystem).

In order to achieve these objectives, two sets of experiment are defined and carried out using the symbiotic simulation system based on the tube manufacturing shop floor.

6.1 Raw Material Experiments

As introduced previously, after cleaning raw materials, the distribution conveyors transfer raw materials to different cutting stations based on a set of probabilities. There are five cutting stations and each cutting station contains a certain type of cutting machines. Because the company purchased different types of cutting machines in different periods, various types of cutting machines have different efficiencies. The job distribution conveyors transfer the raw materials to different cutting stations according to a set of experiential probabilities which are stored in the SAP R/3 system in advance.

Raw material experiments are designed to validate that the symbiotic simulation system can respond in real time unlike off-line simulations. In addition, the SSCS subsystem's function is validated by generating and outputting the

optimised probability parameters of job distribution conveyors to the SAP R/3 system.

- Machine breakdown

The first experiment is to get the optimised distribution parameters when a machine breaks down. A button in the simulation model is used to emulate machine breakdown. When the button is pressed, a random cutting machine breaks down. When machine breakdowns are detected, MMO invokes the simulation model and OptO is requested to obtain the best probability values for each exit of the distribution conveyors. The objective is to get the least lead time of cutting batch raw materials. After running the simulation model, optimised parameters are outputted to the SAP R/3 system (Microsoft Excel spreadsheets) automatically. Because the predefined manufacturing parameters, such as cycle time of cutting machines, are stochastic. Ten experiments are executed in order to get convictive results. In a specific time, 50 pieces of raw material are received.

Scenario 1: Traditional off-line simulation

The simulation model cannot respond in real-time. After initialising the simulation model with optimised probability parameters of the distribution conveyors, it does nothing when the breakdown button is pressed.

Scenario 2: Symbiotic simulation

The symbiotic simulation system can respond in real-time. When the breakdown button is pressed, MMO receives the anomaly notification and invokes the subsystems. New optimised probability parameters are generated and transferred to the SAP R/3 system (Microsoft Excel spreadsheets) to control the distribution of raw materials.

The experiment result shows that the symbiotic simulation system can respond in real-time and reduce lead time efficiently. The average lead time of the ten experiments is reduced by 24.8 mins from 82.8 mins to 58 mins. Table 6-1

shows the total lead time (mins) for the 50 raw materials and Figure 6-1 shows the line chart of the simulation results.

Table 6-1 Lead time (mins) of each scenario

Series	Scenario 1	Scenario 2
1	96	66
2	91	53
3	65	53
4	86	58
5	69	52
6	81	61
7	80	62
8	83	54
9	88	63
10	89	58

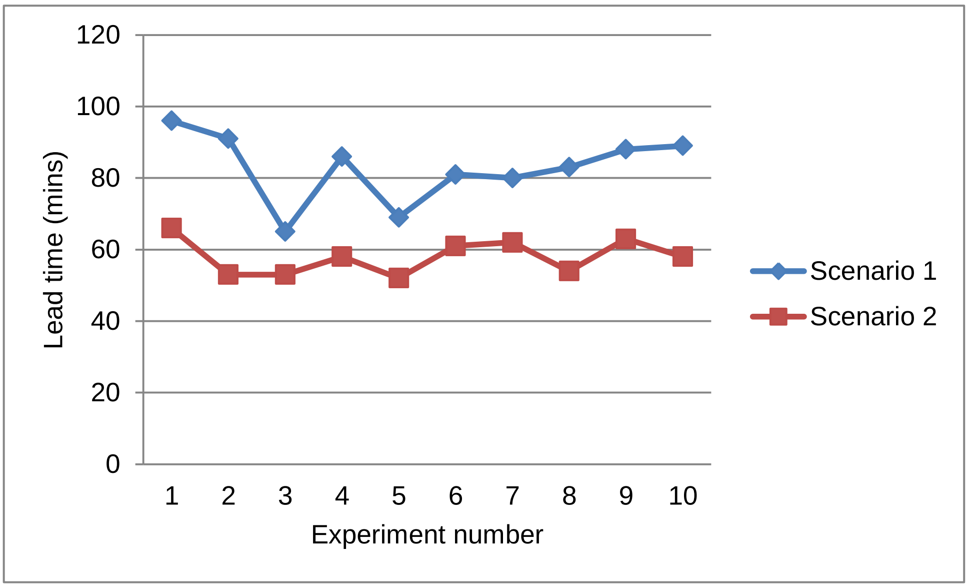


Figure 6-1 Line chart of the breakdown experiments

- Period updating

The second experiment is to validate that the symbiotic simulation system can be periodically updated and continuously improve performance of the shop floor. Three experiment scenarios are defined below:

Scenario 1: Traditional off line simulation

Use the experiential probability parameters of the conveyor system to initialise and run the simulation model.

