The Pennsylvania State University

The Graduate School

ANALYTICAL AND EMPIRICAL MODELS FOR PROCESS TRANSFORMATION TOWARDS SMARTER MAINTENANCE

A Dissertation in

Industrial Engineering

by

Hyun Jong Shin

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2021

The dissertation of Hyun Jong Shin was reviewed and approved by the following:

Vittaldas V Prabhu
Professor in Industrial and Manufacturing Engineering and Charles and Enid Schneider Faculty Chair in Service Enterprise Engineering
Dissertation Advisor
Chair of Committee

Ling Rothrock
Professor in Industrial and Manufacturing Engineering

Paul Marshal Griffin
Professor in Industrial and Manufacturing Engineering

Suvrat Saurabh Dhanorkar
Assistant Professor in Supply Chain and Information Systems

Felisa Preciado Higgins
Associate Clinical Professor in Supply Chain and Information Systems and Associate Dean for Undergraduate Education in the Smeal College of Business
Special Member

Steven James Landry
Professor in Industrial and Manufacturing Engineering
Head of the Harold and Inge Marcus Department of Industrial and Manufacturing Engineering

ABSTRACT

It is widely believed that the world's economies are on the cusp of a mega-trend in automation in terms of physical and cognitive tasks. This dissertation investigates analytical and empirical models for process transformation especially focused on maintenance. We have elected to focus on maintenance because associated processes represent significant opportunities in terms of productivity improvement. Furthermore, maintenance could be among the most challenging processes for automation because it relies heavily on human cognition and decision-making. This dissertation has the following three major parts:

1. An empirical study in a controlled laboratory setting to compare diagnosis tasks in maintenance using traditional fault-tree and a new artificial intelligence supported system.
2. Modeling maintenance time using negative hypergeometric distribution considering the experience effect of technicians and evaluation of the variability in the number of diagnosis attempts depending on diagnosis support systems
3. Modeling stochastic dynamics in business processes using Markov Chains with a focus on tacit processes, making it an attractive approach for investigating and reengineering maintenance processes

In the first part of the dissertation, the effect of diagnosis support systems on the performance of a technician during maintenance activity is investigated by conducting a set of controlled lab experiments. These experiments reveal that human subjects took different amounts of time to complete a task depending on whether a fault-tree diagnosis support system or an artificial intelligence diagnosis support system was provided. However, there was no statistical difference in the workload perceived.

The second part of the dissertation consists of modeling maintenance activities considering the experience effect of technicians since the diagnosis support system may support inexperienced technicians. The study reveals that the diagnosis support systems probably lessen the variability in the number of diagnosis attempts.

The third part of the dissertation consists of modeling the stochastic dynamics of business processes due to the variability introduced by informal, tacit processes. The different types of tacit processes depending on the interaction types are defined. Then, the modeling method using the discrete-time Markov chain is introduced. In the case study, the flow time difference between the actual estimation and the model prediction was only 3%. In addition, the UPS maintenance process reengineering options are evaluated using the proposed method.

Although the domain of this study is rooted in maintenance, the method can be expanded to other sectors and domains.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

But let the one who boasts boast about this: that they have the understanding to know me, that I am the Lord, who exercises kindness, justice and righteousness on earth, for in these I delight," declares the Lord (Jeremiah 9:24).

Special thanks to my incredible advisor and mentor, Dr. Prabhu, Vittaldas and my committee: Dr. Dhanorkar, Suvrat Saurabh, Dr. Griffin, Paul, Dr. Higgins, Felisa Preciado and Dr. Rothrock Ling.

Also, this journey would not be successfully ended without support from my friends. Especially thanks to Jonathan Binder, YKKPC and DISCRETE Lab members: Amol, Chintan, Kai-wen, Rakshith, Toyosi, and Shiyu. In addition, I really appreciate those who supports me by prayers and encouragement.

Lastly and most importantly, I am especially thanks to my wife, Yumi who were always with me throughout this journey. I also thanks to my two lovely children, Chaeeun(Chloe) and Theo (Theodore) who made to me smile. I deeply appreciate the sacrifice and support of my parents: Seungyoul Shin & Kyungim Lee and Myungjin Kim & Hyeonok Park. Also I am grateful to be brother of you: Hyelim, Kangil(Kevin) and Wonyoung.

It has been a long journey for a business undergraduate student to earn a Ph.D. in Industrial Engineering. It would not be possible without the grace of God and the support of people around me.

Once again, thanks for being with me

Chapter 1

Introduction

It is widely believed that technological innovation is at the center of the mega trend of the global economy. The rapid advancement of technology brings about additional opportunities to automate physical and cognitive tasks and transform processes [1], [2]. The manufacturing industry, for instance, is considered one of the most automated industries but there are significant possibilities for further automation. A report states that there is a large amount of automation potential in the manufacturing industry [2], [3]. Maintenance is an area where great improvements could be made in automation. Currently, only a few maintenance tasks are fully automated and these tasks are composed of technical and non-technical tasks [4]. Retrieving instructions or information from manuals, for instance, consumes about 45% of maintenance technicians' time. In addition, the cognitive time, which refers to the time that technicians are not engaged with a device or instrument, accounts for almost half of total maintenance time [5]. However, fully-automated maintenance processes are impractical considering technical and financial feasibility. Semi-automated processes, instead, are practical where interactions between human and non-human operators are inevitable.

In this kind of environment, the process is designed in a way such that technology enhances the performance of human operators [1], [6]–[8]. Since the impact of technology differs case by case, any technology must be evaluated before implementation. In addition, the impact of any technology on workflow needs to be evaluated since tasks are connected and influence one another.

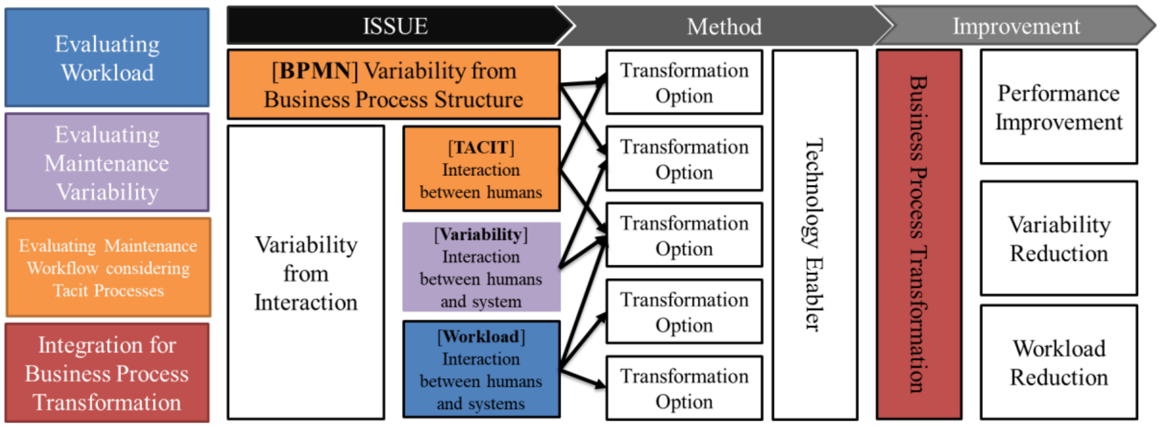


Figure **1-1**: Dissertation Concept Diagram

Therefore, this dissertation investigates semi-automated business process transformation especially focused on maintenance as shown in Figure **1-1**. The dissertation examines impact on performance, variability and workload on a maintenance task and maintenance process. Then, it proposes a framework for business process reengineering as shown in Figure **1-2**.

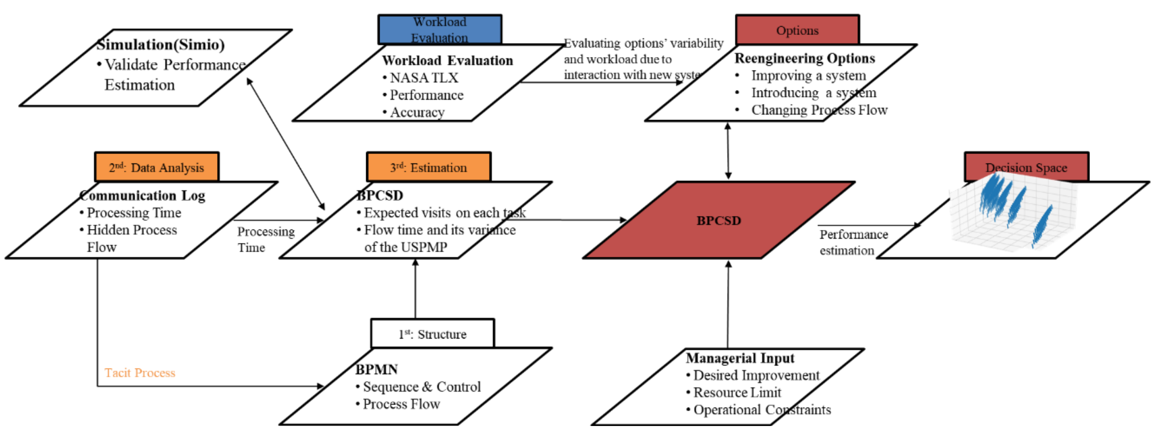


Figure **1-2**: Framework for Business Process Transformation

The rest of this dissertation is structured as follows; the effect of different diagnosis support systems which are meant to lessen the workload of technicians during maintenance is evaluated in Chapter 2. Machine learning applications in the field of manufacturing and the impact of a system interface on users are reviewed. Then, the design and results of the experiment with 45 participants are explained.

In Chapter 3, business process variability (especially due to human interactions) and a way to model this workflow variability are studied. Various models for business process modeling and metrics are reviewed. Then, the informal processes, tacit processes are defined with examples. The modeling method, business process complexity with stochastic dynamic, is proposed. Two business process cases are analyzed using the proposed method.

In Chapter 4, a framework for business process reengineering is proposed. The case presented in Chapter 3 is used as an example. The framework integrates models to propose options for decision-making.

Lastly, in Chapter 5, the findings of this dissertation are summarized. Then, future studies for this domain are suggested considering the limitations of this dissertation

Chapter 2

Impact of Diagnosis Support System on Technicians During the Maintenance

Advancement of digital technologies enables smart and automated processes. Businesses introduce new technologies to achieve competitive advantages over their competitors by changing tasks, processes, and the ways we work [9]. Maintenance is an especially promising area where adopting new digital technologies can bring great productivity improvements because currently only a few maintenance activities are fully automated [5], [10], [11]. For many maintenance activities, a large portion of maintenance time is used for diagnosis which is similar to trial-and-error processes. This requires repetitive activities until a technician finds the cause of machine failure [12]. Thus, utilizing a diagnosis support system can improve maintenance productivity. For practical implementation, studies in various perspectives are needed [13], especially in the domain of effects on users of such systems [9], [13], but few studies have been done to analyze the effect of the different diagnosis support systems on technicians. Therefore, we investigated the effect of two types of diagnosis support systems on the workload of technicians during maintenance: FTd-system and AId-system.

The rest of this chapter is focused on modeling corrective maintenance time considering the experience effect on the number of diagnosis attempts. The structure of this chapter is as follows. In Section 1, we review artificial intelligence (AI) applications in the domain of maintenance. Also, the effects of the system interface on users are reviewed. Next, in Section 2, we present the experiment which aims to compare the effects of two different diagnosis support systems. In Section 3, we discuss our findings using natural goals, operators, methods, selection rules language (NGOMSL). Lastly, in Section 4, we conclude with directions for future research.

2.1 Literature Review

Traditionally, diagnosis tasks are aided by a fault-tree analysis that systematically localizes the cause of a failure mode. With the advancement in digital technologies, an alternative approach, the AI method, is discussed [14]–[17]. Industrial internet of things, cloud technology, and information and communication systems are enablers of such a system [10].

2.1.1 Machine Learning and Artificial Intelligence in Maintenance

Many AI application studies focus on finding the root cause of production defects using different methodologies such as the implementation of a Bayesian network, artificial neural network, or support vector machine [14], [18]. The industries studied include the semiconductor and the automobile industries [16], [19], [20]. The other AI application usage in the literature is a method to predict the performance of the process by finding errors in the processing stage [21], [22]. Besides these usages, AI is also utilized to find quality issues during or after production. For example, the surface roughness of the processed materials is predicted using an AI approach [23], [24].

Lastly, AI is also used to cope with a machine failure mode in different situations. It can be used for the corrective maintenance. For example, detecting vehicle faults is one common application [17]. Another usage is to detect the degradation of machines and tools or detect the fault of a machine [18], [25]–[27]. However, AI is more popularly used for preventive or condition-. In these situations, AI method is used to plan an effective maintenance schedule by predicting the condition of a system based on real-time data or previous collected data. For example, the healthy state of a component in a wind turbine was predicted based on the vibration data [28]. Similarly, the healthy state of engine was also predicted based on the vibration data

[29]. In addition to the vibration data, the data, such as temperature, collected from 21 sensors were used to predict the current condition of an engine [30]. Instead of focusing on the healthy state of a component, AI is also used to capture and classify the failure signature with potential failure occurrence [31].

In sum, most of current studies of AI in maintenance focus on applying new machine learning methodologies in different domains rather than its effect on the technician workload.

2.1.2 Human Centered System Interaction

The increased use of digital technology in manufacturing is meant to improve operators' performance. However, interacting with an inappropriately designed system may act as a barrier and creates even more unidentified challenges [32] because many focus on the technology and fail to consider the contribution and role of humans [33]. Thus, the importance of human-centered design is emphasized [32]–[35]. For example, Pacaux-Lemoine addressed the importance of designing a system based on human-machine cooperation principles and tested the workload of the different systems using NASA-TLX [6]. Cimini stated that the development of new smart technologies should be together with the developments in the human-related aspects [36].

Depending on the interface and methodology behind a diagnosis support system, technician workload may differ. As more information is reposited and transmitted electronically, the workload difference due to the presentation of media has been studied by various methods [37]–[39]. Besides studying the effect of media presentation, other researchers are focused on the effect of the interface on performance and workload in selected domains. For example, the effect of the method of interaction with a mobile application on performance and workload has been studied [40]. Another experiment studied the influence of user interface on performance and situational awareness under a controlled lab environment [41]. Some have studied operator workload in highly stressful environments such as nuclear power plants and shipping ports using NASA-TLX and the eye-tracking method [42]–[44].

Therefore, to develop the smart diagnosis support system which aids technicians’ task, the study of its effectiveness during the maintenance task and its impact on technicians in various measures is desired.

2.2 Experiment

There are many studies that explore the role of AI applications in fault detection and workload difference due to different interfaces. However, limited studies are available in the domain of maintenance. Therefore, we conducted a controlled lab environment to study the effect of AI usage during maintenance from various perspectives since working with a real industry machine in a real environment is expensive and involves many environmental variables that cannot be controlled.

2.2.1 Experiment Design and Setup

In this study, we prepared an electric circuit model with a proximity sensor. We prepared this model because a proximity sensor, which is used to detect the presence of an object, is widely used in many automated machines such as computer numerical control (CNC) machines and has a high failure rate. Failure to maintain this small part can cause the machine to stop functioning. However, maintaining this sensor is not as simple as replacing it with a new one. A technician is required to check not only the sensor itself but also all other related components connected to the sensor such as cables, the power supply, and the input/output (I/O) board. Therefore, in this experiment, we examine the effect of diagnosis support systems on maintaining an electric circuit model with a proximity sensor.

Experiment Setting

The electric circuit model for this experiment is composed of one sensor, one battery, one switch, five cables, and one light bulb. Every component in the model represents some component in a real industry machine as shown in Table **2-1**. The proximity sensor in the model detects whether or not the door is closed and shuts off the power depending on the position of the door.

Table **2-1**: Proximity Sensor Model Setting

Real Machine Setting	Model Setting	Quantity in the Model	Broken
Sensor	Sensor	1	No
Power	Battery	1	Yes
Power Switch	Switch	1	Yes
Sensor Connection	Signal Cable to Light Bulb	1	Yes
Cables	Cables	4	No
I/O Board	Light bulb	1	Yes

In the experiment, four components are purposely broken: (1) battery, (2) switch, (3) light bulb, and (4) signal cable to light bulb (black cable). Participants diagnose and repair the model by interacting with either a fault tree diagnosis support system (FTd-system) or an artificial intelligence diagnosis support system (AId-system).

Diagnosis Support Systems

FTd-systems are commonly used in repairing machines. The system supports participants by identifying the locations of problems using the deductive failure analysis method. The approach is logical, but the common cause of failure is not obviously identified.

On the other hand, AId-systems help participants by suggesting the locations of problems based on pre-calculated probability. In this approach, big data need to be obtained and analyzed using supervised machine learning algorithms in advance. The advantage of this approach is that it is able to identify a common cause of failure first and also eliminate options or update the failure probabilities as the condition of components is updated by users or sensors.

In practice, the probability can be calculated based on the past maintenance log and data collected from IoT sensors. For example, we can calculate the probability that Component A's condition not good as shown in Table **2-3** based on the data which contains the condition of each component at the time of the failure as shown in Table **2-2**.

Table **2-2**: Maintenance Log Example

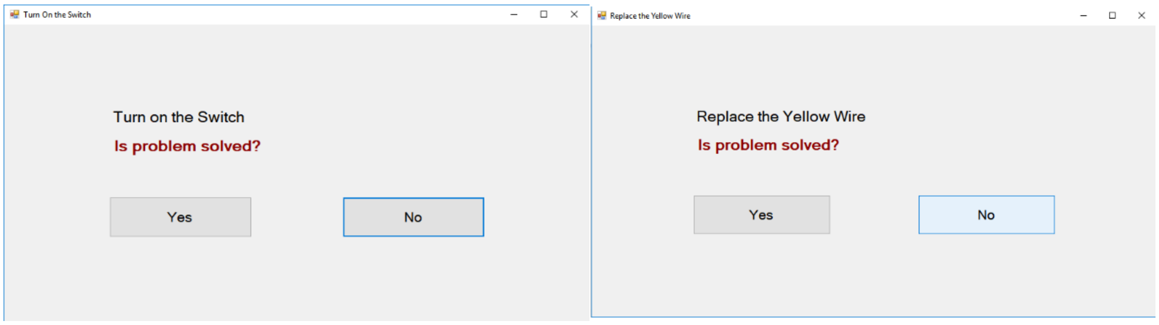
Log #	System	Sensor	Switch	Power	LB	YW	RW	GW	WW	BW
1	f	g	b	g	g	g	g	g	g	g
2	f	g	b	g	g	g	g	g	g	g
3	f	g	g	b	g	g	g	g	g	g
					…					
n	f	g	g	b	b	g	g	g	g	b

f = Fail, g = in good condition, and b = in bad condition

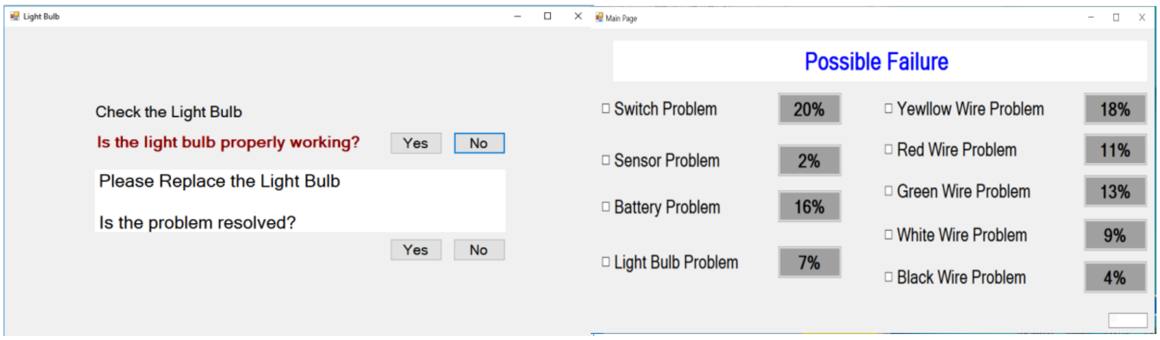
Table **2-3**: Cause of Failure Probability Calculation Example

Description of Conditions	Count	Probability
P(switch=b |system=f and others =g)	10	14%
P(sensor=b |system=f and others =g)	2	3%
P(Power=b |system=f and others =g)	9	13%
P(LB=b |system=f and others =g)	8	11%
P(YW=b |system=f and others =g)	3	4%
P(RW=b |system=f and others =g)	2	3%
P(GW=b |system=f and others =g)	4	6%
P(WW=b |system=f and others =g)	3	4%
P(BW=b |system=f and others =g)	7	10%
P(more than 2 components = b |system=f)	22	31%

In this experiment, the probability given by the AId-system is arbitrarily assigned for the experimental purpose rather than a probability calculated based on a real maintenance log. The interface of two systems is as shown in Figure **2-1**.



(a) FTd-System Interface



(b) AId-System Interface

Figure **2-1**: Diagnosis Support System Interface

Experiment Measure

To analyze the impact of the diagnosis support system on technicians during maintenance, five measures are used: (1) time to complete, (2) NASA Task Load (NASA TLX), (3) diagnosis attempt, and (4) unnecessary replacement.

Time to complete is the time that a participant takes to diagnose components and replace them. This is comprised of diagnosis time, such as using a diagnosis support system and a multimeter, and the time to isolate and replace components.

NASA Task Load Index (TLX) is a subjective assessment tool that rates the perceived workload of participants. Participants rate their workload using the rating scales form at the end of the experiment. The rates are weighted by the source of the workload.

Maintenance attempt is the number of parts that are diagnosed by a participant. If the participant strictly follows the suggestion of the diagnose support systems and makes no errors, the number of attempts should be eight.

Unnecessary replacement is a count of components that are in good condition but were replaced by a participant during the maintenance.

Experiment Procedures

The experiment procedures can be described in four sessions: (1) screening & training, (2) preliminary study, (3) study, and (4) post-study.

The screening & training session begins upon the arrival of a participant. The requirements of the experiment are explained to the participant. If the participant meets all requirements of the experiment, a written informed consent form which has been confirmed by the Penn State University Institutional Review Board is given to the participant to review. A signature is not collected from the participant, but the participant expresses their implicit consent by participating in the study. Only after reviewing the informed consent form, demographic information such as age and gender of the participant is collected. Then the parts of the electric circuit with the proximity sensor used for the experiment and the ways to diagnose each part with a multimeter are explained to the participant. Furthermore, the support system that the participant will interact with during the study session is introduced. After the explanation, the participant has time to experiment with the electric circuit, the multimeter, and the support system as much as they want.

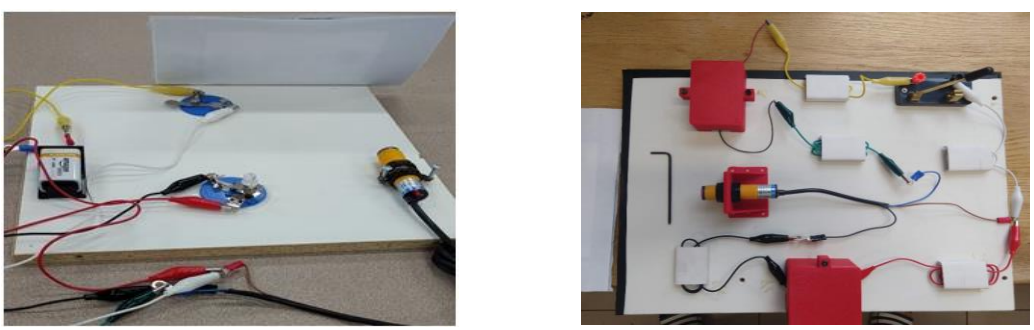
After the screening & training session, the preliminary study is conducted. In this session, the basic performance of the participant is measured. The participant is asked to remove the cover, isolate, or detach one part from the circuit model, diagnose the condition of the part, put it in the circuit, and put the cover back.

A break time is given to the participant after the preliminary study session if they desire. Then, the same circuit model but with broken parts and the support system is presented to the participant. The study session begins as the participant clicks “Start Diagnosis” in the support system. The study ends when the participant successfully completes maintenance of the model or attempts to diagnose all 9 components in the model.

Upon maintenance completion, the post-study session begins as the participant is asked to complete a NASA TLX survey. Then, a monetary award is given to the participant after signing the award receipt. Lastly, the participant has time to raise concerns or ask questions regarding the experiment and is dismissed if they do not have any questions.

Pilot Study

With 10 participants, the pilot study was conducted to explore any unexpected issues and to determine the sample size for the experiment. After the pilot study, the covers were added to the tops of each component to create an environment of similar complexity to the practical maintenance as shown in Figure **2-2**.



(a) Experiment setting before adding covers (b) Experiment setting after adding covers

Figure **2-2**: Experiment Setting Before and After Changes

Sample Size

With the mean and standard deviation collected from the preliminary study, the number of participants for the experiment was determined using power analysis. Since the standard deviation of the preliminary study was 114.83, the experiment required at least 9 participants for each group at two-tailed, 5%-level, and 80% power.

2.2.2 Experiment Result and Analysis

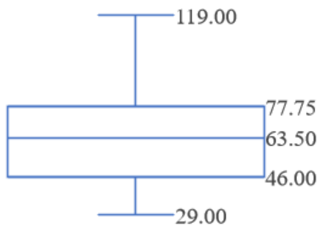
Thirty adults (mean age = 24.87 years old, range 19 – 38) participated in the study. All participants were recruited by posting advertisements in the Penn State University library, the Hertzel Union building, and classrooms. There were 20 male participants and 10 female participants. To investigate the difference in the basic performance due to the gender difference, a two-sample t- test is conducted. As shown in Table **2-4**, the difference in the basic performance due to the gender is not observed. The same number of female participants were, nevertheless, assigned to each group in order to minimize gender possible effects.

Table **2-4**: The Average Performance Time by Gender

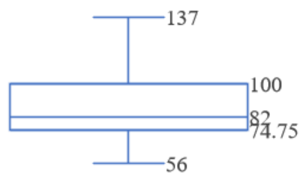
Task	Average Time		T-value	P-value
	Male	Female		
Sensor	62.8	73.9	1.24	0.23
Battery	81.0	95.8	1.78	0.10
Light Bulb	82.6	90.9	0.73	0.48
Wire	46.1	43.8	-0.48	0.64

The average times to diagnose the sensor, the battery, the light bulb, and the wire were 66 seconds, 86 seconds, 85 seconds, and 45 seconds, respectively as shown in Figure **2-3**.

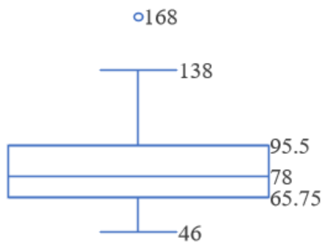
To see the effect of two different diagnosis support systems, each group's performance was compared by a two-tailed t-test.



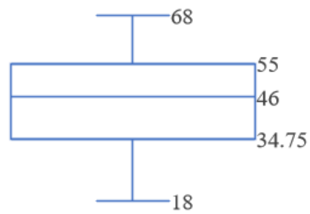
(a) Sensor Diagnose Time



(b) Battery Diagnose Time



(c) Light Bulb Diagnose Time

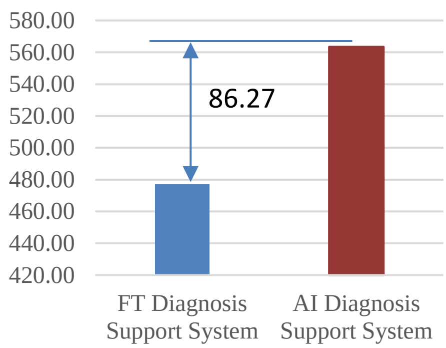


(D) Wire Diagnose Time

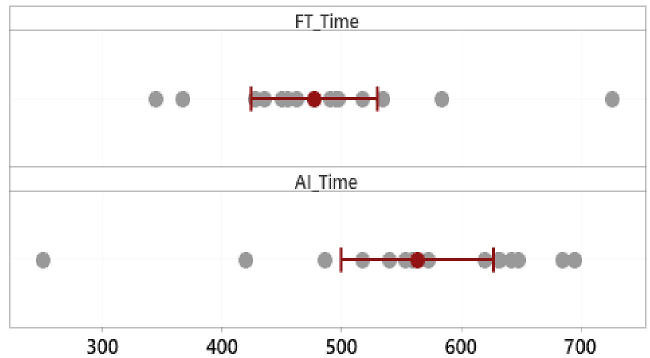
Figure **2-3:** Preliminary Study Result

Time to Complete

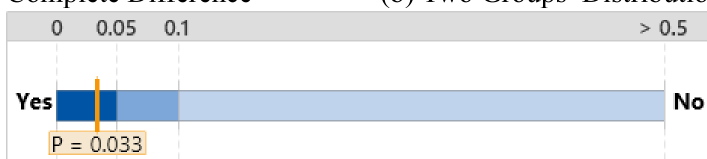
The mean and standard deviation of completion time for the group that used the FTd-system was 477.20 seconds and 94.96 whereas the mean and standard deviation of completion time for the other group was 563.47 seconds and 114.85. The coefficient of variation of the group that used the FTd-system was 19.90% and the other group was 20.38%. The group that used the FTd-system completed the task 86.27 seconds faster than the other group. The p-value for the two samples t-test is 0.03. Therefore, at α equals 0.05, the mean difference between the two groups is statistically significant as shown in Figure **2-4**.



(a) Mean Time to Complete Difference



(b) Two Groups' Distribution of Data



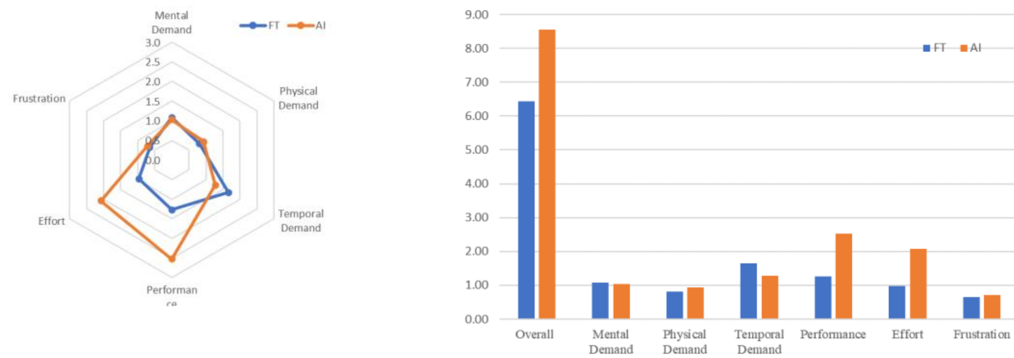
(c) Two Sample Mean t-test

Figure **2-4**: Mean Difference of Time to Complete

NASA TLX

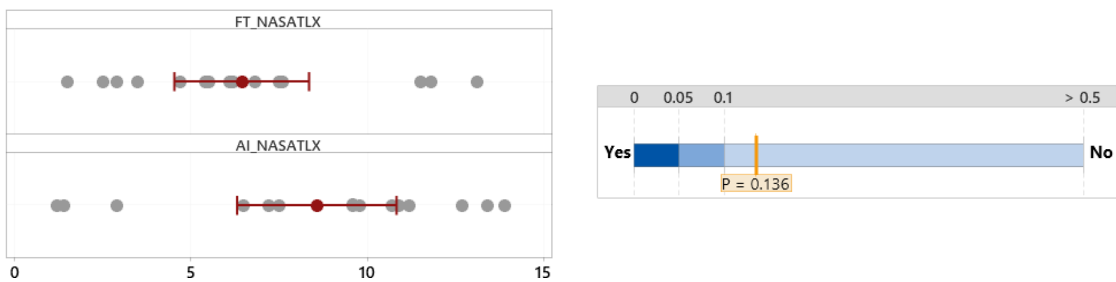
The FTd-system group's mean and standard deviation of the overall workload was 6.44 and 3.35. The two major sources of this workload were the 'temporal demand' and 'performance'. The other group's mean and standard deviation of the overall workload was 8.56 and 4.09. The two major sources of this workload were 'performance' and 'effort' as shown in Figure **2-5**.

The difference in the mean of the overall workload between the two groups was 2.12. More specifically, the AId-system group rates more loads on the effort and performance compared to the other group as shown in Figure **2-5**. The p-value for the two samples t-test is 0.14. Therefore, at α equals 0.05, there is no statistically-significant difference between the two groups' mean of overall workload as shown in Figure **2-6**.



(a) Source of Overall Workload (b) Workload Comparison

Figure **2-5**: NASA TLX



(a) Distribution of Data (b) Two Sample Mean t-test

Figure **2-6**: Mean Difference of NASA TLX

Maintenance Attempt

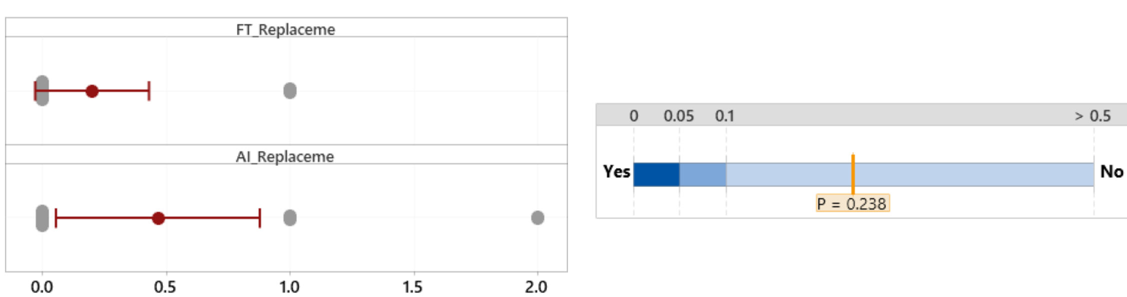
The number of maintenance attempts for both groups should be 8 if all participants strictly followed the suggestion of the diagnosis support system and made no errors. The mean of the FTd-system group's maintenance attempts was 8.20 with standard deviation 0.41. The mean of AId-system groups' maintenance attempt was 8.07 with standard deviation 1.16. The mean difference between the two groups was only 0.13. The p-value for the two samples t-test was 0.68. Therefore, at α equals 0.05, there was no statistically-significant difference between the two group means as shown in Figure **2-7**.

(a) Distribution of Data (b) Two Sample Mean t-test

Figure **2-7**: Mean Difference in Maintenance Attempt

Unnecessary Replacement

Unnecessary replacement can occur when a participant does not properly diagnose a component. One-fifth of the AId-system group participants replaced unnecessary components whereas only 1/3 of the FTd-system group participants replaced unnecessary components. The AId-system group's mean and standard deviation of unnecessary replacement is 0.20 and 0.41 respectively. The other group's mean and standard deviation was 0.47 and 0.74 respectively. The mean difference between the two groups was 0.27 and the p-value for the two samples t-test was 0.24. Therefore, at α equals 0.05, there was no statistically-significant difference between the two groups' means as shown in Figure **2-8**.



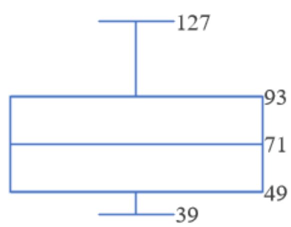
(a) Distribution of Data (b) Two Sample Mean t-test

Figure **2-8**: Mean Difference in the Number of Unnecessary Replacement

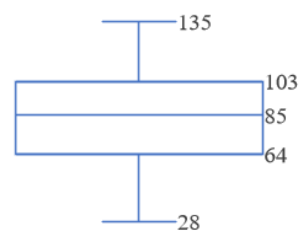
2.2.3 Additional Experiment

Another 15 participants (mean age = 26.20 years old, range 19 – 36) were recruited. These participants were asked to do the same tasks in the same setting but only without the diagnosis support system. This additional experiment result helps to identify the impact of the diagnosis support system during the maintenance activity.

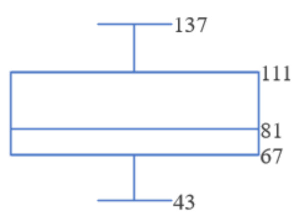
The average time to diagnose the sensor, the battery, the light bulb, and the wire of newly recruited participants was 75 seconds, 85 seconds, 86 seconds, and 54 seconds, respectively as shown in Figure **2-9**.



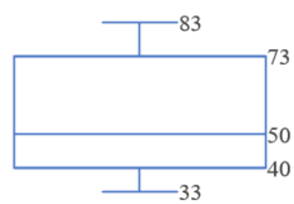
(a) Sensor Diagnose Time



(b) Battery Diagnose Time



(c) Light Bulb Diagnose Time



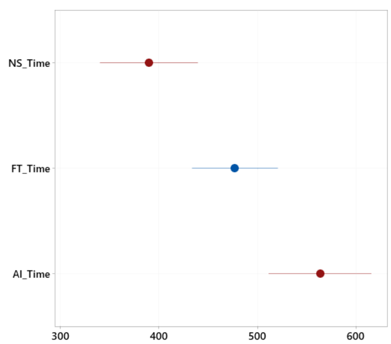
(d) Wire Diagnose Time

Figure **2-9**: Preliminary Study Result of Additional Participants

To observe how the usage of a diagnosis support system affects technicians during the maintenance activities by comparing three groups, one-way ANOVA analysis was conducted.

Time to Complete

The group that did not use either type of diagnosis support system took 389.87 seconds on average to complete the maintenance task with a standard deviation of 110.52. The group without the diagnosis support system took the least amount of time, and the group that used the AId-system took the most time to complete the task as shown in Figure **2-10**. The statistically-significant difference only exists between the AId-system group and the group that does not use a diagnosis support system as shown in Figure **2-11**.



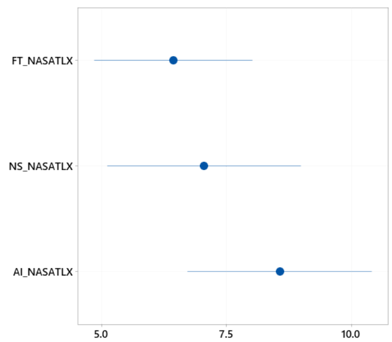
#	Group Name	Mean differs from
1	No Diagnosis Support System (NS_Time)	3
2	FT Diagnosis Support System (FT_Time)	-
3	AI Diagnosis Support System (AI_Time)	1

Figure **2-10:** One Way ANOVA Analysis for Time to Complete

Figure **2-11**: Significant Test for Mean Time to Complete Difference

NASA TLX

The mean workload of the group without the diagnosis support system was 7.06 with standard deviation 4.27; this workload is greater than that of the group that used the AId-system but less than that of the group that used the FTd-system as shown in Figure **2-12**. Although the mean workload of the group that used the AId-system is slightly greater than the other two groups' mean workloads, a statistically-significant difference between the three groups' workloads is not identified according to the ANOVA analysis as shown in Figure **2-13**.



#	Group Name	Mean differs from
1	FT Diagnosis Support System (FT_NASATLX)	-
2	No Diagnosis Support System (NS_NASATLX)	-
3	AI Diagnosis Support system (AI_NASATLX)	-

Figure **2-12**: One Way ANOVA Analysis for NASA TLX

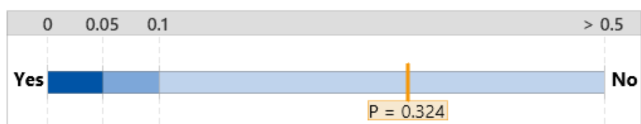
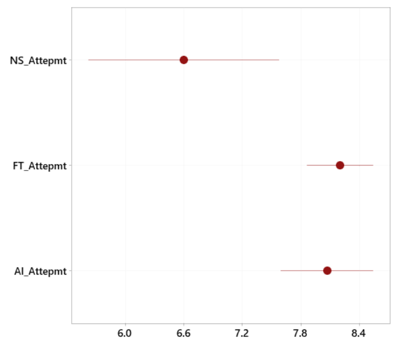


Figure **2-13**: Significant Test for Mean NASA TLX Difference

Maintenance Attempts

The expected number of attempts for the group that did not use any diagnosis support system was 8 attempts [45]; this number of attempts can also be achieved when the other groups strictly follow the recommendation of the support systems and make no errors. The mean number of maintenance attempts of the group that did not use the support systems was 6.60 attempts with standard deviation 1.92. This is less than the analytical expectation and the other groups as shown in Figure **2-14**.

There is a statistical difference in the mean number of maintenance attempts between groups. The mean of the group that did not use the maintenance support system is less than the other groups' mean as shown in Figure **2-14** and Figure **2-15**.



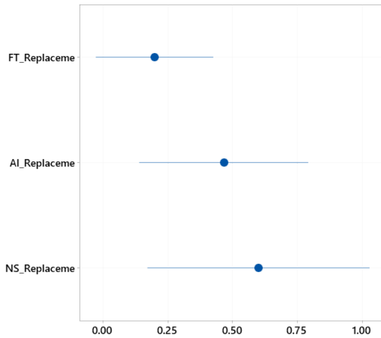
#	Group Name	Mean differs from
1	No Diagnosis Support System (NS_Attempt)	2, 3
2	FT Diagnosis Support System (FT_Attempt)	1
3	AI Diagnosis Support system (AI_ Attempt)	1

Figure **2-14:** One Way ANOVA Analysis for Maintenance Attempt

Figure **2-15**: Significant Test for Mean Attempts

Unnecessary Replacement

The mean of unnecessary replacement for the group that did not use the diagnosis support system was 0.6 with standard deviation 0.91. Again, the mean for this group was slightly greater than the other two groups as shown in Figure **2-16**. However, there is no statically-significant difference between the means of the three groups as shown in Figure **2-17** according to the one-way ANOVA analysis.



#	Group Name	Mean differs from
1	FT Diagnosis Support System (FT_Replaceme)	-
2	AI Diagnosis Support System (AI_Replaceme)	-
3	NS Diagnosis Support system (NS_Replaceme)	-

Figure **2-16:** One Way ANOVA Analysis for Unnecessary Replacement

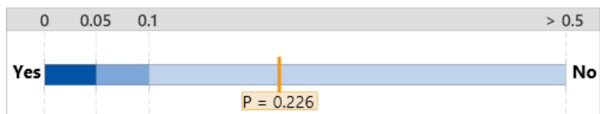


Figure **2-17**: Significant Test for Mean Unnecessary Replacement

2.3 NGOMSL Analysis

According to the results of the experiments, no differences were observed in the workloads of technicians and unnecessary replacement between groups. However, the statistically-significant difference of the metrics related to maintenance speed is as shown in Table **2-5**.

Table **2-5**: Experiment Result Summary

	FTd-System	AId-System	No Support System
Time to complete	477.20 seconds	563.47 seconds	389.87 seconds
NASA TLX	6.44	8.56	7.06
Diagnosis Attempt	8.20 attempts	8.07 attempts	6.60 attempts
Unnecessary Replacement	0.20 parts	0.46 parts	0.60 parts

We modeled the experiment settings using NGOMSL for further analysis. First, the difference in 'time to complete' is investigated. Next, the effect of the number of options shown in the AId-system is studied.

2.3.1 The difference in Time to Complete

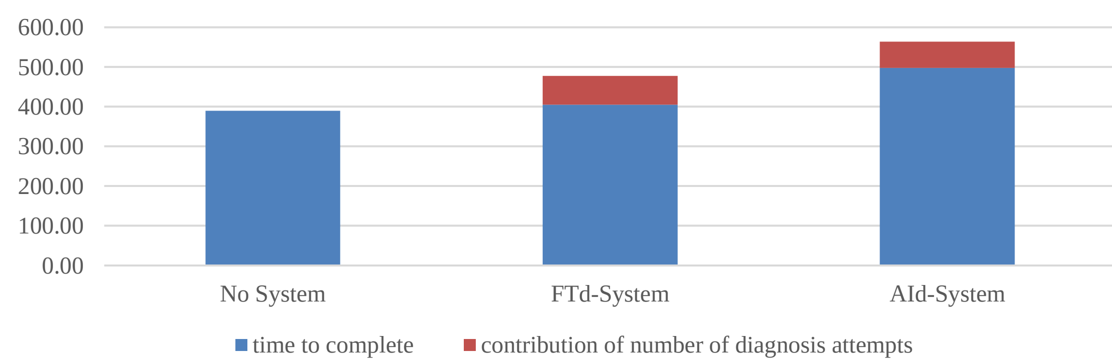


Figure **2-18**: Difference in Time to Complete between groups

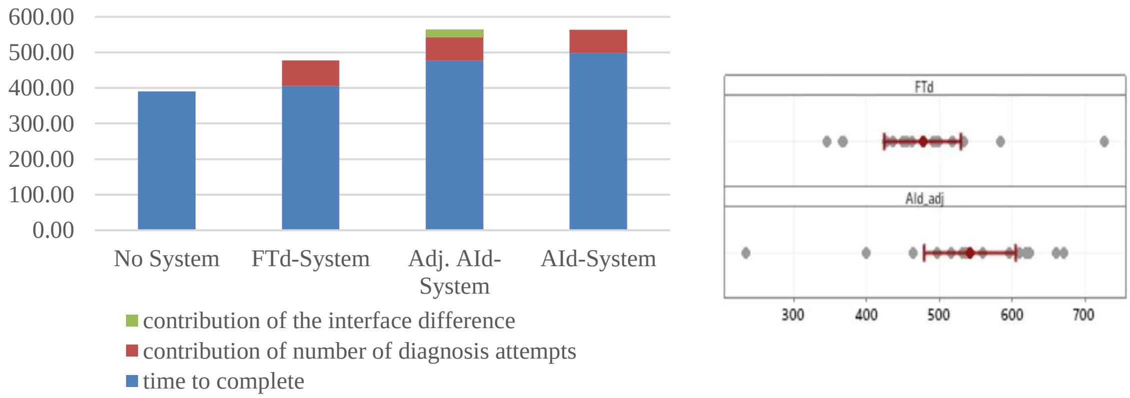
Time to complete mainly depends on two factors [45]. The number of maintenance attempts and the time spent on diagnosing, repairing, or replacing a component.

If the number of diagnosis attempts is significantly different between groups, the mean time to complete should be different. Every additional attempt to diagnose a wire takes 45 seconds and an attempt to diagnose a sensor takes 66 seconds according to the preliminary study session result of this experiment. Since there is a difference of approximately 1.5 diagnosis attempts between the group without a diagnosis support system and with a diagnosis support system, the contribution of the difference in attempts is about 68 seconds as shown in Figure **2-18**.

Table **2-6**: Keystroke Level Model Analysis on the Interface Difference

FT diagnosis support System				AI diagnosis support System			
#	Type	Statement	Time	#	Type	Statement	Time
				E1	t_p	Read Screen	0.10
				E2	M	Select a component (Mental)	1.35
				E3	P	Move the mouse to the selected component	1.10
				E4	B	Click the selected component	0.10
1	t_p	Read Screen (Diagnosis Instruction)	0.10	1	t_p	Read Screen (Diagnosis Instruction)	0.10
2	-	Do Diagnosis	-	2	-	Do Diagnosis	-
3	P	Move the mouse to the “NO” Button	1.10	3	P	Move the mouse to the “NO” Button	1.10
4	B	Click the “YES” Button	0.10	4	B	Click the “YES” Button	0.10

To understand the difference in completion times between groups that use a diagnosis support system, the interface difference is evaluated by comparing NGOMSL statements. For every diagnosis attempt, the interaction with the AId-system requires an additional 4 NGOMSL statements which contribute roughly 2.65 seconds according to the keystroke level model [46], [47] as shown in Table **2-6**. If we adjust this difference, the average time to complete maintenance for the AId-system becomes 542.09 seconds. After the adjustment, the two groups' difference in the average time to complete is 65 seconds, the difference is not statistically-significant (p-value 0.1) as shown in Figure **2-19**.



(a) Average Time to Complete (b) Distribution of Data

Figure **2-19:** Average Time to Complete Difference

2.3.2 Impact of the number of options presenting in AId-System

In the current experiment setting, the AId-system shows all possible options at the same time with the corresponding probabilities which may demand a larger cognitive load. The effect of the number of options showing in the AId-system is analyzed using two metrics: the peak load on working memory and the working memory density. The interaction between the user and the system is modeled using NGOMSL; this model is composed of 8 methods and 2 selection rules as shown in Table **2-7**.