Scenario 2: Symbiotic simulation

Use the experiential probability parameters of the conveyor system to initialise the simulation model. PT sends a period notification to MMO every 10 minutes. When a notification is received, MMO updates the simulation with dynamic data and invokes the subsystems. New optimised probability parameters are generated and transferred to the SAP R/3 system (Microsoft Excel spreadsheets) to control the distribution of raw materials.

Scenario 3: Symbiotic simulation

Use the experiential probability parameters of the conveyor system to initialise the simulation model. PT sends a period notification to MMO every 5 minutes. When a notification is received, MMO updates the simulation with dynamic data and invokes the subsystems. New optimised probability parameters are generated and transferred to the SAP R/3 system (Microsoft Excel spreadsheets) to control the distribution of raw materials.

The experiment result shows that the symbiotic simulation system can respond periodically and reduce lead time efficiently. Table 6-2 shows the lead time (mins) of each scenario and Figure 6-2 shows the line chart of the simulation result.

Table 6-2 Lead time (mins) of each scenario

Series	Scenario 1	Scenario 2	Scenario 3
1	82	50	45
2	91	48	45
3	65	52	47
4	86	50	46
5	69	48	50
6	81	52	46
7	80	52	48
8	83	55	52
9	88	50	46
10	89	52	46

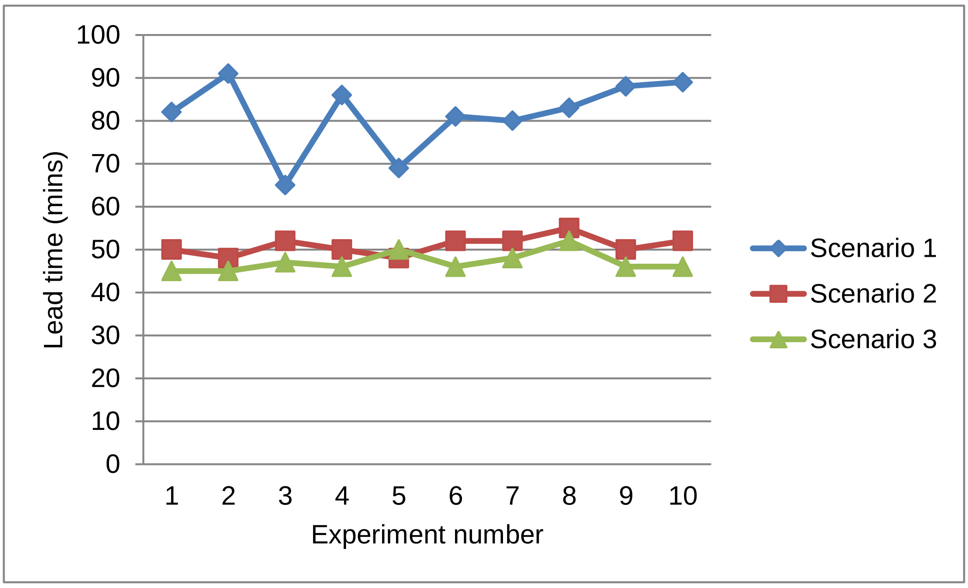


Figure 6-2 Line chart of the breakdown experiments

6.2 Customer Orders Experiments

As introduced previously, the tube manufacturing shop floor is customer-oriented. When sale operators upload new orders in the SAP R/3 system, the manufacturing line starts to produce corresponding tubes. Table 6-3 shows the predefined customer order in the spreadsheet. Experiential ERP lead time is inputted into the SAP R/3 system in advance and the due date is given by customers. In order to avoid penalty and build good cooperation with customers, the tube manufacturing company aims to deliver the corresponding tubes before due dates.

Customer orders experiments are designed to validate that the symbiotic simulation system can generate prediction data (the SSFS subsystem's function). In addition, the SSCS subsystem's function is validated by suggesting the best sequence of jobs to a manufacturing manager.