Table **2-7:** NGOMSL for AId-System

1. Method to 'maintain a system'	
S1	Retrieve LTM tat current item in command sequence is maintain a system
S2	Accomplish the goal of 'diagnose a component'
S3	Decide: if the system is not working,
S4	then Goto 1-S2
S5	Report goal of 'maintain a system' accomplished

2. Method to accomplish the goal of 'diagnose a component'	
S1	Recall 'diagnose a component' and accomplish the goal of 'select a component'
S2	Recall A and accomplish the goal of 'click a component'
S3	Read Screen and retain instruction
S4	DO (External - diagnosing selected component)
S5	Forget instruction and accomplish the goal of 'repair a component'
S6	Report goal of 'diagnose a component' accomplished

3. Method to accomplish the goal of 'select a component'	
S1	read next button in a screen, retain the component name as A, and retain probability as B
S2	read next button in a screen, retain the component name as A' and retain probability as B'
S3	Decide if B' > B
S4	then retain B' as B, A' as A and forget A' and B'
S5	forget B and report the goal of 'select a component' to accomplished

4. Method to accomplish the goal of 'click a component'	
S1	Read screen
S2	Move the cursor to A
S3	Click A
S4	forget A and report the goal of 'click a component' to accomplished

5. Method to accomplish the goal of 'select a component'	
S1	Read screen
S2	Move the mouse to "Yes"
S3	Click "Yes"
S4	report goal of 'report status' and 'repair a component' accomplished

6. Method to accomplish the goal of 'fix a component'	
S1	Read screen
S2	Move the mouse to "No"
S3	Click "No"
S4	read screen and retain instruction
S5	DO (External - replacing a component)
S6	Forget instruction and accomplish the goal of 'report system condition'

7. Method to accomplish the goal of 'finalize the maintenance'	
S1	read screen
S2	move the mouse to "Yes"
S3	Click "yes"
S4	report goal of 'finalize the maintenance', 'report system condition', fix a component' and goal of 'repair a component' accomplished

8. Method to accomplish the goal of 'update the status'	
S1	Read screen
S2	Move the mouse to "No"
S3	Click "No"
S4	report goal of 'update the status', 'report system condition', 'fix a component' and 'repair a component' accomplished

SR1. Selection rule set for the goal of 'repair a component'	
	if the component is in good condition, then accomplish the goal of 'report status
	if the component is not in good condition, then accomplish the goal of 'fix a component'

SR2. Selection rule set for the goal of 'report system condition'	
	if the system condition is good then accomplish the goal of 'finalize the maintenance'
	if the system condition is not good then accomplish the goal of 'update the status'

Under the current experiment setting, participants detect broken components at the 1^{st}, 4^{th}, 7^{th}, and 8^{th} diagnosis attempts when they strictly follow the recommendation of the system. This processing sequence requires a total of 321 NGOMSL statements and a total of 1,429 items to be remembered. In this case, the peak working memory and working memory density for this setting are 7 and 4.45 respectively.

If the AId-system only shows a limited number of options such as two, three, or four options at every diagnosis attempt, the difference occurs in "Method to accomplish the goal of 'select a component'". For example, if there are 9 options to be shown, it requires twenty-six NGOML statements whereas only five NGOML statements are required if only 2 options are shown as shown in Table **2-8**.

Table **2-8:** NGOML Statements for Two Different Cases

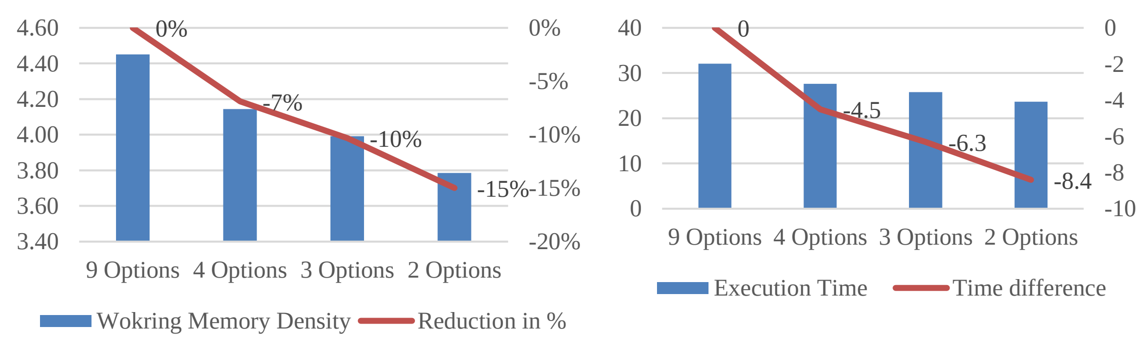
When 9 options are shown			When 2 options are shown		
#	Ref	Statement	#	Ref	Statement
1	1-s1	retrieve LTM tat current item in command sequence is 'maintain a system	1	1-s1	retrieve LTM tat current item in command sequence is 'maintain a system
2	1-s2	Accomplish goal of 'diagnose a component'	2	1-s2	Accomplish goal of 'diagnose a component'
3	2-s1	Recall 'diagnose a component' and accomplish the goal of 'select a component'	3	2-s1	Recall 'diagnose a component' and accomplish the goal of 'select a component'
4	3-s1	read next button in a screen, retain the component name as A, and retain probability as B	4	3-s1	read next button in a screen, retain the component name as A, and retain probability as B
5	3s-2	read next button in a screen, retain the component name as A' and retain probability as B'	5	3s-2	read next button in a screen, retain the component name as A' and retain probability as B'
6	3-s3	Decide if B' > B	6	3-s3	Decide if B' > B
7	3-s4	then retain B' as B, A' as A and forget A' and B'	7	3-s4	then retain B' as B, A' as A and forget A' and B'
Statements 5 through 7 would be executed 7 more times			8	3-s5	forget B and report the goal of 'select a component' accomplish
29	3-s5	forget B and report the goal of 'select a component' accomplish	9		

The total number of NGOML statements, the total number of items to be remembered, peak working memory, and working memory density are compared for the different settings as shown in Table **2-9**.

Table **2-9:** Number of NGOML Statements difference

	# of Statements	# of items to be remembered	Peak Working Memory	Working Memory Density
9 options	321	1429	7	4.45
4 options	276	1144	7	4.14
3 options	258	1030	7	3.99
2 options	237	897	7	3.78

When we limit the number of options shown in the AId-system, the peak working memory does not change. However, as the number of options shown decreases, the execution time and the working memory density decrease. We estimate that if we limit the number of options shown to two, the execution time and cognitive load is reduced by 8.4 seconds and 15%, respectively, as shown in Figure **2-20**.



(a) Working Memory Density (b) NGOMSL Statement Execution Time

Figure **2-20**: The difference in Working Memory Density and Execution Time due to the number of options shown

2.4 Summary and Future work

One challenge to adopt new systems and change the current process is not only the budget but also unknown outcomes. The effect of introducing smart systems needs to be analyzed in multiple perspectives and analyzed in the context of real and practical usage [13]. Although much effort has been made to study tools, methodologies and enablers for offering smart service-product systems [9], more studies are required for social impact and to have an effect on users. Thus, this research is motivated to investigate the unwelcomed effects of a “smart” system when it aids the technician during maintenance. In sum, the results and further analysis of this experiment show that adopting a smart diagnosis support system may improve maintenance productivity by reducing the number of diagnosis attempts without burdening technicians with new work.

The experiment results show that the two different diagnosis support systems do not have statistically-significant impacts on the workload of technicians and maintenance quality. On the other hand, the participants’ maintenance time is statistically different depending on the system with which they interacted.

The effect of the interface differences in the two systems is analyzed by the NGOMSL model. Interacting with the AId-system requires extra keystrokes compared to interacting with the FTd-system. Also, for every diagnosis attempt, the AId-system requires an extra 2.65 seconds, which is one of the main sources of the maintenance time difference between the two systems. Additionally, the effect of the number of options shown in the AId-system is analyzed. As a smaller number of options are shown to the user, a smaller cognitive load is imposed during maintenance. We estimate that the maintenance time and the cognitive load can be reduced by 8.4 seconds and 15% if only two options are shown to participants. However, further experimentation is required to confirm the estimation of cognitive load reduction and its impact.

Based on the experiment and analysis, we recommend using an AId-system during maintenance. We could not find a significant drawback of the AId-system and it may be even better compared to an FTd-system if the accuracy of recommendation improves.

Nevertheless, although we attempted to duplicate a real CNC machine maintenance environment in the lab, the results may not be the same as the results of an experiment conducted at a real site with a real machine. Further studies are necessary with a more realistic setting in order to completely capture the effects of diagnosis support systems on technicians during maintenance. Also, additional studies are warranted in the future with different maintenance tasks since the experimental result differs based on the machine that needs to be maintained or the skill of technicians.

Chapter 3

Modeling Maintenance Time: Effect of Technician's Proficiency

Although maintenance is not directly related to value creation activities in the manufacturing or service sectors, poorly managed maintenance can trigger great losses. A small component in and of itself may not be a core part of a particular machine or system, but failure to maintain even a minor component may cause a machine or whole system to stop. This, in turn, hinders value creation activities. Therefore, different maintenance policies that utilize sensors and cumulative data to prevent such failures are popularly adopted in practice and researched in academia. Nevertheless, corrective maintenance is still required since not all failures can be prevented by such policies.

In corrective maintenance, a technician repeats two tasks until they fix a system or find all components that need to be repaired [48]. The first task is to select a component on which to focus their attention and the subsequent task is to diagnose and perform maintenance on the selected component if necessary [49]. Thus, maintenance time can be decreased in two ways: by reducing the time to localize, isolate, adjust, disassemble, repair, interchange, reassemble, align, and inspect a component [49] or by reducing the number of diagnosis attempts.

The rest of this chapter is organized as follows. In Section 1, we summarize various methods to model the maintenance time and proficiency effects on performance. In Section 2, the difference between a skilled technician and a naive technician is explained. Then, the maintenance time estimation model using the Negative-Hyper Geometric distribution, which has the following inputs: (1) number of components, (2) number of components not in working state, and (3) time to repair/replace a component considering the proficiency effect, is proposed. In Section 3, the experiment results presented in Chapter 2 are evaluated using the proposed model. Lastly, Section 4 discusses the limitations of the model and possible future studies.

3.1 Literature Review of Maintenance Time Models

Maintenance is the last area to be automated in manufacturing because humans are responsible the majority of its activities [10], [11]. Humans learn by doing tasks repeatedly which, in turn, affects their performance. Thus, considering the proficiency of technicians is reasonable when one models maintenance time.

3.1.1 Proficiency Effect on Performance

The proficiency of technicians depends on multiple factors. Of many factors, experience plays a critical role. According to a survey conducted in 2000, the years of experience with equipment and the proficiency of technicians are correlated [12].

A later study reports that the duration of the training period is another factor that affects proficiency [50]. The learning curve [51], is a traditional method to describe the proficiency effect. First proposed by Wright [52], many of its applications can be found in the domain of manufacturing. Especially, many studies focused on the learning curve effect on assembly and production lines or production planning. For example, the production schedule was studied considering not only learning but also forgetting due to scheduled maintenance and other production interruptions [53]. Similarly, a new learning curve was proposed considering the interruption of the production process due to production quality issues [54]. Another studied assembly line balancing problems considering learning with realistic assumptions [55].

Cabahug et al. identified three personal attributes as key for predicting the proficiency of technicians: years of relevant working experience on the machine, personal disposition, and operator reliability [56]. Magne and Dag, however, claimed in their study that experience has little impact on predicting problems in software maintenance although they considered the number of years in general maintenance rather than the number of years in a particular type of maintenance [57]. Tien considered the proficiency of technicians for a maintenance schedule model [58].

As discussed, a limited number of studies have investigated the effect of the technicians' proficiency and included it in the maintenance models, although this may potentially change the maintenance time and quality. Thus, studying how technicians' proficiency can affect maintenance time is necessary.

3.1.2 Maintenance Time Model

Various methods to model maintenance time can be found in the literature. One of the most popular methods is modeling failure and repair of a machine as queuing models which assumes an exponential distribution [59], [60]. To overcome the limitation of a model with exponential distributions, generalized exponential distributions were considered to better represent the behavior of machine failure and repair [59]. Markov process with phase-type distributions was another method used to model the maintenance process [60], [61]. Similarly, a Markov system with hypo-exponential distribution was also used to model the optimal rate of periodic preventive maintenance [62]. A model that used two different distributions was also used. The Weibull distribution was used for the failure modes; the exponential distribution was used for the failure probability for each failure mode [63].

Due to the analytical traceability of the exponential distribution family, extant research on maintenance models uses such distributions. At the cost of fidelity, these models are limited in their ability to capture essential characteristics of maintenance such as technician's proficiency. Hence, developing models with better fidelity for maintenance operations is needed to represent maintenance activities more precisely.

3.2 Modeling Maintenance Time using Negative Hyper-Geometric Distribution

There are two main tasks in the corrective maintenance mechanism. A technician repeats these until they fix a system or find all components that need to be repaired. The first task is to select a component on which to focus their attention and the subsequent task is to diagnose and perform maintenance on the selected component if necessary. Thus, the maintenance time (MT) depends on X, the number of components selected for diagnosis, and the time to diagnose and perform maintenance on the selected components as shown in Equation 3-1 [45] assuming that the executing time for selecting a component is trivial. In this model, there are N possible components which might cause a machine failure. Since the time to perform the second type of task, diagnosing and performing maintenance on the selected component, varies by failure types and components, it is defined as a random variable T_i, which is generally distributed.

$$\mathrm{MT} = \sum_{i=1}^{X} T_i \tag{3-1}$$

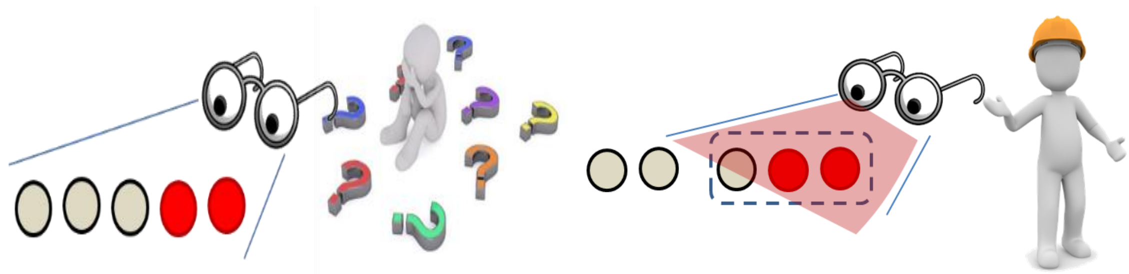
Given that

T_i = time to check, repair and replace task i, is i.i.d. to T_1

X = number of selected components to focus technicians' attention

3.2.1 Modeling Component Selection Process

The process of corrective maintenance is similar to drawing all 'fail' balls from the box which contains N balls without replacement. Therefore, the number of maitenance attempts, X, is defined as the Negative Hyper-Geometric distribution, $X \sim NHG(N, M, k)$ [64] as shown in Equation (3-2) [45]. In this model, k is always equal to M because we assume that technicians do not terminate their task of system maintenance unless they find and repair/replace all broken components.



(a) without proficiency (b) with proficiency

Figure **3-1**: Proficiency Effect

$$\Pr(X = x) = \frac{\binom{x-1}{k-1}\binom{N-x}{M-k}}{\binom{N}{M}} = \frac{\binom{x-1}{M-1}}{\binom{N}{M}}$$

Where we define,

k is the number of components that need to be selected; (3-2)

M is the number of components that are not in working order;

N is the number of possible components that are not in a working state

$k = M$ and $M \leq N$.

As illustrated in Figure **3-1**, a more proficient technician typically requires a smaller number of maintenance attempts in order to perform system maintenance compared to a less proficient technician. This is because proficient technicians do not consider some components as possible components that could cause the failure of the system. To mode this experience effect, we define N in Equation 3-2 as N^*as shown in Equation 3-3 [48]. EE_N cannot be greater than $N - M$ since at least M components must be diagnosed even for a highly proficient technician.

$$N^* = N - EE_N \tag{3-3}$$

Where

N^*: number of possible components in technician's opinion

EE_N: proficiency effect of N, $0 \leq EE_N \leq N - M$

3.2.2 Analytical and Simulation Model

As shown in Equation 3-1, the number of maintenance attempts plays a critical role in estimating the maintenance time. EE_N and M are two key inputs for the estimation as shown in Equation 3-2 and 3-3.

However, accurately estimating these are a bit challenging. Therefore, analyzing their impact on the model is needed. The sensitivity of model inputs, EE_N and M are analyzed using the simulation model [48]. The general simulation setting is as follows:

1. There are 20 possible components (N = 20)
2. The time for maintaining each task is constant and equal to 10 ($T_i = 10$)
3. The broken component(s) is randomly assigned at the beginning of each simulation integration

One's level of EE_N can be inferred by various methods. Analyzing the maintenance log is one such method. However, too many things need to be considered in order to determine one's level of EE_N. Therefore, maintenance time is collected by changing the variance of EE_N in the simulation to see how the variance of EE_N affects the maintenance time. In this simulation, the broken components, M, is equal to 1 and the proficiency effects are followed by uniform distributions with three different intervals $EE_N \pm 1$, $EE_N \pm 2$, $EE_N \pm 3$. As shown in Figure **3-2**, the variance of EE_N does not influence the maintenance time [48].

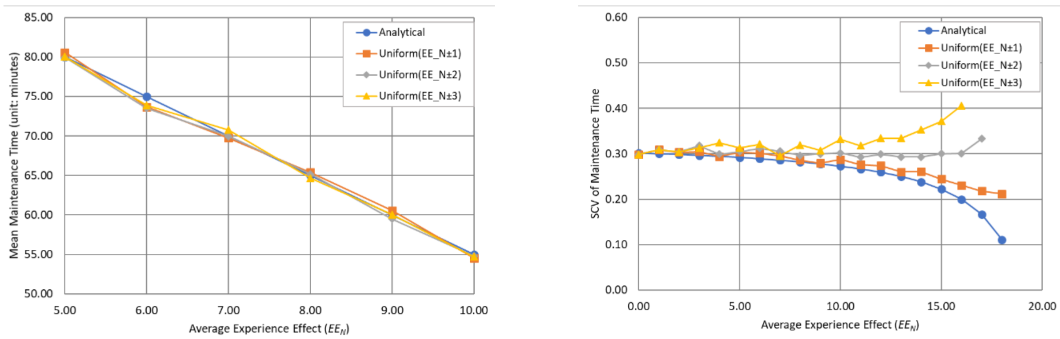


Figure **3-2**: Variance of the proficiency effect on the maintenance time

In most cases, the technicians do not know the exact failure mode or the exact cause of a particular failure mode. In those cases, specifying the number of broken components is the challenge. Therefore, the impact on the maintenance time due to the variation in M needs to be studied. In the simulation, M is treated as a random variable followed by uniform distributions with three different intervals $M \pm 1$, $M \pm 2$, $M \pm 3$. The simulation assumes that EE_N = 0. The result shows that the maintenance time is not highly dependent on the variance of M as shown in Figure **3-3** [48].

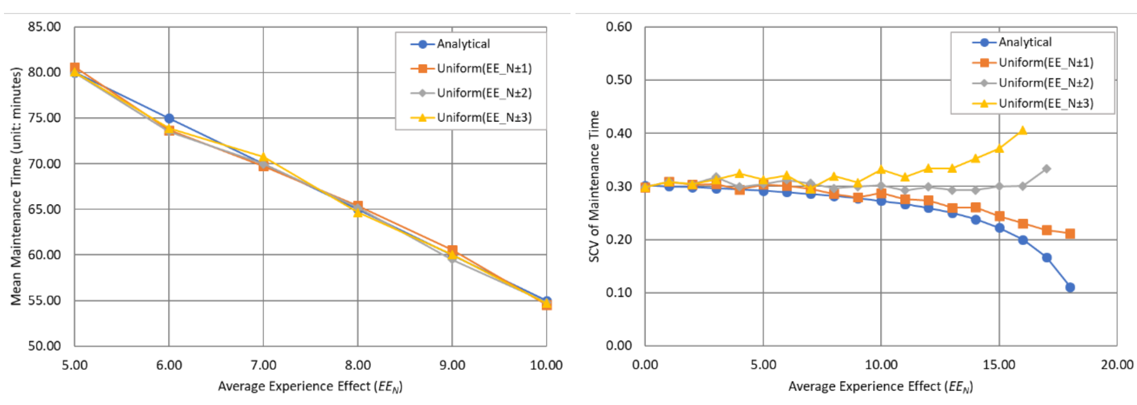


Figure **3-3**: Variance of the number of broken components on the maintenance time

3.3 Analyzing the Performance of Maintenance using NHG Model

In this section, we analyze the results of the experiment presented in Chapter 2 using the NHG model. In the experiment, participants were asked to perform maintenance on a system that is composed of 9 components, with or without the diagnosis support systems. There was one failure mode, and it was due to 4 broken components: switch, light bulb, battery, and black wire.

3.3.1 The distribution of the number of maintenance attempts

According to Equation 3-2, the expected number of maintenance attempts is 8 if the system has 9 components and 4 of them are broken as shown in Table **3-1**. The probability distribution is skewed to the right as shown in Figure **3-4**

The average number of diagnosis attempts for the group without a support system, the AId-system group, and the FTd-system group is 6.60, 8.06, and 8.12 attempts, respectively, as shown in Table **3-1**. The experiment result shows that the probability distribution of the NS group is similar to the uniform distribution whereas the probability distribution of the groups that used the support systems is similar to the NHG estimation as shown in Figure **3-4**.

The NS group completes the maintenance in fewer steps than what the NHG model, which assumes that participants have no proficiency. The NS group results are rather similar to the expectation which assumes that participants understand about 22% of the system. In other words, participants can eliminate 2 options on average (N* = 7, EE_n= 2) when they perform maintenance on the system.

Table **3-1**: Probability Distribution

# of Attempts	NHG		NS		AId-system		FTd-system	
	Frequency	P(x)	Frequency	P(x)	Frequency	P(x)	Frequency	P(x)
4		0.0079	3	0.2000	-	0.0000	-	0.0000
5		0.0317	3	0.2000	1	0.0667	-	0.0000
6		0.0794	-	0.0000	1	0.0667	-	0.0000
7		0.1587	3	0.2000	-	0.0000	-	0.0000
8		0.2778	3	0.2000	7	0.4667	12	0.8000
9		0.4444	3	0.2000	6	0.4000	3	0.2000
Average	8.0		6.6		8.1		8.2	

In addition to comparing the probability distribution, the uncertainty in the number of diagnosis attempts is evaluated using Shannon's entropy model. A high entropy, in this case, indicates that it is more difficult to correctly predict how many attempts are required to complete the maintenance task. On the other hand, a low entropy, in this case, indicates a or some specific numbers of attempts are more likely to be expected than others.

According to Shannon's entropy model, the NS group has the highest uncertainty followed by the AId-System group, the NHG model, and the FT-system group as shown in Figure **3-5**. Although the expected number of attempts required is similar for the NHG, AId-system, and FTd-system groups, their entropies are different.

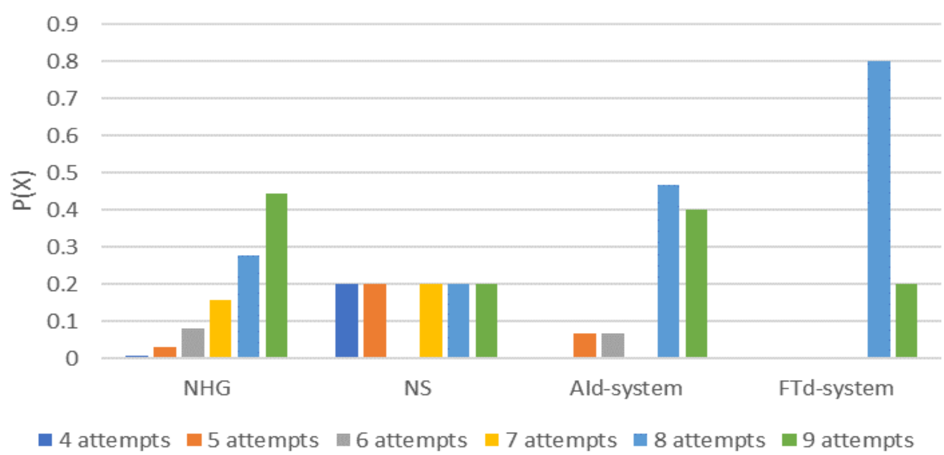


Figure **3-4**: The probability distribution of the number of maintenance attempts

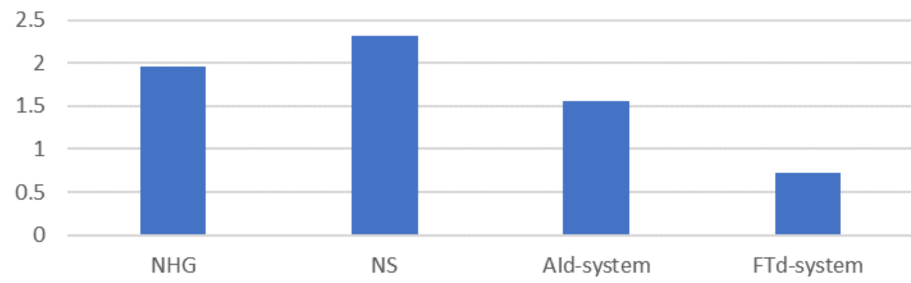


Figure **3-5**: Shannon's Entropy

3.3.2 Maintenance Time Estimation

The results of experiments are compared to the results of a simulation which is based on the NHG model. In the simulation, the data collected during the preliminary study of the experiment in Chapter 2 is used as maintenance time for each component. Each component's maintenance time follows the normal distribution after the Box-Cox transformation except battery maintenance time and switch maintenance time as shown in Table **3-2**. The maintenance time is constant and equal to 5.00 seconds. The simulation setting is adjusted to replicate the settings of the experiment as shown below:

- $N = 9$ components (C1: switch, C2: black wire, C3: battery, C4: light bulb, C5: yellow wire, C6: white wire, C7: red wire, C8: green wire, and C9: sensor)
- $M = 4$ components: instead of randomly assigning the broken components, C1, C2, C3, and C4 are assigned as broken components.
- Technician randomly identifies some components which they think should be checked. All failed components must be identified.
- With proficiency effects EE_N, technicians complete maintenance in fewer maintenance attempts.
- The maintenance time for each component follows the distribution stated in Table **3-2** except C1. C1's maintenance time is constant and 5 seconds.