Table 6-3 Pre-defined customer order in the spreadsheet

Type	ERP lead time		Due date		Simulation results	
Type	Hours	Days	Hours	Days		
C100	48	2	72	3		
A100	24	1	96	4		
A101	48	2	72	3		
A102	48	2	72	3		
C102	72	3	72	4		
C101	48	2	48	3		

- Visualisation and Prediction

The first experiment aims to validate that the symbiotic simulation system can provide the real-time state and future prediction. The spreadsheet shown in table 6-3 is accessed by the symbiotic simulation model and the what-if scenarios of the SSFS subsystem are executed. After running the simulation model, the real-time and future states are shown in 2D or 3D animations. Figure 6-3 shows the screenshot of 2D visualisation. A clock is used to demonstrate time and a panel is used to control the simulation model. In addition, analysis tools are used to demonstrate statistic results. In this case, bar charts are used to show the number of entities in the queues before each working centre.

Furthermore, 3D animations are generated which are shown in Figure 6-4. 3D visualisation has the advantage in monitoring and analysing simulated results. Figure 6-5 shows the prediction of this customer order that C102 and C101 cannot meet the due date. The results can be output to the Microsoft Excel spreadsheet automatically as shown in table 6-4. The symbiotic simulation system continuously accesses data from spreadsheets and outputs a more accurate finish date to influence the SAP R/3 system. When tardy jobs are predicted, the symbiotic simulation system will remind the manufacturing manager with the delay time.

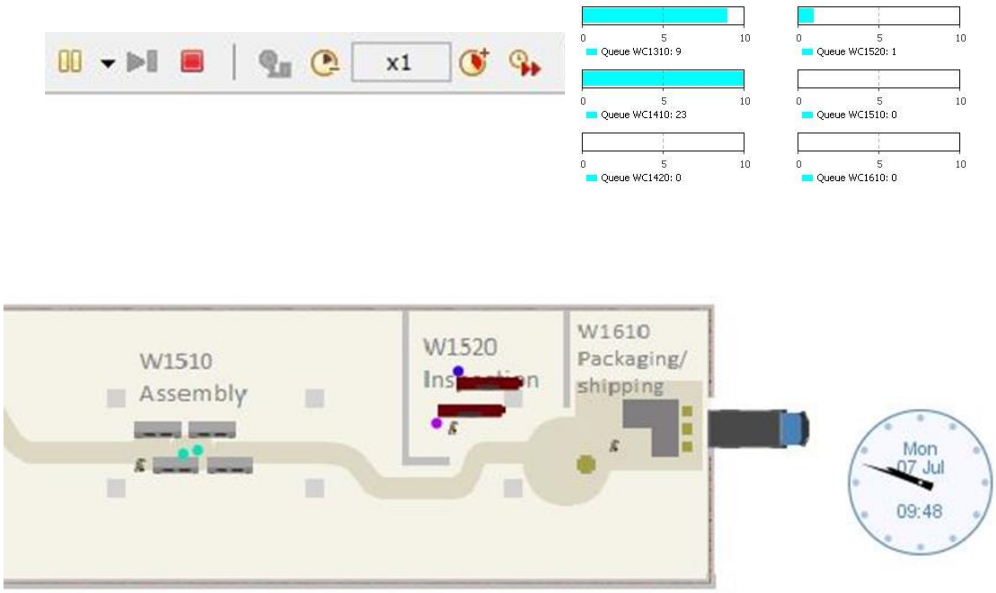


Figure 6-3 Partial screenshot of 2D visualisation

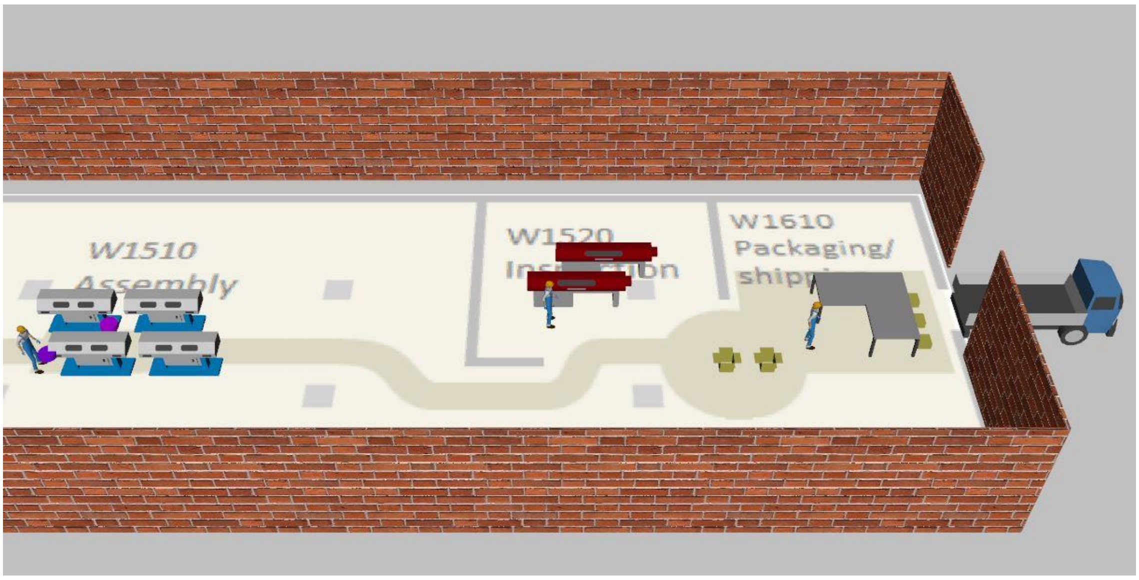
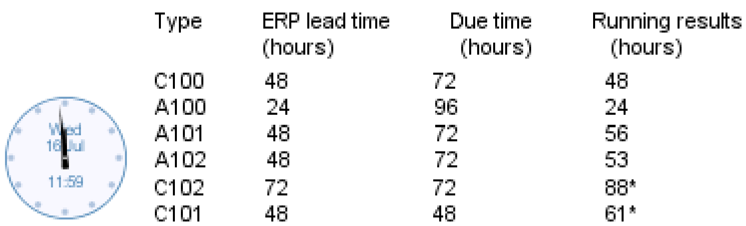


Figure 6-4 Partial screenshot of 3D visualisation



Type	ERP lead time (hours)	Due time (hours)	Running results (hours)
C100	48	72	48
A100	24	96	24
A101	48	72	56
A102	48	72	53
C102	72	72	88*
C101	48	48	61*

Figure 6-5 Screenshot of the prediction data in the simulation model

Table 6-4 Simulation results are outputted to the spreadsheet

Type	ERP lead time		Due date		Simulation results	
Type	Hours	Days	Hours	Days		
C100	48	2	72	3	48	On Time
A100	24	1	96	4	24	On Time
A101	48	2	72	3	56	On Time
A102	48	2	72	3	53	On Time
C102	72	3	72	4	88	Delay!
C101	48	2	48	3	61	Delay!

- Acquire Best Sequence

The second experiment is to validate that the symbiotic simulation system can calculate the best sequence of the given jobs. The trigger condition is a sale operator inputting an order data in the SAP R/3 system which is reflected in the

spreadsheets. As shown in the previous section, if the symbiotic simulation system is just used as visualisation and prediction tools, there will be two tardy jobs. In this experimental scenario, OptO is requested to find the best sequence of jobs in order to decrease the tardy jobs. Objective function, constraints and requirements have been defined to meet the objective. All the possible sequences are stored in a collection. Figure 6-6 shows the optimisation result which recommends using the number 589 sequence in the collection. When using the optimised sequence to initialise the simulation model, all the jobs can meet their due date as shown in Figure 6-7. Meanwhile, the simulation results are outputted into the SAY R/3 system as shown in table 6-5. The Symbiotic simulation system will continuously monitor the manufacturing state and output an estimated finish time of each product. Output data are generated dynamically according to the simulation model. In this way, the symbiotic simulation system gives a suggested sequence of jobs to the manufacturing manager.

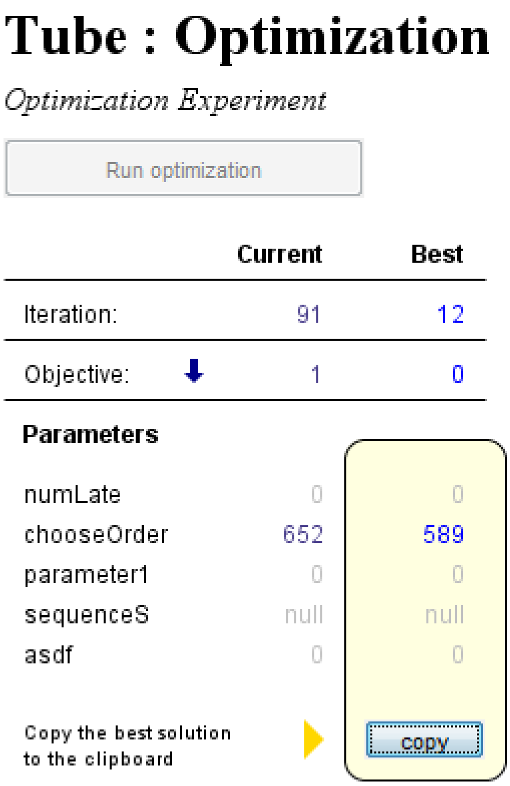


Figure 6-6 Screenshot of the optimisation result

Type	ERP lead time (hours)	Due time (hours)	Running results (hours)
C102	72	72	72
C101	48	48	48
C100	48	72	64
A102	48	72	56
A100	24	96	48
A101	48	72	72