- Maintenance time is collected after the technician repaired the last failed component.
- Data of 2000 maintenance sessions is collected (simulation iterations = 2000)

Table **3-2**: Maintenance Time for each Task

	Sensor	Battery	Light Bulb	Wire	Switch
Average	69.24	82.58	85.69	48.69	5.00
Standard Deviation	23.27	22.46	25.60	14.32	0.00
Lambda (Transformation)	0.50	1.00	-0.50	0.50	-
After Transformation Average	8.21	-	0.11	6.90	-
After Transformation Standard Deviation	1.40	-	0.02	1.03	-
Goodness of Fit (P-value)	0.66	0.40	0.59	0.99	-

Simulation I: $\boldsymbol{EE_N = 0}$

The NHG model estimates that 8 attempts are required to perform maintenance on the system used in the experiment without the proficiency effect. According to the simulation, it took 8.01 attempts on average to perform maintenance on the system with a standard deviation of 1.14. The coefficient of variation (CV) is 0.14. The distribution is similar to the prediction of the NHG model as shown in Figure **3-6**.

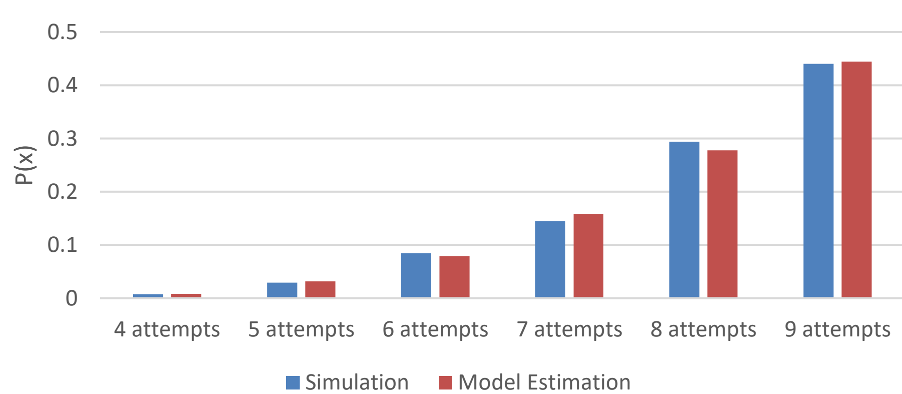


Figure **3-6**: The Probability Distribution when Proficiency Effect = 0

In the simulation, performing maintenance took 326.37 seconds on average, with a standard deviation of 57.30. The longest time and shortest time for a single maintenance session are 605.81 seconds and 177.80 seconds, respectively. The CV of maintenance time is 0.18.

Simulation II: $\boldsymbol{EE_N = 2}$

When there is a proficiency effect of 2, the NHG model estimates that 6.40 attempts are required to perform maintenance on the system. According to the simulation result, 6.41 attempts are required on average with a standard deviation of 0.78. The CV is 0.12. The distribution of the simulation result is similar to the NHG model prediction as shown in Figure **3-7**.

In the simulation which assumes that $EE_N = 2$, maintenance took 317.58 seconds on average, with a standard deviation of 49.04. The longest time and shortest time for a single maintenance session are 632.82 seconds and 179.56 seconds, respectively. The CV of maintenance time is 0.15.

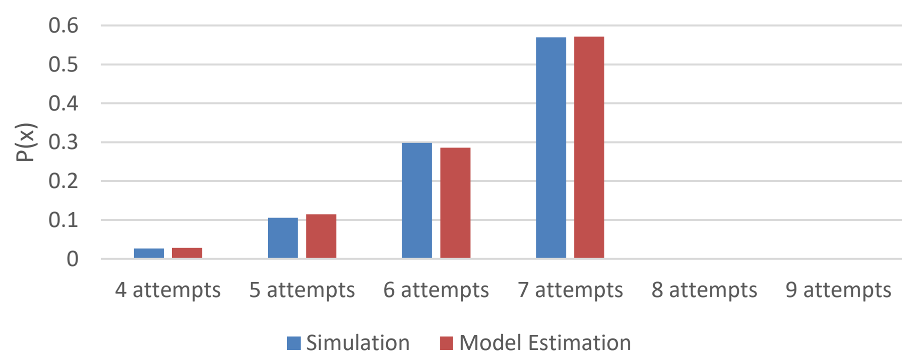


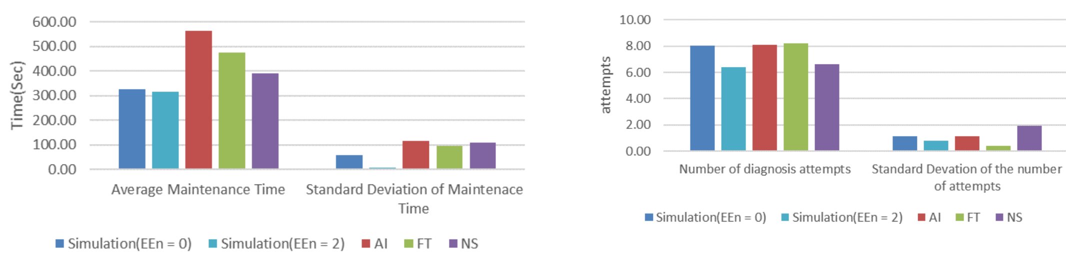
Figure **3-7**: The Probability Distribution when Proficiency Effect = 2

Analysis of the simulation results

There is a difference in the maintenance time between the simulation results and the experimental results as shown in Figure **3-8**. However, CVs are similar except for the NS group as shown in Figure **3-9**

The CV difference in the number of diagnosis attempts between simulation results and the AId-group is similar to the prediction. The CV of the simulation and the AI group only differs by 0.02 as shown in Figure **3-9**

When the simulation assumes that $EE_N = 2$, the difference in the number of diagnosis attempts between the simulation and NS group is insignificant. However, the NS group's maintenance time and the number of diagnosis attempts tend to have greater variability than the model estimation.



(a) Maintenance Time (b) Diagnosis Attempts

Figure **3-8**: Maintenance Time and Diagnosis Attempts Comparison

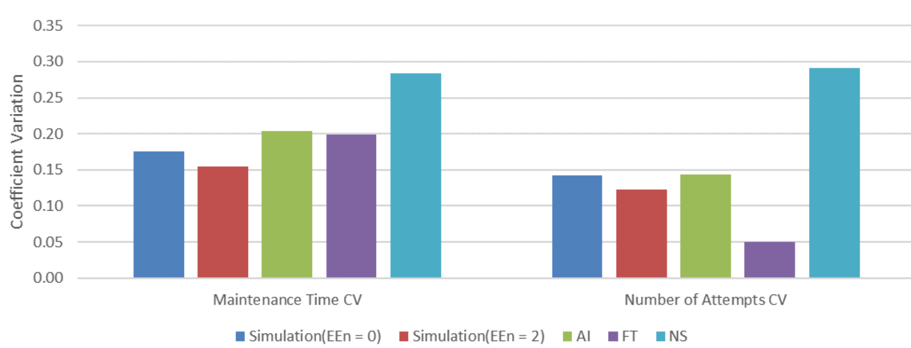


Figure **3-9**: CV of Maintenance Time and Diagnosis Attempts Comparison

3.3.3 Variability Impact on Skilled Technicians

Since the diagnosis system may improve the predictability of outcome by reducing the variability, further analysis is conducted to investigate its effect on different proficiency groups. The experiment data is analyzed using the k-means clustering and elbow methods. The following four attributes are used for the clustering: maintenance time for the sensor, battery, light bulb, and wire. First, the elbow plot is constructed to find the ideal number of clusters as shown in Figure **3-10**. Since the line bends at 3, the experiment data is clustered into 3 clusters.

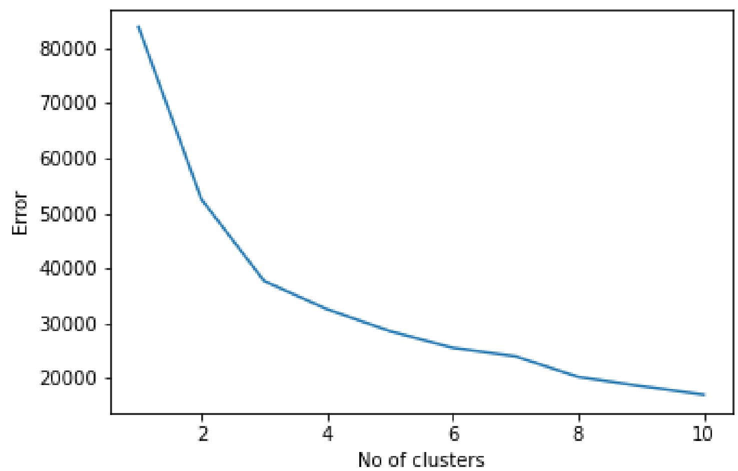


Figure **3-10**: Elbow Plot Result

The Euclidean distance is used to find the centroids of clusters. By summing the values of the centroids, we classify the clusters as follows: very skilled, moderately skilled, and low skilled as shown in Table **3-3**.

Table **3-3**: Summary of Clusters

Cluster #	Classification	Sensor	Battery	Light Bulb	Wire	SUM
Cluster 1	Very Skilled	53.35	67.30	73.40	40.10	234.15
Cluster 2	Low Skilled	61.86	104.00	131.57	53.57	351.00
Cluster 3	Moderately Skilled	89.78	98.72	81.50	56.30	326.30

According to the result of the k-means clustering method, of 45 participants, the number of very skilled, moderately skilled, and low skilled participants are 20, 18, and 7 respectively as shown in Table **3-4**.

Table **3-4**: Result of K-means clustering method

	NS	FT	AIL	Total
Very Skilled	6	9	5	20
Moderately Skilled	6	5	7	18
Low Skilled	3	1	3	7

The average maintenance time, standard deviation of maintenance time, and coefficient of variation of each group by skill level are calculated as shown in Table **3-5**. If the participants are either low skilled or moderately skilled, using a diagnosis support system reduces the CV as shown in Figure **3-11**.

Table **3-5**: Maintenance Time by Clusters

Very Skilled	NS	AId-system	FTd-system
Average Maintenance Time	350.83	468.60	446.56
Standard Deviation of Maintenance Time	81.76	135.91	69.21
Coefficient of Variation	0.23	0.29	0.15
Delta Coefficient of Variation		24.44%	-33.50%
Moderately Skilled	NS	AId-system	FTd-system
Average Maintenance Time	361.83	593.86	524.20
Standard Deviation of Maintenance Time	108.72	75.76	129.19
Coefficient of Variation	0.30	0.13	0.25
Delta Coefficient of Variation		-57.54%	-17.98%
Low Skilled	NS	AId-system	FTd-system
Average Maintenance Time	524.00	650.67	518.00
Standard Deviation of Maintenance Time	72.99	32.08	0.00
Coefficient of Variation	0.14	0.05	0.00
Delta Coefficient of Variation		-64.60%	-100.00%

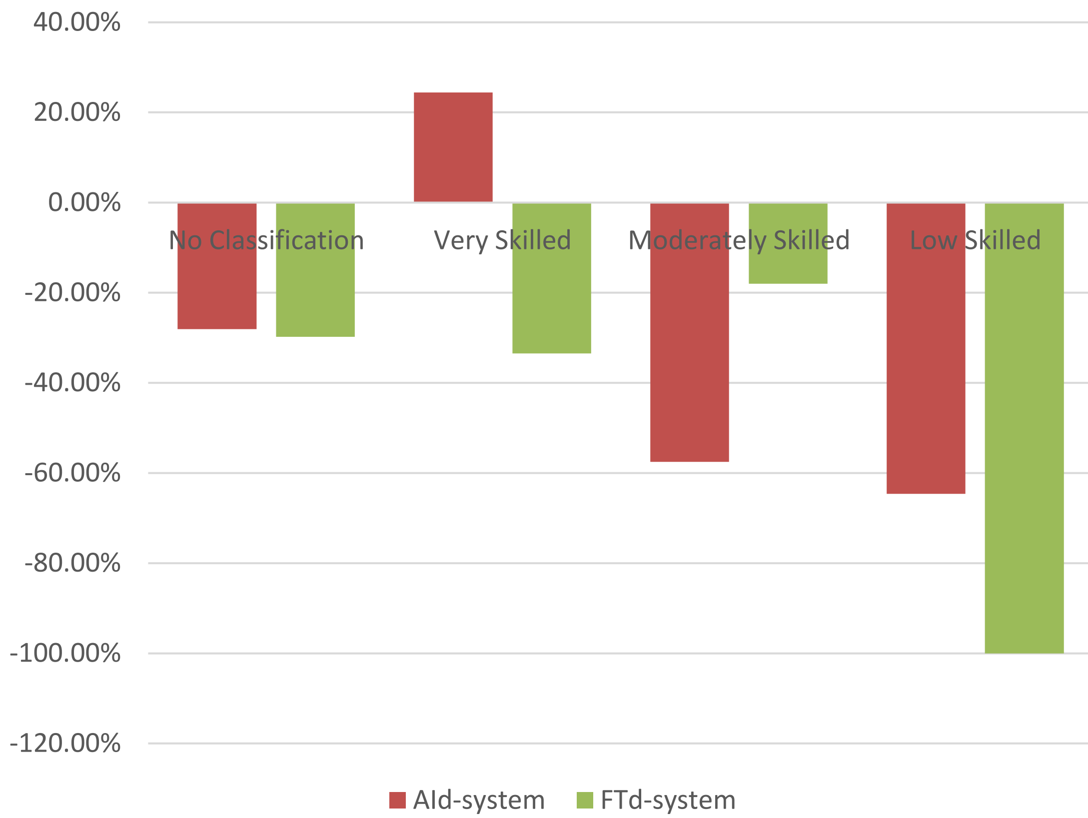


Figure **3-11**: CV Delta Comparison by skilled groups

According to the factory physics [65], the CV of maintenance time is positively correlated to the CV of effective processing time as shown in Equation 3-4. Therefore, the CV of maintenance time is critical to the performance of a manufacturing line since its effect propagates to the cycle time (CT) and WIP level (WIP) of the line through the expected waiting time spent in a queue (CT_q) as shown in Equation 3-5, 3-6, and 3-7.

$$c_e^2 = \frac{\sigma_e^2}{t_e^2} = c_0^2 + (1 + c_r^2)A(1 - A)\frac{m_r}{t_0} \tag{3-4}$$

When the line is G/G/1 queue (3-5)

$$CT_q \approx V \times U \times t \approx \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t_e$$

$$CT = CT_q + t_e \tag{3-6}$$

$$WIP = r_a \times CT \tag{3-7}$$

Where,
m_r = the maintenance time
t_e = the process time
u = the utilization
c_e^2 = the squared CV of processing time
r_a = the rate of arrivals jobs per unit time
c_a^2 = the squared CV of inter-arrival time
CT_q = the expected waiting time spent in queue.

To illustrate the impact of the reduction in the CV of maintenance time on the performance of manufacturing, the case of a manufacturing cell line is presented. This cell line is a part of a commercial vehicle manufacturing line which consists of a robot and 3 machines as shown in Figure **3-12**. In this line, the robot picks up parts that arrive on an input conveyor belt and loads them into each machine.

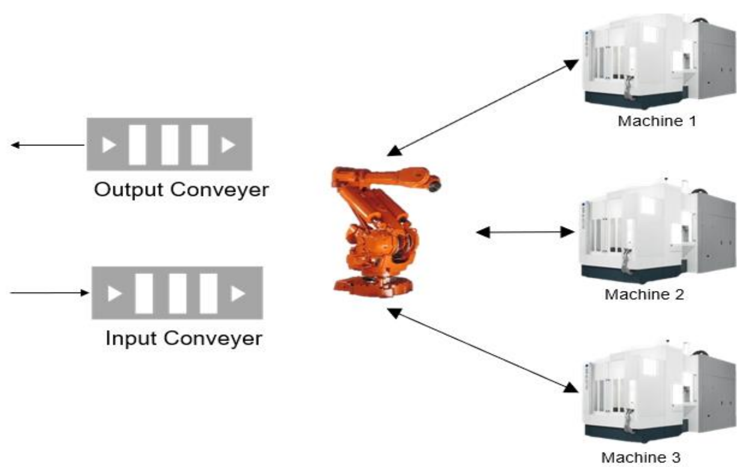


Figure **3-12**: Robotic Cell Line

The basic statistic of this case is collected from the field study as shown in Table **3-6**. With the current settings, the expected CT and WIP of each machine using Equation 3-3 and 3-4 are as shown in Table **3-7**.

Assuming that introducing a type of diagnosis support system only changes the CV of repair time, its impact on the CT and WIP is summarized in Table **3-7**. In the case, the cycle time and WIP level of the machine 1 are impacted more as shown in Figure **3-13**. If technicians are assumed to be low skilled, the impact of using the support system is greater.

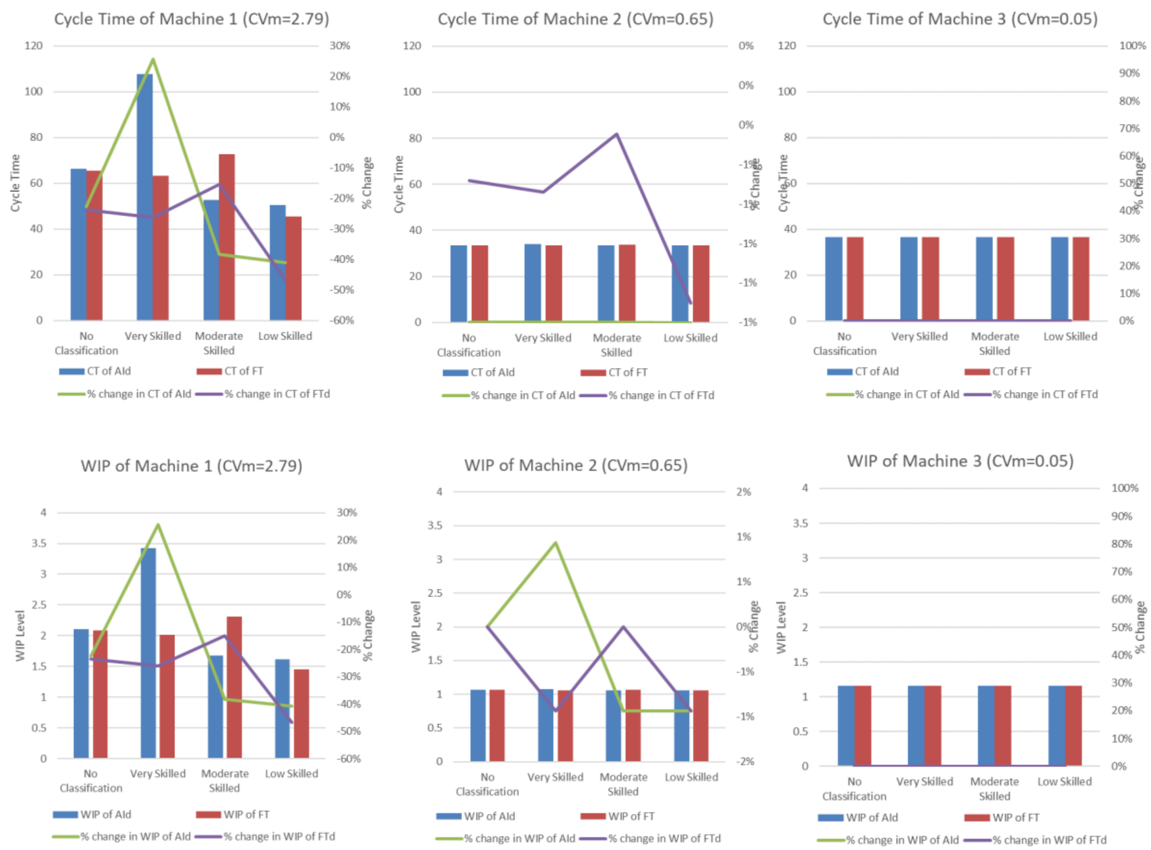


Figure **3-13**: Cycle Time and WIP of each machine

Table **3-6**: Basic Statistic of a Cell Line

Parameter	Value		
Available Production time/shift		430 mins	
Parts per shift		41 parts	
Availability for 3 machines		0.95	
	Machine 1	Machin 2	Machine 3
Processing Time	27.07 mins	19.67 mins	17.62 mins
Standard Deviation of Processing Time	8 mins	15 mins	20 mins
Repair(maintenance) Time	21.5 mins	21.5 mins	21.5 mins
Standard Deviation of Repair(maintenance) Time	60 mins	14 mins	1 min

Table **3-7**: CT and WIP of the system

		Machine 1		Machine 2		Machin 3	
		CT (mins)	WIP (parts)	CT (mins)	WIP (parts)	CT (mins)	WIP (parts)
Current		85.67	2.72	33.76	1.07	36.47	1.16
AId-System	Very Skilled	107.66	3.42	34	1.08	36.47	1.16
	Moderately Skilled	52.81	1.68	33.4	1.06	36.47	1.16
	Low Skilled	50.61	1.61	33.37	1.06	36.47	1.16
FTd-System	Very Skilled	62.26	1.98	33.5	1.06	36.47	1.16
	Moderately Skilled	72.55	2.31	33.61	1.07	36.47	1.16
	Low Skilled	45.58	1.45	33.32	1.06	36.47	1.16

3.4 Summary and Future work

In this chapter, based on the NHG distribution, the method to estimate the maintenance time considering the proficiency effect is proposed.

We noticed that the uncertainty in the number of diagnosis attempts in the NS group is greater than in others. However, the uncertainty in the groups that used a type of diagnosis support system is similar to the model estimation. This implies that the NS group completes maintenance beyond normal randomness. In other words, one benefit of using diagnosis support systems is probably lessening performance uncertainty by improving the predictability of the outcome. The recommendation or instruction of the diagnosis support system ensures that maintenance is performed in a structured way regardless of the people who maintain the system.

In future research, the limitation of the NHG model which eliminates the probability of diagnosing N components when EEn is greater than 0 needs to be improved. Also, the probability distribution of the number of diagnosis attempts which depend on the different levels of proficiency needs to be studied in future research.

Chapter 4

A Mode-Driven Framework for Business Process Reengineering: Integrating Analytical and Simulation Process Models

Since the 1990s, the use of business process modeling (BPM) has been growing along with the growing use of information technology across industries [66]. Besides its traditional use for communication between different stakeholders, BPM can be useful for transforming processes that manage work-flow [66]–[69]. Especially, the integration of sensors, actuator technology, communication solution [70], blurs the geographical and functional boundary between actors, parties, and departments. This, in turn, creates enormous amounts of data and promotes more internal and external interactions than in the past [1], [71]. Managing a business process by integrating technologies and data is needed [69], [72].