Figure 6-7 Screenshot of the simulation result with optimised job sequence

Table 6-5 Simulation results are outputted to the spreadsheet

Type	ERP lead time		Due date		Simulation results	
Type	Hours	Days	Hours	Days		
C102	72	3	72	4	72	On Time
C101	48	2	48	3	48	On Time
C100	48	2	72	3	64	On Time
A102	48	2	72	3	56	On Time
A101	48	2	72	3	48	On Time
A100	24	1	96	4	72	On Time

In conclusion, two sets of experiments have been carried out to validate the functions of the symbiotic simulation system. The simulation results show that the symbiotic simulation system can respond in real time, periodically update the simulation results, tackle tardiness issues, suggest solutions and control the SAP R/3 system directly. The showcase indicates that, based on the provided generic prototype, a symbiotic simulation system can be generated. Additionally, all the objects and subsystems of the symbiotic simulation system can work effectively to achieve pre-defined purposes.

7 Discussion and Conclusions

This research focuses on developing a generic framework for integrating ERP systems to symbiotic simulation systems. In these symbiotic simulation systems, ERP systems and simulation models can be mutually beneficial to each other. On the one hand, ERP systems as the database of main manufacturing data can support simulation models. On the other hand, simulation models can generate better simulation results to influence or control ERP systems and physical systems. A case example has been developed to validate that the generic framework can be implemented in practical environments. Furthermore, experiments have been carried out to validate that the case example can work successfully to achieve certain purposes.

In this chapter, first of all, findings are reviewed by comparing with research objectives addressed in chapter 1. Secondly, contributions to knowledge are concluded and stated. Finally, limitation and future work are identified and described in detail.

7.1 Findings Compared with Objectives

Compared with research objectives addressed in chapter 1, four findings have been achieved as set out below:

- Objective 1: Developing a generic framework for integrating ERP systems to symbiotic simulation systems

The mathematical concept of the generic framework has been presented first to show the overall processes. Basic functions and common activities between ERP systems and simulation models have been analysed and summarised. After that, a generic framework for integrating ERP systems to symbiotic simulation systems has been developed with the consideration of applicability, extensibility, and scalability. The generic framework consists of a SSFS subsystem, a SSADS subsystem, a SSDSS subsystem, a SSCS subsystem, and various objects. It can achieve the pre-defined functions in manufacturing environments, such as visualisation, prediction, anomaly detection, suggesting decision parameters and controlling ERP systems directly. The generic

framework can be used as guidance for developers to build their own ERP-based symbiotic simulation systems.

- Objective 2: Demonstrating how the framework can be practically implemented

A symbiotic simulation system has been developed as a case example using Anylogic 6 and SAP R/3. The case example was based on the proposed generic framework and served as a showcase to demonstrate that the generic framework can be used as guidance for developing an ERP-base symbiotic simulation system. In addition, the symbiotic simulation system was developed using a real tube manufacturing shop floor and has the ability to tackle practical issues in manufacturing environments.

- Objective 3: Validating the functionalities of the symbiotic simulation system

Experiments have been carried out to validate the functions of the case example. Experiment results show that the symbiotic simulation system has the ability to visualise, predict, influence and control the ERP system and the tube manufacturing shop floor. For the tube manufacturing, it is validated that the symbiotic simulation system is applicable in dealing with manufacturing uncertainty, such as reducing lead time and tardy jobs. In this symbiotic simulation system, on the one hand, the simulation model acquires relevant manufacturing data from SAP R/3, such as raw material delivery information and customer orders. On the other hand, the simulation model controls the SAP R/3 system by revising parameters such as estimated due date and probability parameters of job distribution conveyors. The showcase indicates that all the objects and subsystems can work effectively to achieve pre-defined purposes.

7.2 Contributions to Knowledge

The primary contribution of this research is the generic framework for integrating ERP systems to symbiotic simulation systems. In the past, the applications of symbiotic simulation systems were focused on engineering areas such as transportation systems, military communication networks, air traffic controllers and multi-agent systems (Fujimoto et al., 2002; Aydt et al.,

2008a; Aydt et al., 2011; Kamrani and Ayani, 2007). This research enables future applications of symbiotic simulation beyond the engineering applications. The framework enables symbiotic simulation systems to be extended to IT and Information Management systems, including ERP systems. This will particularly benefit manufacturing organisations which have already used both simulation tools and ERP systems.

Secondly, the research also improves the current methods of linking ERP systems and simulation tools through analysing and summarising the interactions between them, for instance, those that were proposed by Moon and Phatak (2005). The research addresses the drawbacks of their system that did not take into account the automated optimisation and controlling feedback. Figure 7-1 shows they only use one SSFS to predict the lead time. In order to get satisfactory results, human intervention is required. In reality, the manufacturing environment may be very complicated, thus it will be extremely time consuming in dealing with practical issues. Improvements can be addressed followed by the proposed framework. For example, OptO, SSDSS, and SSCS can be added to automatically acquire the best overtime data and control the ERP system directly. The research addresses a better way to enhance the functionalities of ERP systems by using symbiotic simulation.

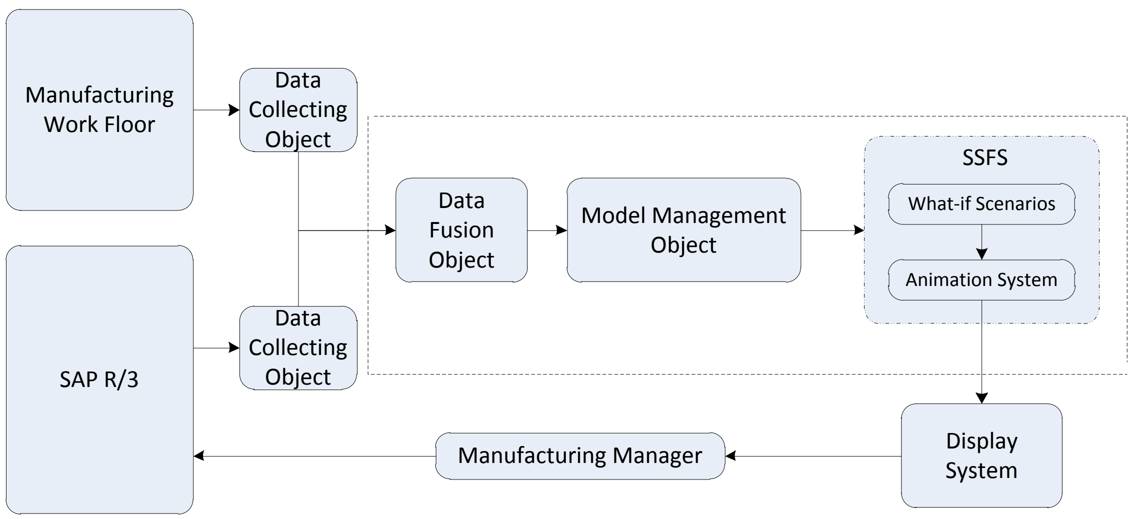


Figure 7-1 Structure of Moon and Phatak's case by using the proposed framework

The third contribution of the research is related to the applicability of the commercial off-the-shelf (COTS) simulation package. An important issue in implementing a symbiotic simulation is the development of an interface between the existing simulation package and symbiotic simulation systems. This research has extended the practicality of existing theories, e.g. those that are proposed by Aydt et al. (2009a), and has demonstrated how the symbiotic simulation can be built using a COTS package. In this research, Anylogic 6 was used to practically deploy the framework into a fully working symbiotic simulation system. The system includes a simulation model, a user interface the real-time simulation engine, the what-if scenarios optimisation engine and the control functions. The techniques and methods devised in this research can be used as a reference for simulation developers to integrate existing simulation packages into a symbiotic simulation system.

7.3 Limitations and Future Work

This research contains some limitations in a few areas. Therefore, some enhancements can be conducted to make current research work better. Limitations and some future work are summarised and illustrated below.

The related work on symbiotic simulation and ERP systems has been reviewed to address the research gap. It is therefore recommended to try different types of manufacturing problems which can potentially be resolved by using this generic framework.

Due to confidentiality restriction, the case example used in this research does not fully represent the tube manufacturing shop floor. Furthermore, the data inside the SAP R/3 system and the external factors involved are in fact far more complex than that in this research. Some assumptions had also need to be made when developing the simulation model. Further work should, whenever possible, focus on adding more real data so as to make the system more realistic and credible.