Models and metrics for measuring their complexity have been studied. The need for such models and metrics becomes imperative with wider efforts to transform, reengineer, and automate business processes [73]–[75]. However, extant business process metrics predominantly rely on one aspect of the business process such as the process structure [74], [76]–[78]. Thus, in this paper, we focus on developing and integrating business process models with an emphasis on tacit processes caused by interactions among human agents and between human agents and automated systems.

The rest of this chapter is structured as follows; in Section 1, we review various business process models and methods for business process reengineering. Next, in Section 2, we introduce the concept of tacit processes in a business process. Then, in Section 3, we propose a business process complexity model with stochastic dynamic (BPCSD) to analytically characterize them using Markov chains and estimate the expectation and variance of the processing steps and flow time. In Section 4, we propose a framework for reengineering and apply it to a real-world business process. Lastly, in Section 5, we discuss our findings and conclude with directions for future research.

4.1 Literature Review of Business Process Models and Reengineering

In many cases, business process developers, architects and analysts use tools such as unified modeling language (UML) and business process modeling and notation (BPMN) as the base for modelling and communicating business processes. In addition, other methodologies are used to support managing variants of business processes during execution. This paper addresses the opportunity to map business processes described in BPMN or UML into Markov Chain models to bring the stochastic dynamics dimension into the investigation of the underlying processes. In addition, the paper explores the proposed method application as a tool for business process reengineering.

4.1.1 Models and Metrics for Structured Business Process

The objective of modeling and measuring business processes is primarily to solve issues in their design and execution. Since this objective is the same or at least similar to software engineering's objective, most business process measurements originate from software engineering [73], [79]–[83].

Based on software engineering, Cardoso developed control-flow-complexity (CFC), inspired by McCabe's Cyclomatic Complexity. CFC measures the complexity of a given process by measuring decision nodes such as XOR, OR, and AND. This measure was originally developed to be used during the design stage of a process but can be used to evaluate the difficulty of implementing a given process or can be used as a metric to decide whether a given process has to be maintained or redesigned [80], [84]. The CFC metric measures how difficult it is to test a model, but lacks the ability to measure how difficult it is to understand the model. In order to overcome this shortcoming of CFC, several different models have been proposed. One model is to use the maximum nesting depth in addition to the CFC [74]. Another model is to apply cognitive weights as defined by Shao & Wang to business processes. Cognitive weights are applied to one business process modeling method, Yet Another Workflow Language (YAWL), and complexity is measured [85].

Also based on CFC, a metric called Cross Connectivity (CC) has been proposed. This metric is used to quantify the understandability of a model by measuring connection tightness between nodes in a business process. The CC provides meaningful information when it is combined with other workflow measurements [79]. Similarly, there have been attempts to find better workflow designs by measuring cohesion and coupling. A better workflow can be selected in terms of execution and understandability [81].

Besides business complexity metrics based on software engineering, some other complexity metrics can be found [86]. These metrics, again, focus on measuring the structural complexity of business process. Inspired by social network analysis, there is another metric which looks at a business process as a density problem. It considers the minimum and maximum number of arcs for a given business process [87]. In addition, there is a metric which measures how likely a business process is to generate errors. This approach was based on the idea that comprehensibility of a model is related to whether or not the process is prone to errors in the design phase of the business process [88].

4.1.2 Models for Semi-Structured Business Process

The popular use of process-aware information systems (PAIS) allows for collection and analysis of the details of business process execution. During business process execution, the process is not always executed in the way it was designed, and creates variability [89]–[93]. Since traditional methods like BPMN are not the best for modeling such types of complexity, efforts are made to model semi-structured business processes [92], [94]–[96].

As a result, new models are proposed, allowing for some degree of variability during execution. Some models integrate more than a modeling language. Others amend new features to existing modeling languages to capture the variability in business processes [89]–[91], [97], [98].

Others focus on using different methods to cope with variability during execution. One method is to measure variability by quantifying business process goals: processing costs, process duration and customer performance [99]. Chen-Yang proposes Business Process Analysis in Goal, Operations Methodology (BPA-GOMS). In this, complexity is measured in two attributes: the internal complexity of an activity and its interaction complexity [100]. The other modeling effort is based on information theory [76], [101]. Petri-net is also used to predict the next process based on the locations and external operations [91], [102] or the duration of process [103], [104].

Using Markov chains is another popular way to model the dynamics of a business process due to its stochasticity [77]. Many models based on Markov chains predict a future process or remaining process time although other methods such as a decision tree, support vector and non-parametric regression are also available [105]–[107]. For example, the event logs from PAIS are clustered by the sequence similarity. Then the Markov model is made for each cluster to predict the next task [108], [109]. Other approaches predict the likelihood of future tasks based on the instance data value and available documents [93].

4.1.3 Methods and Models for Business Process Reengineering

Since the idea of business process reengineering (BPR) has been proposed [110], [111], BPR, business process transformation, or business process improvement has been commonly used to describe any effort to change a business to improve its performance [112]–[115]. The studies report that these efforts radically change processes, technologies, and associated labor [113], [116], [117].

The studies of BPR initially focus on information technologies and their role in BPR since they are known to be prerequisite [112], [118]–[120]. As the concept of BPR becomes popular, studies focus on fundamentals issues of BPR such as BPR principles and theories [112], [121]–[123]. Furthermore, the BPR case studies by major businesses such as IBM, Kodak, and Ford can be found in the literature [124]–[127]. Despite success cases presented in the literature, the outcomes of many are not promising [128]–[130]. To overcome this, many studies focus on the obstacles of BPR. These studies point out that the efforts of executive managers and setting realistic and cost effective objectives are key to reengineering business processes [127], [131], [132]. Others study tools, techniques, and methods to support business process improvement. However, those efforts either focus on supporting a specific area or are scattered in diverse domains. There is still need for a complete framework for the BPR [115]. Vanwersch et al. propose the six areas where the methodological decision support is vital based on the van der Aalys classification in [133]: aim, actor, input, output, technique, and tool.

Various techniques and methodologies are suggested for the different stages of BPR. In the stage of understanding a business process, mathematical models, diagrammatic models, or simulation models are proposed. These models are used to analyze the performance or validate the business process [134]. In the stage of evaluating and changing a business process, similar models such as diagrammatic models and mathematical models are proposed but serve different purposes. The graphical models search for a better process by reducing or consolidating tasks whereas the mathematical models try to find a new design which optimizes a predefined objective such as cost or quality with constraints [134]–[136]. For example, a business process optimization problem treats a business process as a complex network with decision variables and an objective function which is subject to constraints [135], [136]. Besides, some variations of the typical optimization problem can be found in the literature such as a study based on performance evaluation, a study using decision models and a study using multi-objective evaluation [137]–[139].

Process mining is another way to understand, enhance and transform a business process. Often, event logs are analyzed to discover workflow patterns or identify the optimal workflow. For example, patterns were recognized from the events and the workflow was simplified in order to provide clear graphical representation and yield better results for process mining [140]. Others focused on automatically generating the optimal workflow from event logs. The workflow model was drawn from information stored in a spreadsheet [141]. Another automatically generated workflow diagram was based on process logs using Constraint Satisfaction Problem [68]. Similarly, the process tree is automatically generated using exhaustive, genetic, and greedy methods from event data [66]. Another approach to finding the optimal workflow, constraint programming, was used [142]. Integrating methods are also proposed. For example, in addition to the result of process mining, the Accimap model was used to analyze complaint handling processes [69]. In the domain of health care, other health care data along with the event logs was used to find better health care processes [72].

Since the goal of businesses is different depending on the firms and the case, the optimal processes should be discovered considering different aspects of a business.

4.2 Stochastic Dynamics Contributed by Tacit Processes

Although the primary purpose of using documented business processes is to describe actual processes, often, it does not represent their true execution [143]. Formally documented business processes may include all steps of the nominal processes. Tacit processes which often occur because of human nature, however, are neglected in the process of modeling the processes. In many cases, these informal processes are not logged in a transactional system because most of the time they are executed in a way that bypasses transactional systems such as exchanging emails and talking over phone. Although tacit processes are not explicitly described in a documented business model, including these are important in order to analyze the true workflow variability of a business process. We believe that the two main types of tacit processes when a human agent is involved in a business process are recheck and rework processes, which will be discussed next.

4.2.1 Recheck and Rework Tacit Processes

In a 'recheck' process, a human agent reprocesses or checks their work instead of passing the output to the next agent as shown in Figure **4-1**. This type of process often occurs when an agent needs to report their work to an agent who is in a higher rank or an agent who could not enter their output to a system due to system restriction or an error message.

The other type of tacit process is called a 'rework' process. This process is similar to the 'recheck' process, but the difference is in whether the process is repeated voluntarily or not. This tacit process is initiated by another human agent who receives the human agent output as shown in Figure **4-1**. This process often occurs when the process is between different divisions within an organization or between two human agents of different rank or role.

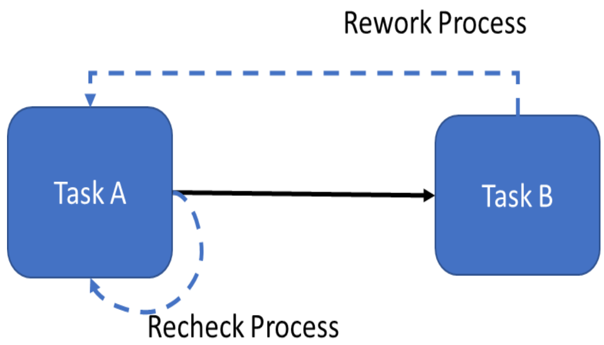


Figure **4-1:** Two Types of Tacit Processes

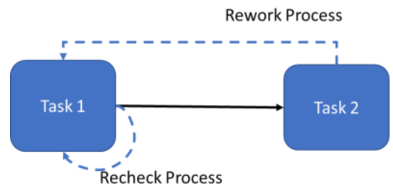
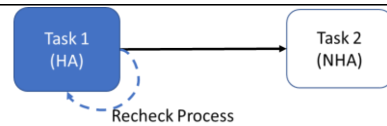
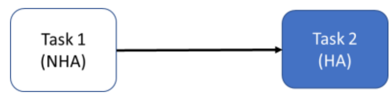
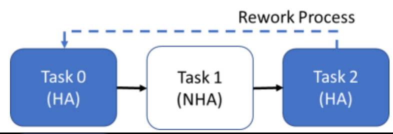
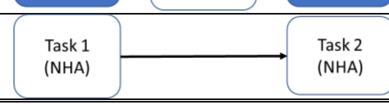
4.2.2 Interaction in Tacit Processes

Depending on the nature of the interaction, such as whether a human agent (HA) interacts with another human agent or a nonhuman agent (NHA), different types of tacit processes can be observed.

A ‘recheck’ process can be observed in the two types of interactions; an interaction between HAs and interaction between HA and NHA as shown in Case A and Case B of Table **4-1**.

Normally, a ‘rework’ process can be observed in only one type of interaction; an interaction between HA and HA as shown in Case A of Table **4-1**. However, in a special case, this type of tacit process can be observed in the interaction between NHA and HA as shown in Case D of Table **4-1**.

Table **4-1**:Tacit Processes in a Business Process

Case Type	Interaction Type	Associated Tacit Process	Visual Description
Case A	HA and HA	Recheck Process and Rework Process	
Case B	HA and NHA	Recheck Process	
Case C	NHA and HA	None	
Case D		Rework Process	
Case E	NHA and NHA	None	

4.3 Modeling Stochastic Dynamics of Business Process

We propose using Markov chains for analytically characterizing variability of a business process and estimate the expectation and variance of the processing steps and flow time. The Markov property implies that the probability distribution of the next state only depends on the current state, which in turn implies that in this business process model a future task only depends on the outcome of the current task. Moreover, the business process we model has a finite number of states since the number of its tasks is finite, and when the business process completes its execution, the Markov chain reaches an absorbing state.

Let e_i be the elements in a state, $S = (e_1, e_2, \ ... e_n)$ which represent each task or a combination of tasks in a business process. e_i takes a value of 1 if the task is being processed or waiting at task or activity *i*, and 0 otherwise. Thus, the number of states for any given business process is the number of tasks in a business process plus alpha depending on the structure of the business process. In addition, let a transition matrix P which represents a flow of the business process, has *t* transient states and r absorbing states as shown in Equation 4-1. For simplicity, the index column and row of transition matrix for BPCSD will be represented by task numbers or names instead of e_i as shown in Table **4-2**.

$$P = \begin{pmatrix} Q & R \\ 0 & I_r \end{pmatrix} \tag{4-1}$$

where Q is a t-by-t matrix, R is a nonzero t-by-r matrix, **0** is an r-by-t zero matrix, and I_r is the r-by-r identity matrix

Table **4-2**: Transitional Probability Matrix Example

$$P_{Example} = \begin{array}{r|c|c} & \text{Start} \quad \text{T1} \quad \ldots \quad \text{Tn} & \text{End} \\ \hline \begin{array}{r}\text{Start}\\ \text{T1}\\ \ldots\\ \text{Tn}\end{array} & Q & R \\ \hline \text{End} & 0 & I_r \end{array}$$

When business processes have a logical completion then the resulting BPCSD will have an absorbing state. In an absorbing Markov chain, the expected number of steps before being absorbed, ES and its variance, VAR can be expressed as shown in Equation 4-2 and Equation 4-3 [144]. Thus, the processing steps (PS) for a business process, the number of steps required from start to end of a business process, can be estimated by simply subtracting 1 from ES as shown in Equation 4-4.

$$ES = N1 \tag{4-2}$$

$$VAR = (2N - I_t)\mathrm{t} - t_{sq} \tag{4-3}$$

Where,
N is $N = (I_t - Q)^{-1}$,
1 is a length-t column vector whose entries are all 1
t_{sq} is the Hadamard product of t with itself
I_t is the t-by-t identity matrix. The (i, j) entry of matrix N is the expected number of times the chain is in state j, given that the chain started in state i

$$Processing\ Steps = ES - 1 \tag{4-4}$$

4.3.1 Estimating Flow Time of Business Process

Since the average and the variance of visits required by each task in a business process can be estimated using the Markov chain's property of absorbing another form of performance metric, the flow time can be estimated. Assuming that all the tasks' flow times in a business model are independent from each other and the number of visits to task i, and the time need to be spent in task i per visit are independent, then the expectation of the business process flow time (FT) is the sum of the expectation of each task's flow time as shown in Equation 4-5. Furthermore, the variance of the business process flow time also can be estimated as the sum of the variance of each task's flow time as shown in Equation 4-6 since the covariances are equal to 0 as the result of our assumption [145].

$$E(FT_{BP}) = \sum_{i=1}^{n} E(FT_i) = \sum_{i=1}^{n} E(T_i \cdot V_i) = \sum_{i=1}^{n} E(T_i)E(V_i) \tag{4-5}$$

$$Var(FT) = Var\left(\sum_{i}^{n} FT_i\right) = Cov\left(\sum_{i}^{n} FT_i, \sum_{j}^{n} FT_j\right) = \sum_{i}^{n}\sum_{j}^{n} Cov\big(FT_i, FT_j\big)$$
$$= \sum_{i}^{n} Var(FT_i) + 2\sum_{i<j}^{n} Cov\big(FT_i, FT_j\big) = \sum_{i}^{n} Var(FT_i)$$

Where, (4-6)

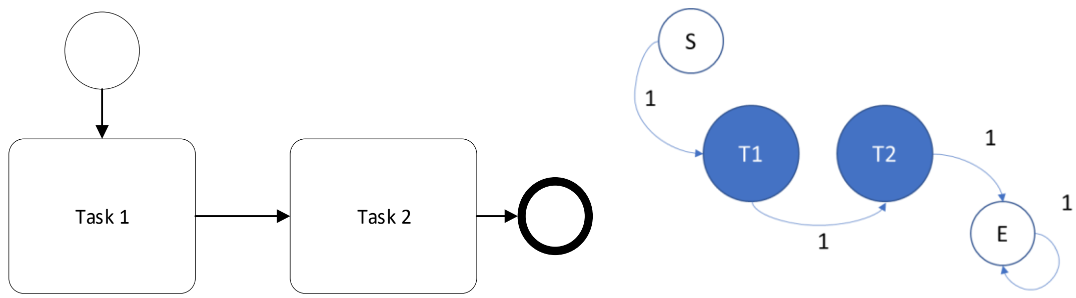
$$\begin{aligned} Var(FT_i) = Var(T_i \cdot V_i) \\ &= Cov\big(T_i^2, V_i^2\big) + [Var(T_i) + E(T_i)^2] \cdot [Var(V_i) + E(V_i)^2] \\ &- [Cov(T_i, V_i) + E(T_i) \cdot E(V_i)]^2 \\ &= Var(T_i)Var(V_i) + E(V_i)^2\, Var(T_i) + E(T_i)^2\, Var(V_i) \\ &+ E(T_i)^2 E(V_i)^2 - [E(T_i) \cdot E(V_i)]^2 \\ &= E(T_i)^2\, Var(V_i) + E(V_i)^2\, Var(T_i) + Var(T_i)Var(V_i) \end{aligned}$$

and,
n = number of tasks in a business process
T_i = time need to be spent in task *i per visit*
V_i = number of visits required to task *i,*

4.3.2 Mapping Business Process into Markov Chain

BPMN and Markov chain diagrams are very similar in that both use node and link. Mapping BPMN into a Markov chain, however, requires two steps: (1) defining states and (2) creating the transition probability matrix by assigning transition probabilities. One way to define this probability is by estimating probability from observing the current business process. However, in this model, we proposed to formulate the transition probability based on the structure of the business process as Cardoso defined in CFC metric if the real transition probabilities are unknown. In this way, BPCSD can consider not only the operational variability but also the structural variability of a business process. In the rest of this section, details of how to define the transition probability of different types of processes and calculate PS are explained.

1) Sequence Process



Sequence Process in BPMN Notation Sequence Process in Markov Chain

Figure **4-2**: Sequence Processes

The sequence process, Figure **4-2**, has only one subsequent path after Task 1(T1). Therefore, the transition probability from T1 to Task 2(T2) is 1 and the transition probability matrix, P_{Seq} is defined as Table **4-3**. Then the PS and its variance are 2 and 0 respectively.

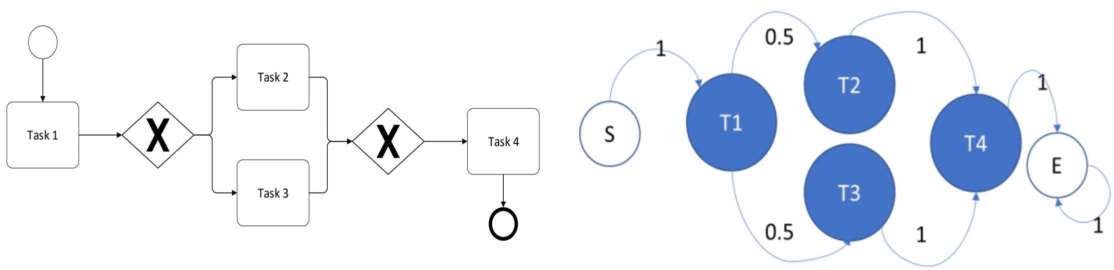
Table **4-3**: Transition Probability for a Sequence process

$P_{Seq} =$

	Start	T1	T2	End
Start	0	1	0	0
T1	0	0	1	0
T2	0	0	0	1
End	0	0	0	1

2) XOR Process

The XOR process is used when the business operates by selecting only one of the subsequent paths. The corresponding transition probability to each subsequent path can be calculated as $\frac{1}{number\ of\ subsequent\ paths}$ as Cardoso defined. In Figure **4-3**, after Task 1, the operation will be taken in either Task 2 or Task 3. Therefore, the probability assigned to 'from Task 1 to Task 2' path and 'from Task 1 to Task 3' path are 1/2 since only one of the subsequence paths will be selected. The transition probabilities for splits, however, can be modified appropriately if the path selection is not made uniformly or the actual path selection probability is known which makes the variability measure more practical and realistic. The transition probability metric, P_{XOR} is defined as Table **4-4**. Then the PS and its variance are 3 and 0 respectively.



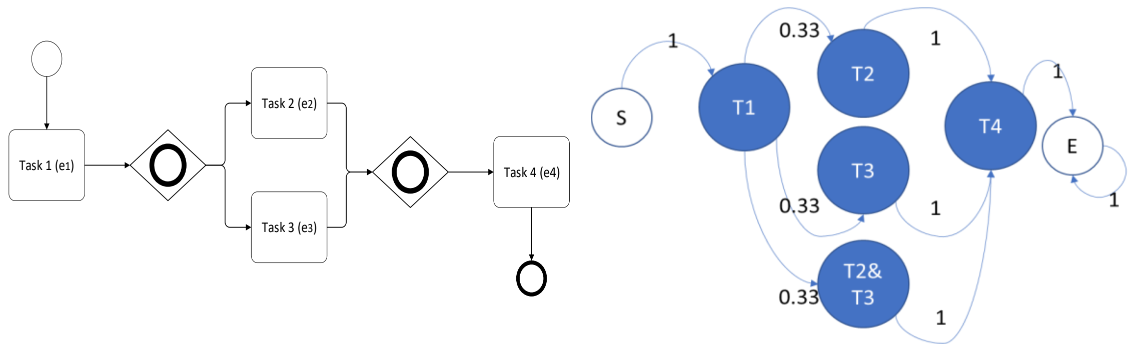
XOR Process in BPMN Notation XOR Process in Markov Chain
Figure **4-3**: XOR Process

Table **4-4**: Transition Probability for XOR Process

$P_{XOR} =$

	Start	T1	T2	T3	T4	End
Start	0	1	0	0	0	0
T1	0	0	1/2	1/2	0	0
T2	0	0	0	0	1	0
T3	0	0	0	0	1	0
T4	0	0	0	0	0	1
End	0	0	0	0	0	1

3) OR Process



OR Process in BPMN Notation   OR Process in Markov Chain

Figure **4-4**: OR Process

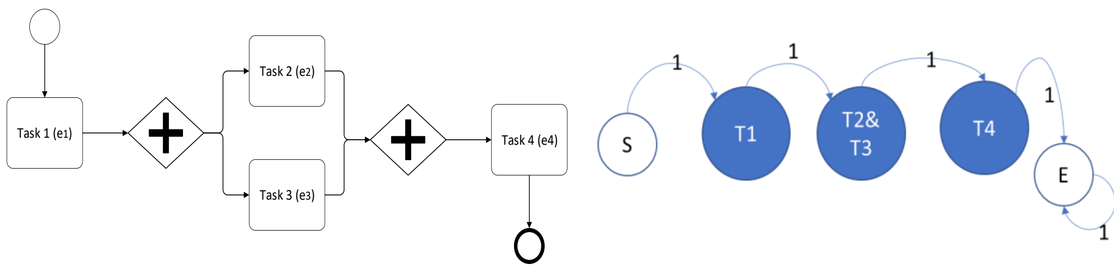
Unlike the XOR process, in the OR process, one or more subsequent paths can be selected depending on different conditions. The associated transition probability for each path can be defined as $\frac{1}{2^{\#\,of\,paths}-1}$. In Figure **4-4**, after Task 1, the subsequent process can be Task 2, Task 3 or Task 2&3. In this case, the transition probability from Task 1 to subsequent process is $\frac{1}{2^2-1} = \frac{1}{3}$. The transition probability metric, P_{OR} is defined as Table **4-5**. Then the PS is 3 and its variance are 3 and 0 respectively.

Table **4-5**: Transition Probability for OR Process

$P_{OR} =$

	Start	T1	T2	T3	T2T3	T4	End
Start	0	1	0	0	0	0	0
T1	0	0	1/3	1/3	1/3	0	0
T2	0	0	0	0	0	1	0
T3	0	0	0	0	0	1	0
T2T3	0	0	0	0	0	1	0
T4	0	0	0	0	0	0	1
End	0	0	0	0	0	0	1

4) AND Process



AND Process in BPMN Notation AND Process in Markov Chain

Figure **4-5**: And Process

The AND process is used when all subsequent paths are processed in parallel. In Figure **4-5**, after Task 1, both Task 2 and Task 3 will be processed, and Task 4 will be processed if and only if both Task 2 and Task 3 are processed completely. In this case the transition probability from Task 1 to Task 2 and Task 3 is 1 like the sequence process. However, the transition to Task 4 only exists when both Task 2 and Task 3 is completely processed. The transition probability metric, P_{AND} is defined as Table **4-6**. Then, the PS is 3 and its variance is 0.