In order to extend the usage of the generic framework, more ERP-based industries should be investigated and the generic framework should be improved to suit more application areas. For instance, an ERP-based retail industry can develop a symbiotic simulation system to guide their investment. By accessing data about market, products, and customers in ERP systems, the symbiotic simulation can help the company to continuously analyse future investments.

REFERENCES

Abu-Taieh, E. M. O. and El-Sheikh, A. (2007), "Commercial simulation packages: a comparative study", *International Journal of Simulation,* vol. 8, no. 2, pp. 66-76.

Addo-Tenkorang, R. and Helo, P. (2011), "Enterprise resource planning (ERP): A review literature report", *Proceedings of the World Congress on Engineering and Computer Science,* Vol. 2, 19-21 Oct. 2011, San Francisco, pp. 1-9.

Anderson, G. (2011), *Sams Teach Yourself SAP in 24 Hours,* 4th ed, Mark Taub, Indianapolis, Indiana.

Anderson, G. and Larocca, D. (2005), *Sams teach yourself SAP in 24 hours,* 2nd ed, Paul Boger, Indianapolis, Indiana.

Anylogic (2014)*, Anylogic company website*, available at: http://www.anylogic.com/ (accessed 3, July).

April, J., Glover, F., Kelly, J. P. and Laguna, M. (2003), "Practical introduction to simulation optimization", Chick, S. E., Sanchez, P. J., Ferrin, D., et al (eds.), in: *Simulation Conference, 2003. Proceedings of the 2003 Winter,* Vol. 1, 7-10 December 2003, New Orleans, LA, IEEE, pp. 71-78.

Aydt, H., Turner, S. J., Cai, W. and Gan, B. P. (2011), "Symbiotic simulation for optimisation of tool operations in semiconductor manufacturing", Jain, S., Creasey, R., Himmelspach, J., et al (eds.), in: *Proceedings of the 2011 Winter Simulation Conference,* 11-14 December 2011, Winter Simulation Conference, pp. 2093-2104.

Aydt, H., Turner, S. J., Cai, W. and Low, M. Y. H. (2009a), "An agent-based generic framework for symbiotic simulation systems", *Multi-Agent Systems,* vol. 3, pp. 357-387.

Aydt, H., Turner, S. J., Cai, W. and Low, M. Y. H. (2009b), "Research issues in symbiotic simulation", Rossetti, M. D., Hill, R. R., Johansson, B., et al (eds.), in: *Proceedings of the 2009 Winter Simulation Conference,* Winter Simulation Conference, pp. 1213-1222.

Aydt, H., Turner, S. J., Cai, W., Low, M. Y. H., Lendermann, P. and Gan, B. P. (2008a), "Symbiotic simulation control in semiconductor manufacturing", in Bubak, M., Van Albada, G. D., Dongarra, J., et al (eds.) *Computational Science–ICCS 2008,* Springer, Berlin Heidelberg, pp. 26-35.

Aydt, H., Turner, S. J., Cai, W. and Low, M. Y. (2008b), "Symbiotic simulation systems: An extended definition motivated by symbiosis in Biology", *22nd*

Workshop on Principles of Advanced and Distributed Simulation, 3-6 June 2008, Roma, IEEE, pp. 109-116.

Babulak, E. and Wang, M. (2008), "Discrete event simulation", *Discrete Event Simulation: State of the Art,* vol. 1, pp. 1-10.

Banks, J. (1999), "Introduction to simulation", Farrington, P. A., Nembhard, H. B., Sturrock, D. T., et al (eds.), in: *Proceedings of the 31st conference on Winter simulation: Simulation---a bridge to the future,* Vol. 1, ACM, New York, pp. 7-13.

Battista, C., Dello Stritto, G., Iannone, R. and Schiraldi, M. M. (2011), "Integrating MRP in production systems simulation tools", *Proceedings of the Conference" breaking down the barriers between research and industry",* 14-14 September 2011, Abano Terme, Padova, IT, pp. 1-6.

Carson, Y. and Maria, A. (1997), "Simulation optimization: methods and applications", Andradottir, S., Healy, K. J., Withers, D. H., et al (eds.), in: *Proceedings of the 29th conference on Winter simulation,* IEEE Computer Society, Washington, DC, pp. 118-126.

Davenport, T. H. (1998), "Putting the enterprise into the enterprise system", *Harvard business review,* vol. 76, no. 4, pp. 1-12.