Table **4-6**: Transition Probability for AND Process

$P_{AND} =$

	Start	T1	T2T3	T4	End
Start	0	1	0	0	0
T1	0	0	1	0	0
T2T3	0	0	0	1	0
T4	0	0	0	0	1
End	0	0	0	0	1

4.4 Framework for Business Process Reengineering with a UPS Maintenance Case Study

A business process is executed by the interactions between sub-processes that follow a predefined logic. Thus, the effect of the change in a sub-process propagates to other processes. It eventually affects the whole business process. Therefore, there is a need for supporting the BRP decision that considers the BPR effect on the whole business process.

Therefore, we propose a framework for a BPR decision. To validate the effectiveness of the BPCSD and the framework, we present a case study. This involves a maintenance service process for information technology infrastructure with a particular focus on uninterrupted power supply (UPS) maintenance, which the management considers a good representative of their assets which include voice and data networks, various servers, and a data center. The purpose of UPS is to provide backup power using its batteries for short periods of time when there are power outages or loss of power quality thereby protecting critical assets from adverse impact.

4.4.1 Framework for Process Reengineering

The proposed framework is composed of 3 layers: business process structure, business process analysis, and business process estimation as shown in Figure **4-6**. The first two layers serve as the input for the third layer.

The first layer, the business process structure, defines the flow and sequence of a current business process. In this case, the structure of the UPS maintenance process (UPSMP) is modeled in BPMN. The second layer, process data analysis, validates the structure of the UPSMP defined in the first layer, identifies tacit processes, and estimate the individual task processing time. In the case of UPSMP, the communication records are used. The third layer, business process analysis serves as the main model for the framework. The status of UPSMP is estimated using BPCSD with inputs from the other two layers. The estimation is also compared to the simulation result for the purpose of validation.

The managerial input layer provides information about the target goal, resource constraints, and operational constraints of the BPR. Based on this information, the 3rd layer generates options for BPR.

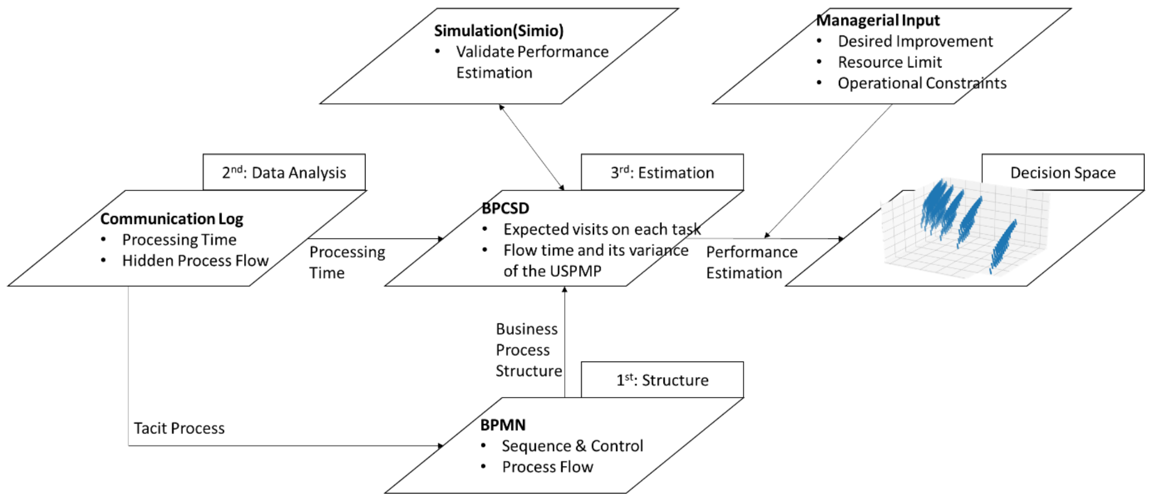


Figure **4-6**: Transformation Framework

4.4.2 1st layer: Business Process Structure

Figure **4-7** shows the UPS maintenance process (UPSMP) model that we developed based on interviewing managers and roughly reviewing incident logs. Although we realize that ‘T3: Maintaining’ is composed of sub-processes as shown in Figure **4-8**, details are often omitted in the communication records. We use, thus, Figure **4-7** which has 5 activities with 1 XOR split: (1) Starting work order, (2) Assigning technician, (3) Maintaining, (4) Incompletion/waiting, and (5) Final checking and reporting completion as our base for analyzing the workflow variability of UPSMP.

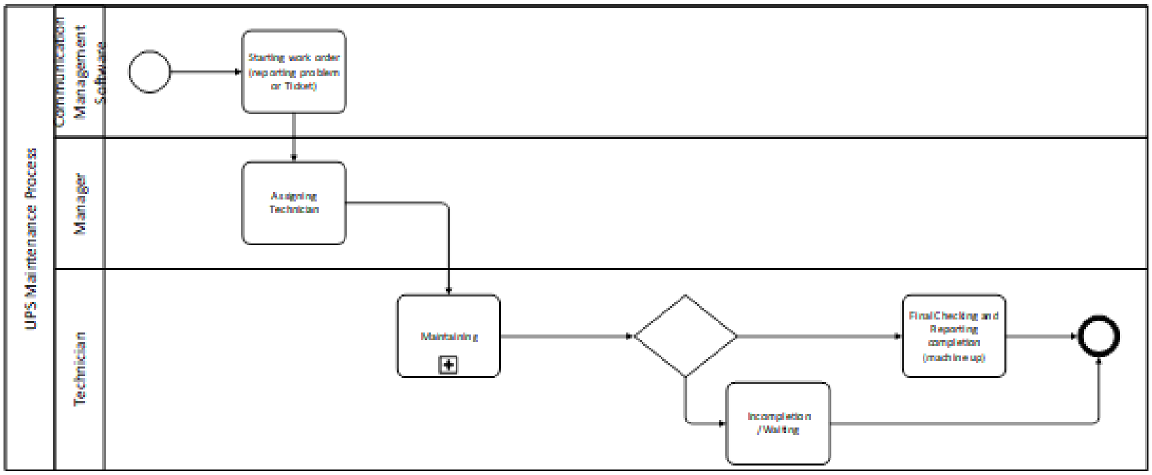


Figure **4-7**: UPS maintenance process

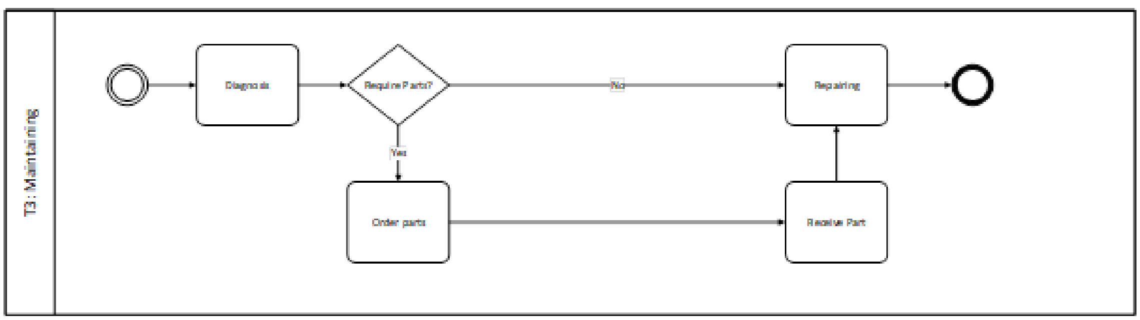


Figure **4-8**: Sub Process of T3: Maintaining

4.4.3 2nd Layer: Data Analysis of Communication Records

Since tacit processes are executed in a way that bypasses the transactional system, the tacit processes are derived from the communication records instead of the event or incident logs. The records cover 218 incidents over 28 months and contain self-notes and exchanged messages between actors.

Organizing the UPSMP communication records

The UPSMP communication records are merged into the UPSMP incident logs as shown in Figure **4-9**. The merged logs contain information such as the incident number, timestamp for the incident, resolution note, and communication records. Each incident can have multiple messages or/and notes. Thus, the merged logs are reorganized in a way that each log represents a message instead of an incident as shown in Figure **4-11**.

Number	Short Description	Parent Incident	Contact	Status	Assignment group	Created	Service Offering (Support of the Service)	Resolution Code	Resolution notes (these notes are not sent to the contact)	Comments and Work notes
INC09[redacted]4	Critical 10/01/2018 16:41:56 [redacted]e-UPS01 The UPS has no batteries attached. 17775-77		TNSANS API (tns_ans_api)	Closed	EIT-[redacted]-UPSTeam	10-01-2018 04:22:45 PM	Enterprise Network	Completed Successfully	Since both Incident Task INC0828591 and Change Request CHG0049970 are both closed, and there has been no complaints since, I am closing this Incident.	10-18-2018 10:34:52 AM - Ja[redacted]son [redacted] (Additional comments) Since both Incident Task INC0828591 and Change Request CHG0049970 are both closed, and there has been no complaints since, I am closing this Incident. 10-10-2018 01:05:45 PM - Ja[redacted]n [redacted] (Additional comments) I am re-assigning this Incident to EIT-[redacted]UPSTeam and to myself. Notes should be added to the attached Incident Task INT0012033. 10-10-2018 11:11:55 AM - Ja[redacted]ndo ([redacted] (Additional comments) UPS in question is a floor

Figure **4-9**: UPSMP incident logs + Communication records

Next, each log is classified into 5 tasks in UPSMP: Starting Work Order (T1), Assigning Technician (T2), Maintenance (T3), Incompletion/Waiting (T4), or Final Checking and Reporting Completion (T5). Before the classification, cleaning messages, and applying a machine learning algorithm may help, but a significant portion of classification requires manual effort since most of the messages are written without a standard format. Processing time for each log is calculated by taking the difference of the timestamp of the previous log and the timestamp of the current log as shown in Figure **4-10**. If there is no record for the starting work order, we use the timestamp for the incident creation as the timestamp for the starting work order. The final form of UPSMP communication logs is shown in Figure **4-11**.

Message	Message Time	Created	Incident Number	Classification	Time (MIN)
aaaa	1000-01-01 10:00	1000-01-01 10:00	INCXXXX1	Starting Work Order	0
bbbb	1000-01-01 11:00	1000-01-01 10:00	INCXXXX1	Assigning Technician	60 mins
ccccc	1000-01-01 14:00	1000-01-01 10:00	INCXXXX1	Maintenance	180 mins
dddd	1000-01-01 14:30	1000-01-01 10:00	INCXXXX1	Maintenance	30 mins
eeee	1000-01-01 16:00	1000-01-01 10:00	INCXXXX1	Final Checking and Reporting Completion	90 mins

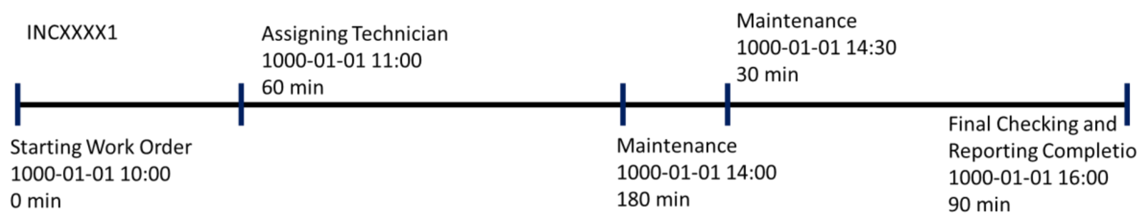


Figure **4-10**: Processing Time Calculation Example

Message	Message Time	Message From	Message Type	Created	Incident Number	TimeStampType	Time
Comment copied from Parent Incident: Update on West [redacted] outage web page. Customers affected are now down to 1-20. Customers Affected:	2017-06-18 18:13	[redacted] James [redacted]	Additional comme	2017-06-18 16:51	INC[redacted]23	Assigning technician	1.359
Work note copied from Parent Incident: called [redacted] said they are still investigating Said he would go into tomorrow am at 630 and check on switches. Al classes held at	2017-06-18 18:25	Craig [redacted]	Work notes	2017-06-18 16:51	INC[redacted]23	Maintenance	0.203
Icon for this device SV.[redacted]UPS01 is gone in [redacted]. There is also no record in [redacted].	2017-06-18 20:14	[redacted] James [redacted]	Additional comme	2017-06-18 16:51	INC[redacted]23	Maintenance	1.810
[redacted] Rayner ([redacted]), SV Sys Admin was contacted. He will go into tomorrow, Monday 6/19, at 630 AM and check on switches. Al classes held at [redacted] Building. Nobody on [redacted] on weekends.	2017-06-18 22:27	[redacted] James [redacted]	Additional comme	2017-06-18 16:51	INC[redacted]23	Maintenance	2.216
I spoke with [redacted] in Edge he added the the object back in [redacted] its up and responding to poll. [redacted] also was on site and checked the UPS he said it was online.	2017-06-19 7:59	Robert [redacted]	Work notes	2017-06-18 16:51	INC[redacted]23	Final Checking and Reporting Completion	9.549

Figure **4-11**: Final Form of UPSMP Communication Log

Calculating Processing Time for Each Task

As shown in Figure **4-11**, multiple maintenance attempts are made for this incident. These attempts are recorded as messages rather than events. As a result, without analyzing the communication records, the maintenance time for this incident is about 4.22 hours. However, we separate the maintenance into multiple maintenance events based on notes and messages; in this case, the maintenance takes about 1.40 hours on average.

From the UPSMP communication logs, the time spent on each task is estimated as shown in Table **4-7**. The distribution of each task's time is also identified although the p-values in a goodness-of-fit test are less than 10%.

Table **4-7**: Time needs to be spent on each UPSMP task

(unit: hrs)	Min	Max	Mean	Var	Distribution
T1: Starting Work Order	0.00	0.00	0.00	0.00	-
T2: Assigning Technician	0.00	6550.64	74.88	270884.39	Weibull(0.27,1.65)
T3: Maintaining	0.00	2218.08	49.16	26734.50	Weibull(0.36,9.94)
T4: Incompletion/Waiting	0.00	342.10	33.57	6366.28	Lognormal(1.30,2.34)
T5: Final Checking and Reporting Completion	0.00	8066.74	194.85	515894.23	Weibull(0.33,21.83)

Calculating Transition Probability for Tacit Processes

The UPSMP potentially has 7 tacit processes because all of the tasks are performed by human agents. All tasks can potentially have a 'recheck' process, and there can be two 'rework' processes: (1) between communication management software and Manager, and (2) between Manager and Technician. Since we assume that the same technician performs T3, T4, and T5, there is no re-work process that can be observed between T3 and T4 or T3 and T5.

If there are no tacit processes, there must be only one record for each task per incident in the communication records. Of 218 incidents, we identified that 13 incidents have more than 1 record of the T2 (Assigning Technician) and 161 incidents have more than 1 record of the T3 (maintenance) as shown in Table **4-8**. Two tacit processes are identified in USPMP as shown in Figure **4-12**.

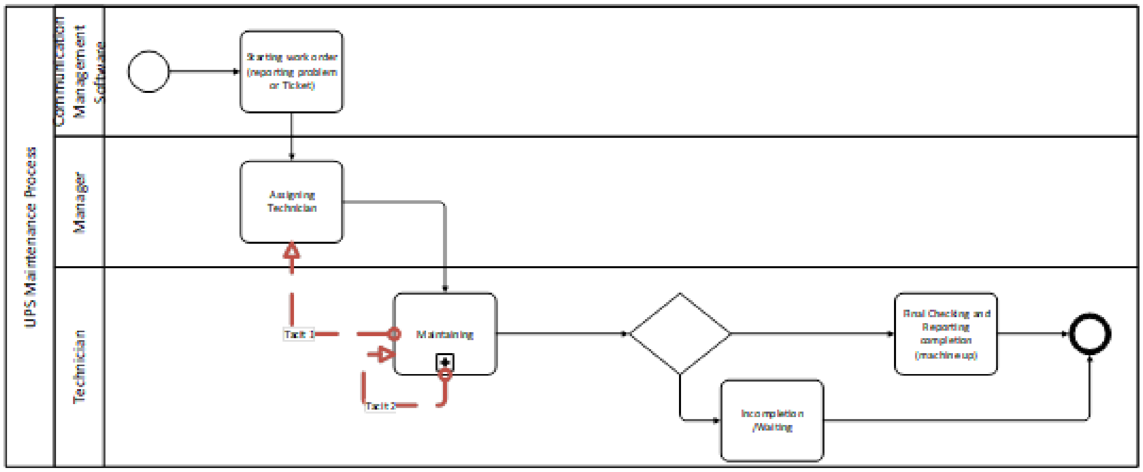


Figure **4-12**: UPS Maintenance Process with Tacit Processes

Table **4-8**: Summary of Tacit Processes in UPSMP

	Tacit 1	Tacit 2
Min. occurrence per incident	0	0
Max. occurrence per incident	2	29
Average occurrence	0.06	2.72
Number of Incidents with Tacit Processes	13	161

4.4.4 3rd layer: Business Process Estimation

As shown in the process data analysis, the presence of tacit processes adds variability to the business process; this, in turn, inflates the business process flow time. The stochastic dynamic impact of tacit process in UPSMP is analyzed using UPSMP and information obtained from the process analysis.

Table **4-9**: Transition Probability Matrix for UPSMP

P_{UPSMP}	Start	Task 1	Task 2	Task 3	Task 4	Task 5	End
Start	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Task 1	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Task 2	0.00	0.00	0.00	1.00	0.00	0.00	0.00
Task 3	0.00	0.00	0.00	0.00	0.50	0.50	0.00
Task 4	0.00	0.00	0.00	0.00	0.00	0.00	1.00
Task 5	0.00	0.00	0.00	0.00	0.00	0.00	1.00
End	0.00	0.00	0.00	0.00	0.00	0.00	1.00

Table **4-10**: Transition Probability Matrix for UPSMP with Tacit Processes

P_{UPSMP}	Start	Task 1	Task 2	Task 3	Task 4	Task 5	End
Start	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Task 1	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Task 2	0.00	0.00	0.00	1.00	0.00	0.00	0.00
Task 3	0.00	0.00	0.02	0.72	0.13	0.13	0.00
Task 4	0.00	0.00	0.00	0.00	0.00	0.00	1.00
Task 5	0.00	0.00	0.00	0.00	0.00	0.00	1.00
End	0.00	0.00	0.00	0.00	0.00	0.00	1.00

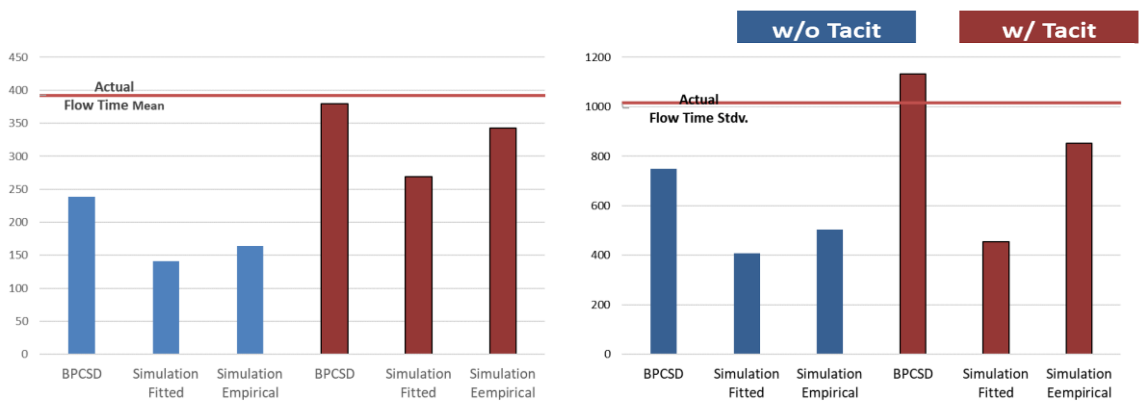
The UPSMP without tacit process, Figure **4-7** can be presented as the transition probability matrix represented in the following Table **4-9**. Since the process is a sequence without tacit processes, the expected number of visits to each task is 1 except Task 4 and Task 5. The expected number of visits to these tasks is 0.5 assuming that path selection is made evenly between two paths. On the other hand, the UPSMP with the tacit processes, Figure **4-12**, the tacit processes add stochastic dynamic to the workflow as shown in the transition probability matrix, Table **4-10**. In turn, the expected number of visits to Task 2, and Task 3 increases by .08 and 2.85, respectively, as shown in Table **4-11**.

Table **4-11**: Expected number of Visits to each task in UPSMP

	Task 1	Task 2	Task 3	Task 4	Task 5
Without tacit processes	1.00	1.00	1.00	0.50	0.50
With tacit processes	1.00	1.08	3.85	0.50	0.50

This impact of the tacit processes on the number of visits inflates, in turn, the flow time of UPSMP. Without tacit processes, the mean flow time and its standard deviation estimated by the BPCSD method are 238 and 749 hours, respectively. Whereas the mean flow time and the standard deviation increased by 61% and 73% becomes 383 and 1296 hours due to the tacit processes. Estimations by other methods show similar results.

The estimation is also compared to the actual flow time as shown in Table **4-12** and Figure **4-13**. If the model does not consider the tacit process, the models underestimate the mean by about 40% to 60%. However, if the models consider the tacit process, the difference between the models' estimations of mean and the actual mean is reduced by 37% on average. In both cases, the estimation of BPCSD is the closest to the actual mean compared to the other estimations.



(a) Mean Flow Time (b) Flow Time Standard Deviation

Figure **4-13**: Comparison between estimations of models and Actual Mean and Standard deviation

Table **4-12**: UPSMP Estimation Comparison

Tacit Processes	Model	Mean (hrs)	Difference to Actual Mean	Standard Deviation	Difference to Actual Stedv.
Without Tacit Processes	BPCSD	238.25	-39%	749.13	-26%
	Simulation (Fitted Distribution)	141.33	-64%	408.33	-60%
	Simulation (Empirical Distribution)	163.94	-58%	503.22	-50%

With Tacit	BPCSD	383.92	-2%	1296.22	28%
	Simulation (Fitted Distribution)	269.36	-31%	453.70	-55%
	Simulation (Empirical Distribution)	342.73	-13%	853.62	-16%
Actual		392.05	-	1014.86	-

4.4.5 Analyzing Business Process Transformation Impact

With the inputs from the UPSMP manager and estimation from BPCSD, we construct an efficient frontier that supports the managerial decision on the UPSMP improvement project. We consider the mean flow time, flow time variance, and cost.

Discovering Current Status of UPSMP

The business process we use for the improvement is similar to Figure **4-12** but without Task4, “Incomplete/waiting” as shown in Figure **4-14** since Task 4 is trivial in UPSMP.

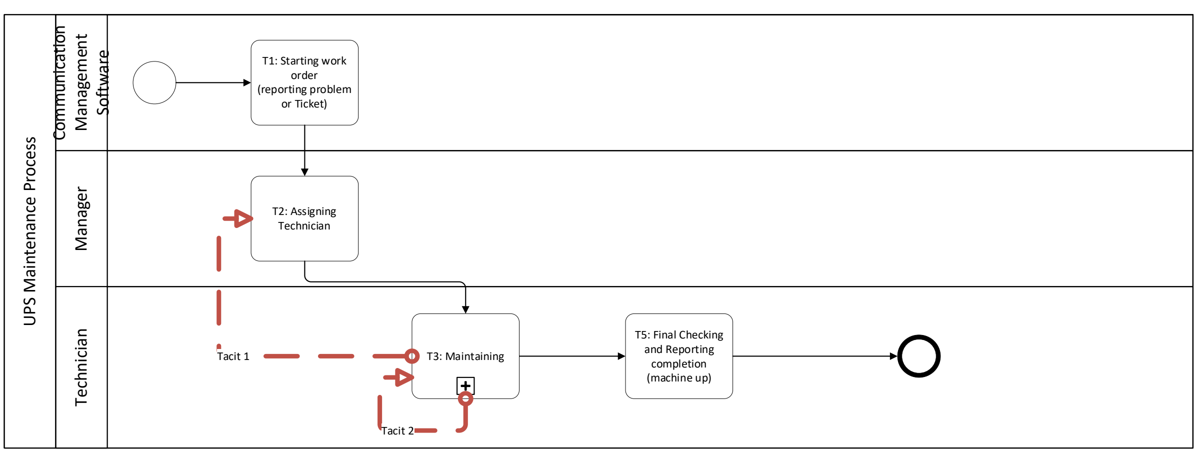


Figure **4-14**: USP Maintenance Process with 4 tasks

The UPSMP, Figure **4-14** currently requires 6.85 steps and 464.57 hours (with a standard deviation of 1,252) on average given that the expected number of visits to each task and processing time for each task as shown in Table **4-13** that are derived from the communication logs.