Fanchao, Z., Turner, S. J. and Aydt, H. (2009), "Symbiotic simulation control in supply chain of lubricant additive industry", *Proceedings of the 2009 13th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications,* IEEE Computer Society, Washington, DC, pp. 165-172.

Fujimoto, R., Lunceford, D. and Page, E and A. M. U (editors) (2002), *Grand challenges for modeling and simulation: Dagstuhl report.* Technical report 350.

Ho, C. and Ireland, T. C. (2012), "Mitigating forecast errors by lot-sizing rules in ERP-controlled manufacturing systems", *International Journal of Production Research,* vol. 50, no. 11, pp. 3080-3094.

Ignatiadis, I. and Nandhakumar, J. (2009), "The effect of ERP system workarounds on organizational control: An interpretivist case study", *Scandinavian Journal of Information Systems,* vol. 21, no. 2, pp. 1-33.

Jänicke, W. (2001), "Excel planning interface for the SAP R/3 system", *Chemical Engineering & Technology,* vol. 24, no. 9, pp. 903-908.

Jovanoski, B., Minovski, R. N., Lichtenegger, G. and Voessner, S. (2013), "Managing strategy and production through hybrid simulation", *Industrial Management & Data Systems,* vol. 113, no. 8, pp. 1110-1132.

Kamrani, F. and Ayani, R. (2007), "Using on-line simulation for adaptive path planning of UAVs", *Proceedings of the 11th IEEE International Symposium on Distributed Simulation and Real-Time Applications,* IEEE Computer Society, Washington, DC, pp. 167-174.

Koh, S. L. and Gunasekaran, A. (2007), "The Effects of Uncertainty on ERP-Controlled Manufacturing Supply Chains", *Modeling and Analysis of Enterprise Information Systems,* vol. 1, pp. 63-86.

Law, A. M. and McComas, M. G. (1998), "Simulation of manufacturing systems", Thesen, A., Grant, H. and Kelton, W. D. (eds.), in: *Proceedings of the 30th conference on Winter simulation,* ACM, New York, pp. 631-643.

Li, D. (2011), "Enterprise systems: state-of-the-art and future trends", *Industrial Informatics, IEEE Transactions on,* vol. 7, no. 4, pp. 630-640.

Low, M. Y. H., Lye, K. W., Lendermann, P., Turner, S. J., Chim, R. T. W. and Leo, S. H. (2005), "An agent-based approach for managing symbiotic simulation of semiconductor assembly and test operation", *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems,* ACM, New York, pp. 85-92.

Manivannan, S. and Banks, J. (1992), "Design of a knowledge-based on-line simulation system to control a manufacturing shop floor", *IIE transactions,* vol. 24, no. 3, pp. 72-83.

Mazzullo, J. (2006), *SAP R/3 for everyone : step-by-step instructions, practical advice, and other tips and tricks for working with SAP,* Prentice Hall Professional Technical Reference, London.

Moon, Y. B. and Phatak, D. (2005), "Enhancing ERP system's functionality with discrete event simulation", *Industrial Management & Data Systems,* vol. 105, no. 9, pp. 1206-1224.

O'Leary, D. E. (2000), *Enterprise resource planning systems : systems, life cycle, electronic commerce, and risk,* Cambridge University Press, Cambridge.

OptTek (2014), *OptTek company website*, available at: http://www.opttek.com/ (accessed 4, July).

Rashid, M. A., Hossain, L. and Patrick, J. D. (2002), "The evolution of ERP Systems: A historical perspective", *Enterprise Resource Planning: Global opportunities & challenges,* , pp. 1-16.

Robertson, N. and Perera, T. (2002), "Automated data collection for simulation?", *Simulation Practice and Theory,* vol. 9, no. 6, pp. 349-364.

Shaul, L. and Tauber, D. (2013), "Critical success factors in enterprise resource planning systems: Review of the last decade", *ACM Computing Surveys (CSUR),* vol. 45, no. 4, pp. 1-39.

Skoogh, A., Johansson, B. and Stahre, J. (2012a), "Automated input data management: evaluation of a concept for reduced time consumption in discrete event simulation", *Simulation,* , pp. 1280-1294.

Skoogh, A., Perera, T. and Johansson, B. (2012b), "Input data management in simulation – Industrial practices and future trends", *Simulation Modelling Practice and Theory,* vol. 29, pp. 181-192.

Winshuttle (2014)*, Winshuttle company website*, available at: http://www.winshuttle.com (accessed 23, June).