Table **4-13**: UPS Maintenance Process Performance Measures

Category		T1: starting work order	T2: assigning technician	T3: maintaining	T5: final checking and reporting completion
Number of Visits	Average	1.00	1.08	3.85	1.00
	Variance	0.00	0.08	10.95	0.00
Processing Time	Average	0.00	74.88	49.16	194.85
	Variance	0.00	270884.39	26734.50	515894.23

All Possible Options for USPMP Improvement

According to the parameters, no lean activities are necessary for T1 since its processing time is 0 and there is no variability in the number of visits to T1 and processing time of T1. All other tasks may provide opportunities for improvement.

Possible activities are suggested to the UPSMP managers as shown in Table **4-14** to promote their ideas and concerns. Selecting the details of activities and estimating their impact cost is impractical at the stage of preliminary planning for the business improvement project. Instead, desired improvement level and willingness to pay for executing improvements are estimated as shown in Table **4-15** considering their current resource availability and urgency.

From the manager’s input, we assume that the maximum improvements can only be achieved by spending the maximum willingness to pay and vice versa. The willingness pay is converted into dollars using the 2019 average weekly earnings of private service-providing employees, $975/week. We assume that any efforts to improve the processing time (IPT) of a task can reduce the variability up to 50% and any efforts cannot exceed the maximum desired reduction (MDR) level. The summary of UPSMP improvement options and associated costs are as shown in Table **4-16**.

$$IPT_a \leq MDR_a, \forall \text{ T2, T3, T5, R1, and R2}$$

$$IPTV_b \leq 50\%, \forall \text{ T2, T3, and T5}$$

Table **4-14**: Potential Activities for UPS Maintenance Process Improvement

Corresponding Task	Activity Description	Activity Type
T2: Assign Technician	Hiring an assistant	Increasing resources
	Adopting a system that automatically assigns technicians	Standardizing Operation Process
T3: Perform maintenance on UPS	Hiring more technicians for maintenance	Increase Resources
	Developing a standard maintenance process for training	Standardizing Operation Process
T4: Completion Report	Reporting completion automatically	Automating Process
	Setting a due date for the final report	Standardizing Operation Process
R1: Re-assigning	Adopting an assignment support system	Automating Process
R2: Re-doing	Adopting a diagnosis support system that suggests what needs to be replaced/repaired	Automating Process

Table **4-15**: Desired Improvement and Willingness to Pay for Transformation Activities

Task	Current Status			Desired Reduction (%)		Willingness to Pay (man hrs)	
	Time (hrs)			Max	Min	Max	Min
	Avg	Min	Max				
T2: Assign Technician	74	0	641	65%	35%	2 weeks	1 week
T3: UPS Maintenance	49	0	2218	20%	5%	2 weeks	1 week
T4: Completion Report	195	0	8066	20%	5%	4 weeks	3 weeks
Task	Current Status			Desired Reduction (%)		Willingness to Pay (man hrs)	
	Occurrence Frequency			Max	Min	Max	Min
R1: Re-assigning	2%			5%	1%	1 week	1 day
R2: Re-do	72%			50%	10%	4 weeks	3 weeks

Table **4-16**: Options for Improving UPS maintenance Process

Task	Improvement in Processing Time	Improvement in Standard Deviation	Cost	Task	Improvement in Frequency	Cost
Task 2 (T2)	30%	0% ~ 50%	975	Tacit 1 (R1)	1%	195
	40%	0% ~ 50%	1219		2%	390
	50%	0% ~ 50%	1463		3%	585
	60%	0% ~ 50%	1706		4%	780
	70%	0% ~ 50%	1950		5%	975
Task 3 (T3)	5%	0% ~ 50%	975	Tacit2 (R2)	10%	2925
	10%	0% ~ 50%	1300		20%	3169
	15%	0% ~ 50%	1625		30%	3413
	20%	0% ~ 50%	1950		40%	3656
Task 5 (T5)	5%	0% ~ 50%	2925		50%	3900
	10%	0% ~ 50%	3250			
	15%	0% ~ 50%	3575			
	20%	0% ~ 50%	3900			

Using the BPCSD model, the expected flow time and standard deviation of the flow time for each option, and the combination of options are estimated. The standard deviation of the flow time, the expected flow time, and the cost for each option are plotted as x-axis, y-axis, and z-axis as shown in Figure **4-15**.

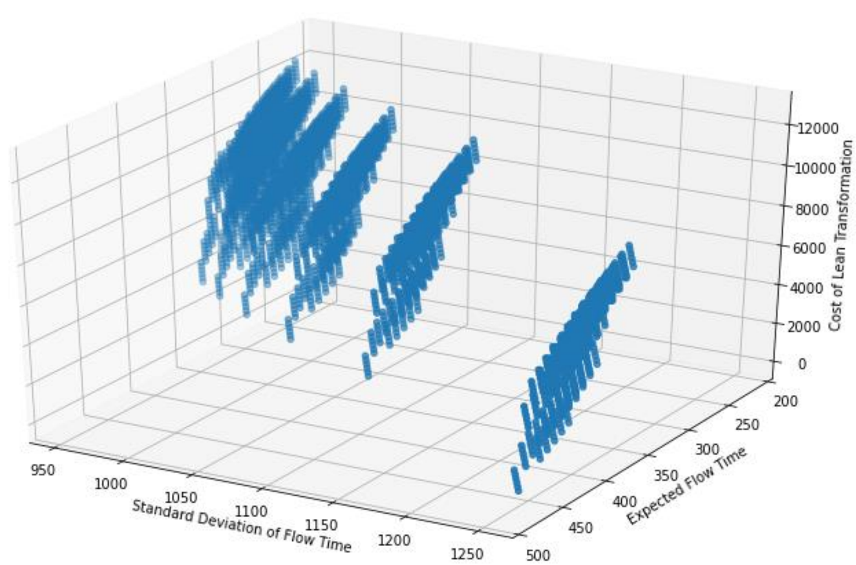


Figure **4-15**: Output of Lean transformation options

UPSMP Improvement Efficient Frontier

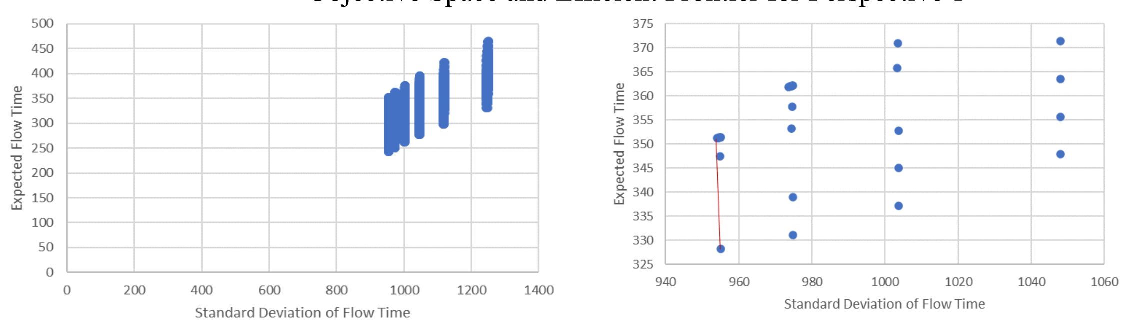
To identify operationally realistic options, the following constraints are assumed. Additions, omissions, and revisions of the following constraints are possible depending on the current operational constraints.

- The 20% reduction in the standard deviation of the flow time must be achieved
- The maximum number of activities that can be initiated at a time are two
- The maximum budget allowed for lean transformation is $5,000

We analyze all possible options in two different perspectives and identify a different efficient frontier per the perspective as shown in Figure **4-16**.

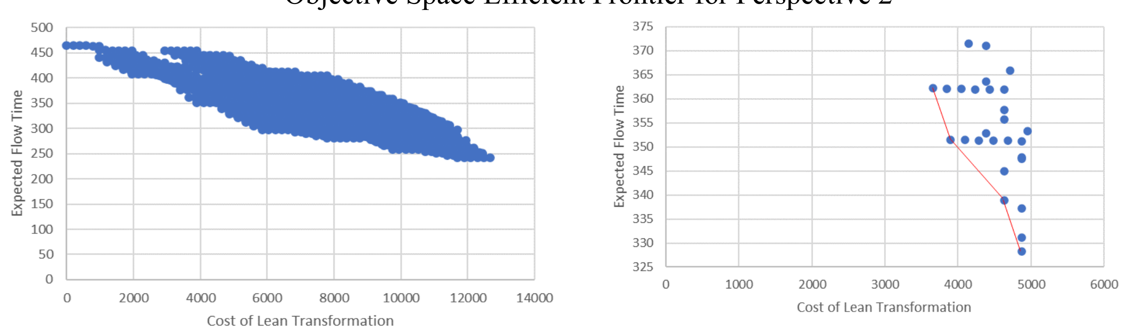
- Perspective 1: Minimize the flowtime and the variability of the flow time
- Perspective 2: Minimize the flowtime and the cost of lean transformation

Objective Space and Efficient Frontier for Perspective 1



a.i before applying constraints a.ii after applying constraints

Objective Space Efficient Frontier for Perspective 2



b.i before applying constraints b.ii after applying constraints

Figure **4-16**: Objective Space for the Lean Transformation

To find efficient options, we highlight the options in red as shown in Figure **4-16**. For this specific case, if the manager wants to find options that minimize the expected flowtime and the variability in the flow time, two options are indifferent as shown in Figure **4-16**.a.ii. Similarly, if the manager wants to find options that minimize the cost and the expected flowtime, the four options are indifferent as shown in Figure **4-16**.b.ii.

The summary of the result is as shown in Table **4-17**. In this case, regardless of the perspectives, Option A, is selected. In addition, all options suggest transforming Tacit 2.

Table **4-17**: Summary of Lean Transformation Options by the Perspectives

Options		Details of Lean Transformation Options	FT	Stedv	NFT	Cost
Options from Perspective 1	A	30% reduction in Task 2 50% reduction in Tacit 2	328	874~955	0.34~0.37	4875
	B	5% reduction in Tacit 1 50% reduction in Tacit 2	351	954	0.37	4875
Options from Perspective 2	A	30% reduction in Task 2 50% reduction in Tacit 2	328	874~955	0.34~0.37	4875
	C	30% reduction in Task 2 40% reduction in Tacit 2	339	894~975	0.35~0.38	4631
	D	50% reduction in Tacit 2	351	955	0.37	3900
	E	40% reduction in Tacit 2	362	975	0.37	3656

4.5 Summary and Future Work

In this chapter, the contribution of the tacit processes to the overall complexity of a business process and the new modeling method, BPCSD, is discussed. In addition, we discuss the possibility of the proposed model as a tool for business process reengineering by integrating the communication record, manger's input and model.

The shortcoming of BPCSD is that BPCSD requires business processes to be illustrated in BPMN or UML with a Markov chain. The ability to accurately estimate the flow time of a business process is one advantage of BPCSD and in the case study presented, the difference between the estimated flow time and the actual flow time was only 2%. The other advantage of BPCSD is that the process variability of a business process can be measured in terms of the steps required to complete the process and the flow time, whereas many other metrics are inadequate.

Another advantage of BPCSD is that it can be used as a tool for BPR. BPCSD can investigate the AS-IS process and estimates improvements in performance and variability of the TO-BE process to support the BPR decision. In the case study, we integrate data from communication records to BPCSD to estimate the performance of TO-BE options. Mangers' inputs are applied to select options. Selected options using BPCSD may possibly reduce the flow time by 22% ~ 29% and the standard deviation by 22 ~ 30%.

Like the variants of business processes, tacit processes create variability in business processes. However, discovering tacit processes is not apparent since they are executed in such a way that do not leave footprints on the PAIS or event logs. In this paper, we present a method to discover them from communication records and how it can be used for estimating and reengineering the current process. However, studies are warranted in the future to establish diverse ways to discover tacit processes in real-time and mitigate them in the business process design and reengineering phase.

Chapter 5

Conclusion

This dissertation investigates one of the challenges, the new interaction impact, to reengineering the maintenance process. New interactions can improve performance by reducing processing time or improving the accuracy of diagnosis but may introduce new variability in different areas in the maintenance process.

In Chapter 2, the possible impact on human workload are studied by conducting a controlled lab experiment focused on the interaction of a smart system and human. In the experiment, the difference in perceived workload between groups is statistically insignificant. The result of the experiment shows that there is a difference between AId-system group and FTd-system group in the time to complete maintenance. The FTd-system group completes the task 86.27 seconds (18% faster) before the AId-system group. However, further analysis implies that this difference is probably due to the difference in the system interface. Furthermore, if the AId-system limits the number of options shown per diagnosis attempt, it might improve the maintenance time. Since these findings are the result of a controlled lab experiment, further experiments in real-world environments are needed in order to accurately capture the impact of using a smart system on human workload. Also, this particular experiment is designed to investigate the impact of a smart diagnosis system on a technician who maintains a simple machine with a proximity sensor. Additional experiments are required for other settings such as different types of machines or the interaction with different types of smart systems.

In Chapter 3, the variability due to the number of diagnosis attempts is studied. A method of modeling maintenance time considering the experience effect of technicians using negative hypergeometric distribution is proposed. This modeling method estimates the maintenance time by focusing on the number of diagnosis attempts required. Furthermore, the model introduces the concept of *experience effect* that experienced technicians are able to perform maintenence on a machine with fewer diagnosis attempts by eliminating options to diagnose like a smart diagnosis support system. Besides, further analysis on the result of the experiment in Chapter 2 using the proposed model is conducted. The result of the NS group is similar to the estimation which assumes that technicians have a 20% understanding of the system. However, the outcome of groups that use the diagnosis support systems is predictable since they have less uncertainty in the expected number of diagnosis outcomes. Overall experiment results reveal that one of the benefits of using a diagnosis support system is lessening the performance variability in the diagnosis. The recommendation or instruction of the diagnosis support system ensures that maintenance is performed in a structured way regardless of the technicians who maintain the system. The proposed maintenance time modeling method has, however, limitations. Methods to quantify and model the experience effect need to be studied further. Also, the relationship between the experience in maintaining individual components such as maintaining a wire, light bulb, battery, and sensor and the experience in maintaining a system and its impact on maintenance time and performance variability needs to be investigated in future research.

In Chapter 4, the business process reengineering framework considering the variability impact on the maintenance process due to the interactions is proposed. In this chapter, tacit processes that cause unexpected variability in the workflow are defined and BPCSD, a method of modeling these, is proposed. To validate the proposed modeling method, UPS maintenance process case is presented. In the case study, the method of extracting tacit processes from communication logs is explained. Two tacit processes are identified from the communication records which cover 218 incidents over 28 months. The difference between the BPCSD estimation of mean and the actual mean is only 3%, if the impact of tacit processes is in the model. Furthermore, BPCSD is used as a tool for business process reengineering. It is used as a tool for evaluating the performance of To-Be options. The desired improvements and associated costs are provided by the managers who oversee UPS maintenance. All possible options' performance that does not violate the practical constraints are evaluated using BPCSD. Then, by constructing efficient frontiers, 2 or 3 indifferent options are selected depending on the goal of the managers.

Changes in workflow are inevitable as the maintenance process becomes smarter. Any change in a task changes how people work at the particular task and how people work with others in the maintenance process. These changes, in turn, not only introduce new opportunities for improvements but also new variability and workloads. Thus, the efforts to understand and investigate the impact on an individual task and the maintenance process are essential. Therefore, business process reengineering activities that work toward smarter maintenance must be carefully executed considering the impact on technicians and others in the process since the purpose of any smart system is meant to support performance of humans rather than present challenges and act as a barrier for them.

References

[1] T. Ruppert, S. Jaskó, T. Holczinger, and J. Abonyi, “Enabling technologies for operator 4.0: A survey,” *Appl. Sci.*, vol. 8, no. 9, pp. 1–19, 2018.

[2] M. M. Chui, Michael, Manyika James, “A Future That Works: Automation, Employment, and Productivity,” *McKinsey Glob. Inst.*, no. January, pp. 1–28, 2017.

[3] B. M. Chui, J. Manyika, and M. Miremadi, “Where machines could replace humans - and where they can’t (yet),” *McKinsey*, no. July, pp. 1–12, 2016.

[4] R. Kothamasu, S. H. Huang, and W. H. Verduin, “System health monitoring and prognostics - A review of current paradigms and practices,” *Handb. Maint. Manag. Eng.*, pp. 337–362, 2009.

[5] U. Neumann and A. Majoros, “Cognitive , Performance , and Systems Issues for Augmented Reality Applications in Manufacturing and Maintenance,” *IEEE Virtual Real. (VR ’98)*, pp. 4–11, 1998.

[6] M. P. Pacaux-Lemoine, D. Trentesaux, G. Zambrano Rey, and P. Millot, “Designing intelligent manufacturing systems through Human-Machine Cooperation principles: A human-centered approach,” *Comput. Ind. Eng.*, vol. 111, pp. 581–595, 2017.

[7] B. Shneiderman and B. Shneiderman, “Human-Centered Artificial Intelligence : Reliable , Safe & Trustworthy Human-Centered Artificial Intelligence : Reliable , Safe & Trustworthy,” *Int. J. Human–Computer Interact.*, vol. 36, no. 6, pp. 495–504, 2020.

[8] B. A. Kadir and O. Broberg, “Human-centered design of work systems in the transition to industry 4 . 0,” *Appl. Ergon.*, vol. 92, no. May 2020, p. 103334, 2021.

[9] L. Silvestri, A. Forcina, V. Introna, A. Santolamazza, and V. Cesarotti, “Maintenance transformation through Industry 4.0 technologies: A systematic literature review,”

Comput. Ind., vol. 123, p. 103335, 2020.

[10] R. Langer, J. Li, S. Biller, Q. Chang, N. Huang, and G. Xiao, “Simulation study of a bottleneck-based dispatching policy for a maintenance workforce,” *Int. J. Prod. Res.*, vol. 48, no. 6, pp. 1745–1763, 2010.

[11] M. Sheikhalishahi, L. Pintelon, and A. Azadeh, “Human factors in maintenance: a review,” *J. Qual. Maint. Eng.*, vol. 22, no. 3, pp. 218–237, 2016.

[12] R. R. Cabahug, D. J. Edwards, and J. Nicholas, “Classifying plant operator maintenance proficiency: Examining personal variables,” *Building Research and Information*. 2004.

[13] F. Pirola, X. Boucher, S. Wiesner, and G. Pezzotta, “Digital technologies in product-service systems: a literature review and a research agenda,” *Comput. Ind.*, vol. 123, p. 103301, 2020.

[14] T. T. Ademujimi, M. P. Brundage, and V. V. Prabhu, “A Review of Current Machine Learning Techniques Used in Manufacturing Diagnosis,” *IFIP Adv. Inf. Commun. Technol.*, vol. 513, pp. 407–415, 2017.

[15] S. Gheisari, M. R. Meybodi, M. Dehghan, and M. M. Ebadzadeh, “BNC-VLA: bayesian network structure learning using a team of variable-action set learning automata,” *Appl. Intell.*, vol. 45, no. 1, pp. 135–151, 2016.

[16] D. T. Nguyen, Q. B. Duong, E. Zamai, and M. K. Shahzad, “Fault diagnosis for the complex manufacturing system,” *Proc. Inst. Mech. Eng. Part O J. Risk Reliab.*, vol. 230, no. 2, pp. 178–194, 2015.

[17] Y. Huang, R. McMurran, G. Dhadyalla, and R. Peter Jones, “Probability based vehicle fault diagnosis: Bayesian network method,” *J. Intell. Manuf.*, vol. 19, no. 3, pp. 301–311, 2008.

[18] C. Wu, P. Jiang, C. Ding, F. Feng, and T. Chen, “Intelligent fault diagnosis of rotating machinery based on one-dimensional convolutional neural network,” *Comput. Ind.*, vol.

108, pp. 53–61, 2019.

[19] L. Yang and J. Lee, “Bayesian Belief Network-based approach for diagnostics and prognostics of semiconductor manufacturing systems,” *Robot. Comput. Integr. Manuf.*, vol. 28, no. 1, pp. 66–74, 2012.

[20] S. J. Hong, W. Y. Lim, T. Cheong, and G. S. May, “Fault detection and classification in plasma etch equipment for semiconductor manufacturing e-diagnostics,” *IEEE Trans. Semicond. Manuf.*, vol. 25, no. 1, pp. 83–93, 2012.

[21] M. Barakat, F. Druaux, D. Lefebvre, M. Khalil, and O. Mustapha, “Self adaptive growing neural network classifier for faults detection and diagnosis,” *Neurocomputing*, vol. 74, no. 18, pp. 3865–3876, 2011.

[22] M. Demetgul, “Fault diagnosis on production systems with support vector machine and decision trees algorithms,” *Int. J. Adv. Manuf. Technol.*, vol. 67, no. 9–12, pp. 2183–2194, 2013.

[23] A. Kumar, R. Shankar, A. Choudhary, and L. S. Thakur, “A big data MapReduce framework for fault diagnosis in cloud-based manufacturing,” *Int. J. Prod. Res.*, vol. 54, no. 23, pp. 7060–7073, 2016.

[24] M. Correa, C. Bielza, and J. Pamies-Teixeira, “Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process,” *Expert Syst. Appl.*, vol. 36, no. 3 PART 2, pp. 7270–7279, 2009.

[25] Z. Zhang, Y. Wang, and K. Wang, “Fault diagnosis and prognosis using wavelet packet decomposition, Fourier transform and artificial neural network,” *J. Intell. Manuf.*, vol. 24, no. 6, pp. 1213–1227, 2013.

[26] Y. W. Hsueh and C. Y. Yang, “Tool breakage diagnosis in face milling by support vector machine,” *J. Mater. Process. Technol.*, vol. 209, no. 1, pp. 145–152, 2009.

[27] G. Yu, C. Li, and S. Kamarthi, “Machine fault diagnosis using a cluster-based wavelet

feature extraction and probabilistic neural networks," *Int. J. Adv. Manuf. Technol.*, vol. 42, no. 1–2, pp. 145–151, 2009.

[28] S. Biswal and G. R. Sabareesh, "Design and development of a wind turbine test rig for condition monitoring studies," in *2015 International Conference on Industrial Instrumentation and Control, ICIC 2015*, 2015, pp. 891–896.

[29] I. Amihai, R. Gitzel, A. M. Kotriwala, D. Pareschi, S. Subbiah, and G. Sosale, "An industrial case study using vibration data and machine learning to predict asset health," in *Proceeding - 2018 20th IEEE International Conference on Business Informatics, CBI 2018*, 2018, vol. 1, pp. 178–185.

[30] N. Kolokas, T. Vafeiadis, D. Ioannidis, and D. Tzovaras, "Forecasting faults of industrial equipment using machine learning classifiers," in *2018 IEEE (SMC) International Conference on Innovations in Intelligent Systems and Applications, INISTA 2018*, 2018.

[31] A. Abu-Samah, M. K. Shahzad, E. Zamai, and A. Ben Said, "Failure prediction methodology for improved proactive maintenance using Bayesian approach," in *IFAC-PapersOnLine*, 2015, vol. 28, no. 21, pp. 844–851.

[32] F. Longo, L. Nicoletti, and A. Padovano, "Smart operators in industry 4 .0: A human-centered approach to enhance operators ' capabilities and competencies within the new smart factory context," *Comput. Ind. Eng.*, vol. 113, pp. 144–159, 2017.

[33] P. Fantini, M. Pinzone, and M. Taisch, "Placing the operator at the centre of Industry 4.0 design : Modelling and assessing human activities within cyber-physical systems," *Comput. Ind. Eng.*, vol. 139, no. February 2018, p. 105058, 2020.

[34] C. Cimini, F. Pirola, R. Pinto, and S. Cavalieri, "A human-in-the-loop manufacturing control architecture for the next generation of production systems," *J. Manuf. Syst.*, vol. 54, no. July 2019, pp. 258–271, 2020.

[35] D. Romero, O. Noran, J. Stahre, P. Bernus, and Å. Fast-Berglund, "Towards a human-

centred reference architecture for next generation balanced automation systems: Human-automation symbiosis," *IFIP Adv. Inf. Commun. Technol.*, vol. 460, pp. 556–566, 2015.

[36] C. Cimini, A. Lagorio, D. Romero, S. Cavalieri, and J. Stahre, "Smart Logistics and The Logistics Operator 4 . 0 Smart Logistics and The Logistics Operator 4 . 0," in *21st IFAC World Congress*, 2020, no. April.

[37] I. Brishtel, S. Ishimaru, O. Augereau, K. Kise, and A. Dengel, "Assessing Cognitive Workload on Printed and Electronic Media using Eye-Tracker and EDA Wristband," in *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, 2018, pp. 1–2.

[38] J. Hou, J. Rashid, and K. M. Lee, "Cognitive map or medium materiality? Reading on paper and screen," *Comput. Human Behav.*, vol. 67, pp. 84–94, 2017.

[39] Z. Zhan, L. Zhang, H. Mei, and P. S. W. Fong, "Online learners' reading ability detection based on eye-tracking sensors," *Sensors (Switzerland)*, vol. 16, no. 9, 2016.

[40] W. Choi and B. Tulu, "Effective Use of User Interface and User Experience in an mHealth Application," *Proc. 50th Hawaii Int. Conf. Syst. Sci.*, pp. 3803–3812, 2017.

[41] T. Jiang and H. Fang, "The influence of user interface design on task performance and situation awareness in a 3-player diner's dilemma game," *PLoS One*, vol. 15, no. 3, pp. 1–14, 2020.

[42] S. Yan, C. C. Tran, Y. Chen, K. Tan, and J. L. Habiyaremye, "Effect of user interface layout on the operators' mental workload in emergency operating procedures in nuclear power plants," *Nucl. Eng. Des.*, vol. 322, pp. 266–276, 2017.

[43] Q. Gao, Y. Wang, F. Song, Z. Li, and X. Dong, "Mental workload measurement for emergency operating procedures in digital nuclear power plants," *Ergonomics*, vol. 56, no. 7, pp. 1070–1085, 2013.

[44] S. Yan, Y. Wei, and C. C. Tran, "Evaluation and prediction mental workload in user

interface of maritime operations using eye response," *Int. J. Ind. Ergon.*, vol. 71, no. 145, pp. 117–127, 2019.

[45] H. Shin, K. Tien, and V. Prabhu, *Modeling the Maintenance Time Considering the Experience of the Technicians*. Springer International Publishing, 2019.

[46] S. K. Card, T. P. Moran, and A. Newel, "The KLM for User Performance Time with Interactive Systems," *Commun. ACM*, vol. 23, no. 7, pp. 396–410, 1980.

[47] D. Kieras, "Using the keystroke-level model to estimate execution times," *Univ. Michigan*, pp. 1–11, 2001.

[48] H. Shin, K.-W. Tien, and V. Prabhu, *Modeling the Maintenance Time Considering the Experience of the Technicians*, vol. 566. 2019.

[49] B. Dhillon, *Engineering Maintenance: A Modern Approach*. 2002.

[50] D. J. Edwards, J. Yang, R. Cabahug, and P. E. d. Love, "Intelligence and maintenance proficiency: An examination of plant operators," *Constr. Innov.*, 2005.

[51] F. Lolli, R. Gamberini, C. Giberti, M. Gamberi, M. Bortolini, and E. Bruini, "A learning model for the allocation of training hours in a multistage setting," *Int. J. Prod. Res.*, vol. 54, no. 19, pp. 5697–5707, 2016.

[52] T. P. WRIGHT, "Factors Affecting the Cost of Airplanes," *J. Aeronaut. Sci.*, vol. 3, no. 4, pp. 122–128, Feb. 1936.

[53] E. Pan, G. Wang, L. Xi, L. Chen, and X. Han, "Single-machine group scheduling problem considering learning, forgetting effects and preventive maintenance," *Int. J. Prod. Res.*, vol. 52, no. 19, pp. 5690–5704, 2014.

[54] M. Y. Jaber and A. L. Guiffrida, "Learning curves for imperfect production processes with reworks and process restoration interruptions," *Eur. J. Oper. Res.*, vol. 189, no. 1, pp. 93–104, 2008.

[55] C. Otto and A. Otto, "Extending assembly line balancing problem by incorporating

learning effects," *Int. J. Prod. Res.*, vol. 52, no. 24, pp. 7193–7208, 2014.

[56] R. R. Cabahug *et al.*, "Classifying plant operator maintenance proficiency : Examining personal variables," *Build. Res. Inf.*, vol. 32, no. 2, pp. 119–127, 2004.

[57] M. Jørgensen and D. I. K. Sjøberg, "Impact of experience on maintenance skills," *J. Softw. Maint. Evol. Res. Pract.*, vol. 14, no. 2, pp. 123–146, Mar. 2002.

[58] K. Tien and V. Prabhu, "Modeling the Influence of Technician Proficiency and Maintenance Strategies on Production System Performance," in *APMS 2018: Advances in Production Management Systems. Smart Manufacturing for Industry 4.0*, 2018, pp. 47–54.

[59] Y. Dallery, "On modeling failure and repair times in stochastic models of manufacturing systems using generalized exponential distributions," *Queueing Syst.*, vol. 15, no. 1–4, pp. 199–209, 1994.

[60] B. Liu, L. Cui, Y. Wen, and J. Shen, "A cold standby repairable system with working vacations and vacation interruption following Markovian arrival process," *Reliab. Eng. Syst. Saf.*, vol. 142, pp. 1–8, Oct. 2015.

[61] B. Liu, L. Cui, Y. Wen, and F. Guo, "A cold standby repairable system with the repairman having multiple vacations and operational, repair, and vacation times following phase-type distributions," *Commun. Stat. - Theory Methods*, vol. 45, no. 4, pp. 850–858, Feb. 2016.

[62] M. L. Yin, J. E. Angus, and K. S. Trivedi, "Optimal preventive maintenance rate for best availability with hypo-exponential failure distribution," *IEEE Trans. Reliab.*, vol. 62, no. 2, pp. 351–361, 2013.

[63] V. Mijailovic, "Probabilistic method for planning of maintenance activities of substation components," *Electr. Power Syst. Res.*, vol. 64, no. 1, pp. 53–58, 2003.

[64] D. Hu, Y. Cui, and A. Yin, "An Improved Negative Binomial Approximation for Negative Hypergeometric Distribution," 2013.

[65] W. J. Hopp, *Factory Physics*, 3rd ed. Long Grove, Ill. : Waveland Press, 2011.

[66] M. Arriagada-Benítez, M. Sepúlveda, J. Munoz-Gama, and J. C. A. M. Buijs, "Strategies to automatically derive a process model from a configurable process model based on event data," *Appl. Sci.*, vol. 7, no. 10, 2017.

[67] J. Becker, M. Rosemann, and C. von Uthmann, "Guidelines of Business Process Modeling," in *Business Process Management*, 2000, pp. 30–49.

[68] P. Wiśniewski, K. Kluza, and A. Ligęza, "An approach to participatory business process modeling: BPMN model generation using constraint programming and graph composition," *Appl. Sci.*, vol. 8, no. 9, 2018.

[69] Q. Wu, Z. He, H. Wang, L. Wen, and T. Yu, "A business process analysis methodology based on process mining for complaint handling service processes," *Appl. Sci.*, vol. 9, no. 16, 2019.

[70] K. Stachová, Z. Stacho, D. Cagánová, and A. Starecek, "Use of digital technologies for intensifying knowledge sharing," *Appl. Sci.*, vol. 10, no. 12, 2020.

[71] L. M. Camarinha-Matos, R. Fornasiero, J. Ramezani, and F. Ferrada, "Collaborative networks: A pillar of digital transformation," *Appl. Sci.*, vol. 9, no. 24, 2019.

[72] T. G. Erdogan and A. Tarhan, "A goal-driven evaluation method based on process mining for healthcare processes," *Appl. Sci.*, vol. 8, no. 6, 2018.

[73] J. Cardoso, "How to Measure the Control-flow Complexity of Web Process and Workflows," *Work. Handb. 2005*, pp. 199–212, 2005.

[74] V. Gruhn and R. Laue, "Complexity Metrics for Business Process Models," *Proc. 9th Int. Conf. Bus. Inf. Syst. (BIS 2006)*, no. September, pp. 1–12, 2006.

[75] Y. Diao and A. Keller, "Quantifying the complexity of IT service management processes," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4269 LNCS, pp. 61–73, 2006.

[76] J. Jung, C. Chin, and J. Cardoso, "An entropy-based uncertainty measure of process

models," *Inf. Process. Lett.*, vol. 111, no. 3, pp. 135–141, 2011.

[77] D. A. Karras and R. C. Papademetriou, "A Systematic Review of Analytical Management Techniques in Business Process Modelling for SMEs Beyond What-If-Analysis and Towards a Framework for Integrating them with BPM," in *BMSD 2017 - Proceedings of the 7th International Symposium on Business Modeling and Software Design*, 2017, pp. 99–110.

[78] V. Gruhn and R. Laue, "Approaches for business process model complexity metrics," *Technol. Bus. Inf. Syst.*, pp. 13–24, 2007.

[79] I. Vanderfeesten, H. A. Reijers, J. Mendling, W. M. P. Aalst, and J. Cardoso, "On a Quest for Good Process Models: The Cross-Connectivity Metric," vol. 5074, no. June, 2008.

[80] J. Cardoso, "Evaluating the process control-flow complexity measure," *Proc. - 2005 IEEE Int. Conf. Web Serv. ICWS 2005*, vol. 2005, no. Icws, pp. 803–804, 2005.

[81] I. Vanderfeesten, H. A. Reijers, and W. M. P. van der Aalst, "Evaluating workflow process designs using cohesion and coupling metrics," *Comput. Ind.*, vol. 59, no. 5, pp. 420–437, 2008.

[82] H. A. Reijers, "Software engineering and business process management: Interpretatio, imitatio, aemulatio," in *2013 IEEE 1st International Workshop on Communicating Business Process and Software Models Quality, Understandability, and Maintainability (CPSM)*, 2013, pp. 1–1.

[83] L. S. González, F. G. Rubio, F. R. González, and M. P. Velthuis, "Measurement in Business Processes: A Systematic Review," *Bus. Process Manag. J.*, vol. 16, no. 1, pp. 114–134, 2010.

[84] J. Cardoso, "Process control-flow complexity metric: An empirical validation," *Proc. - 2006 IEEE Int. Conf. Serv. Comput. SCC 2006*, pp. 167–173, 2006.

[85] V. Gruhn and R. Laue, "Adopting the Cognitive Complexity Measure for Business

Process Models," no. August 2006, 2006.

[86] L. Sánchez González, F. García Rubio, F. Ruiz González, and M. Piattini Velthuis, "Measurement in business processes: a systematic review," *Bus. Process Manag. J.*, vol. 16, no. 1, pp. 114–134, 2010.

[87] J. Mendling, "Testing Density as a Complexity Metric for EPCs," *Analysis*, 2006.

[88] J. Mendling and G. Neumann, "Error Metrics for Business Process Models," *Metrics Process Model.*, vol. 6, pp. 103–133, 2008.

[89] A. Delgado and D. Calegari, "Towards integrating BPMN 2.0 with CMMN and DMN standards for flexible business process modeling," in *XXII Ibero-American Conference on Software Engineering, CIbSE 2019*, 2019, no. April 2019, pp. 697–704.

[90] A. Delgado and D. Calegari, "BPMN 2.0 based modeling and customization of variants in business process families," in *2017 XLIII Latin American Computer Conference (CLEI)*, 2017, pp. 1–9.

[91] X. Zhu, G. Zhu, J. Vanthienen, and B. Baesens, "Enabling flexible location-aware business process modeling and execution," *Decis. Support Syst.*, vol. 83, pp. 1–9, 2016.

[92] M. Hewelt and M. Weske, "A Hybrid Approach for Flexible Case Modeling," in *BPM 2016: Business Process Management Forum*, 2016, pp. 38–54.

[93] G. T. Lakshmanan, D. Shamsi, Y. N. Doganata, M. Unuvar, and R. Khalaf, "A markov prediction model for data-driven semi-structured business processes," *Knowl. Inf. Syst.*, vol. 42, no. 1, pp. 97–126, 2015.

[94] W. M. P. Van Der Aalst, M. Weske, and D. Gru¨nbauer, "Case handling : a new paradigm for business process support," *Data Knowl. Eng.*, vol. 53, pp. 129–162, 2005.

[95] M. L. A. Rosa, W. I. L. M. P. V. A. N. D. E. R. Aalst, M. Dumas, and F. P. Milani, "Business Process Variability Modeling : A Survey," vol. 50, no. 1, pp. 1–45, 2017.

[96] R. M. De Carvalho, H. Mili, J. Gonzalez-huerta, A. Boubaker, and A. Leshob,

"Comparing ConDec to CMMN Towards a Common Language for Flexible Processes," in *2016 4th International Conference on Model-Driven Engineering and Software Development (MODELSWARD*, 2016, pp. 233–240.

[97] M. Pesic and W. M. P. Van Der Aalst, "A Declarative Approach for Flexible Business Processes Management," in *Business Process Management Workshop*, 2006, pp. 169–180.

[98] G. De Giacomo, M. Dumas, F. M. Maggi, and M. Montali, "Declarative Process Modeling in BPMN," in *Advanced Information Systems Engineering. CAiSE 2015*, 2015.

[99] B. Korherr and B. List, "Extending the EPC with performance measures," *Symp. Appl. Comput.*, pp. 287–294, 2007.

[100] C. Y. Cheng, D. Barton, and V. Prabhu, "The servicisation of the cutting tool supply chain," *Int. J. Prod. Res.*, vol. 48, no. 1, pp. 1–19, 2010.

[101] M. Ibl, "Uncertainty Measure of Process Models using Entropy and Petri Nets," pp. 542–547, 2013.

[102] S. Mandal and M. Weske, "A Flexible Event Handling Model For Business Process Enactment," in *IEEE 22nd International Enterprise Distributed Object Computing Conference*, 2018.

[103] A. Rogge-solti and M. Weske, "Prediction of business process durations using non-Markovian stochastic Petri nets," *Inf. Syst.*, vol. 54, pp. 1–14, 2015.

[104] A. Ferscha, "Qualitative and Quantitative Analysis of Business Work ows using Generalized Stochastic Petri Nets," *Connectivity*, no. October, pp. 222–234, 1994.

[105] B. F. van Dongen, R. A. Crooy, and W. M. P. Van Der Aalst, "Cycle Time Prediction : When Will This Case Finally Be Finished ?," in *On the Move to Meaningful Internet Systems: OTM 2008*, 2008, pp. 319–336.

[106] W. M. . van der Aalst, M. H. Schonenberg, and M. Song, "Time prediction based on process mining," *Inf. Syst.*, vol. 36, no. 2, pp. 450–475, 2011.

[107] M. Polato, A. Sperduti, A. Burattin, and M. De Leoni, "Time and activity sequence prediction of business process instances," *Computing*, vol. 100, no. 9, pp. 1005–1031, 2018.

[108] M. Le, D. Nauck, B. Gabrys, and T. Martin, "Sequential Clustering for Event Sequences and Its Impact on Next Process Step Prediction," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems. IPMU 2014. Communications in Computer and Information Science*, 2014, vol. 442.

[109] M. Le, B. Gabrys, and D. Nauck, "A hybrid model for business process event and outcomeprediction," *Expert Syst.*, vol. 34, no. 5, pp. 1–11, 2017.

[110] R. Kasa and A. Guban, "Business Process Amelioration Methods, Techniques, and Their Service Orientation: A Review of Literature," *Res. Decis. Sci. Glob. Bus.*, no. July 2016, pp. 219–238, 2015.

[111] T. H. Davenport and J. E. Short, "The new industrial Engineering: Information Technology and Business Process Redesign," *Sloan Manage. Rev.*, vol. 31, no. 4, pp. 11–27, 1990.

[112] M. Hammer and J. Champy, "Reengineering the Corporation: a Manifesto for Business Revolution," vol. 3, no. 4, pp. 76–79, 1993.

[113] J. Kubicová, "Reengineering – Concept and Review of Literature," 2015.

[114] W. Hamscher, "AI in Business-Process Reengineering," vol. 15, no. 4, pp. 71–72, 1994.

[115] R. J. B. Vanwersch and K. Shahzad, "A Critical Evaluation and Framework of Business Process Improvement Methods," *Bus. Inf. Syst. Eng.*, vol. 58, no. 1, pp. 43–53, 2016.

[116] J. Mechling, "Reengineering Government: Is There a 'There' There?," *Public Product. Manag. Rev.*, vol. 18, no. 2, pp. 189–197, 1994.

[117] J. a. Buzacott, "Commonalities in Reengineered Business Processes: Models and Issues," *Manage. Sci.*, vol. 42, no. 5, pp. 768–782, 1996.

[118] T. H. Davenport, *Reengineering Work through Information Technology*. 1993.

[119] P. Lillrank and S. Holopainen, “Reengineering for business option value,” *J. Organ. Chang. Manag.*, vol. 11, no. 3, pp. 246–259, 1998.

[120] P. S. Chan and C. Land, “Implementing reengineering using information technology,” *Bus. Process Manag. J.*, vol. 5, no. 4, pp. 311–324, 1999.

[121] W. B. Rouse, “A theory of enterprise transformation,” *Syst. Eng.*, vol. 8, no. 4, pp. 279–295, 2005.

[122] R. Abraham, J. Tribolet, and R. Winter, “Transformation of multi-level systems - theoretical grounding and consequences for enterprise architecture management,” *Lect. Notes Bus. Inf. Process.*, vol. LNBIP 146, pp. 73–87, 2013.

[123] V. Purchase, G. Parry, R. Valerdi, D. Nightingale, and J. Mills, “Enterprise transformation: Why are we interested, what is it, and what are the challenges?,” *J. Enterp. Transform.*, vol. 1, no. 1, pp. 14–33, 2011.

[124] T. W. Aurand, D. D. Schoenbachler, and G. L. Gordon, “Reengineering and the marketing function: integration of theory and practice,” *J. Prod. Brand Manag.*, vol. 5, no. 3, pp. 6–23, Jun. 1996.

[125] J. R. Caron, S. L. Jarvenpaa, and D. B. Stoddard, “Business Reengineering at CIGNA Corporation: Experiences and Lessons Learned from the First Five Years,” *MIS Q.*, vol. 18, no. 3, p. 233, 1994.

[126] M. J. J. Earl, J. L. L. Sampler, and J. E. E. Short, “Strategies for business process reengineering: Evidence from field studies,” *J. Manag. Inf. Syst.*, vol. 12, no. 1, pp. 31-57., 1995.

[127] G. Hall, J. Wade, and J. Rosenthal, “How to make reengineering really work,” *Harvard Bus. Rev.*, vol. 71, no. 6, pp. 119–132, 1993.

[128] M. Abdellatif, M. Salah, and N. Saeed, “Overcoming business process reengineering

obstacles using ontology-based knowledge map methodology," *Futur. Comput. Informatics J.*, vol. 3, no. 1, pp. 7–28, 2018.

[129] M. alhaji Musa, M. S. Othman, and W. M. Al-rahimi, "ONTOLOGY DRIVEN KNOWLEDGE MAP FOR ENHANCING BUSINESS PROCESS REENGINEERING," *Comput. Sci. Eng. An Int. J.*, vol. 3, no. 6, pp. 11–19, 2013.

[130] L. Rao, G. Mansingh, and K. Osei-bryson, "Building ontology based knowledge maps to assist business process re-engineering," *Decis. Support Syst.*, vol. 52, no. 3, pp. 577–589, 2012.

[131] E. K. Clemons, M. E. Thatcher, and M. C. Row, "Identifying Source of Reengineering Failures: A Study of the Behavioral Factors Contributing to Reengineering Risks," *J. Manag. Inf. Syst.*, vol. 12, no. 2, pp. 9–36, 1995.

[132] N. Abdolvand, A. Albadvi, and Z. Ferdowsi, "Assessing readiness for business process reengineering," *Bus. Process Manag. J.*, vol. 14, no. 4, pp. 497–511, 2008.

[133] W. M. . van der Aalst, "Business Process Management : A Comprehensive Survey," *ISRN Softw. Eng.*, p. 37, 2013.

[134] K. Vergidis, A. Tiwari, and B. Maieed, "Business Process Analysis and Optimization: Beyond Reengineering," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 38, no. 1, pp. 69–82, 2008.

[135] F. Soliman, "Optimum level of process mapping and least cost business process re-engineering," *Int. J. Oper. Prod. Manag.*, vol. 18, no. 9–10, pp. 810–816, 1998.

[136] I. Hofacker and R. Vetschera, "Algorithmical approaches to business process design," *Comput. Oper. Res.*, vol. 28, no. 13, pp. 1253–1275, 2001.

[137] H. Han, "Business Process Change Design from Decision Model Perspective," *Mangement Sci. Financ. Eng.*, vol. 9, no. 2, pp. 21–46, 2003.

[138] Yonghua Zhou and Yuliu Chen, "Project-oriented business process performance

optimization," in *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483)*, vol. 5, pp. 4079–4084.

[139] T. Jaeger, A. Prakash, and M. Ishikawa, "A framework for automatic improvement of workflows to meet performance goals," in *Proceedings Sixth International Conference on Tools with Artificial Intelligence. TAI 94*, pp. 640–646.

[140] M. Ramos-merino, L. M. Álvarez-sabucedo, F. De Arriba-pérez, and J. M. Santos-gago, "A Pattern Based Method for Simplifying a BPMN Process Model," *Appl. Sci.*, vol. 9, no. 11, 2019.

[141] P. Wiśniewski, K. Kluza, E. Kucharska, and A. Ligeza, "Spreadsheets as interoperability solution for business process representation," *Appl. Sci.*, vol. 9, no. 2, 2019.

[142] L. Parody, M. T. Gómez-López, A. J. Varela-Vaca, and R. M. Gasca, "Business process configuration according to data dependency specification," *Appl. Sci.*, vol. 8, no. 10, 2018.

[143] T. Savickas and O. Vasilecas, "Business process event log use for activity sequence analysis," *2015 Open Conf. Electr. Electron. Inf. Sci. eStream 2015 - Proc.*, pp. 1–4, 2015.

[144] C. M. Grinstead, *Introduction to Probability*, 2nd ed. American Mathematical Society, 1997.

[145] G. Schay, *Introduction to Probability with Statistical Applications*. 2016.

VITA

Hyunjong Shin

Education

Ph.D. in **Industrial Engineering** at Pennsylvania State University, January 2012 ~ Present.
Thesis title: Analytical and Empirical Models for Process Transformation towards Smarter Maintenance

M.S. in **Industrial Engineering** at Pennsylvania State University, 2013.
Thesis title: Analytical and Simulation Models for Evaluating Cash-Flow Bullwhip in Supply Chains

B.S. in **Supply Chain Management and Information Systems** at Pennsylvania State University, 2009.

Academic Experience

[Spring & Summer 2020] *Teaching/Grading Assistant* for the Department of Industrial and Manufacturing Engineering, at Pennsylvania State University, University Park, PA

[Spring 2018] *Course Developer* for BS/MBA 2019 Spring Seminar (SC 497) at Pennsylvania State University, Eberly College of Science, University Park, PA, USA

[Fall 2017&2018] *Lead Teaching Assistant* for Introduction to Supply Chain Management (SCM301) at Pennsylvania State University Smeal College of Business, University Park, PA

Professional Experience

[February 2014 ~ March 2017] *Senior Researcher & Project Leader* at SJ Strategy Institute, Gwangju, South Korea

- Consulted on the policy or plan of Korean government & public sector R&D projects
- Managed administrative duties of projects in charge

[Summer 2013] *Management Support Team Intern* at *Samsung (Hanhwa) Techwin, C*hangwon, South Korea

- Supported SCM consultants to establish an integrated SCM System in Samsung Techwin business process
- Translated the Samsung Techwin SCM Korean manual to English for users who do not understand Korean

Publications

Shin H., Tien K., Prabhu V. (2019) Modeling the Maintenance Time Considering the Experience of the Technicians. Production Management Systems. Production Management for the Factory of the Future. APMS 2019. IFIP Advances in Information and Communication Technology, vol 566.

Shin, H., & Prabhu, V.V., (2018) Evaluating Impact of AI on Cognitive Load of Technicians During Diagnosis Tasks in Maintenance. Advances in Production Management Systems. Smart Manufacturing for Industry 4.0. APMS 2018. IFIP Advances in Information and Communication Technology, vol 536.

Tangsucheeva, R., **Shin, H.**, & Prabhu, V. (2014). Stochastic Models for Cash-Flow Management in SME. In Encyclopedia of Business Analytics and Optimization (pp. 2288-2298). IGI Glob