2018-01-01

A Leveling Production Strategy To Automotive Assembly Using Queuing Systems Software Through Simio Simulation

Ivan Arturo Renteria Marquez
University of Texas at El Paso, iarenteria@utep.edu

A LEVELING PRODUCTION STRATEGY TO AUTOMOTIVE ASSEMBLY USING QUEUING SYSTEMS SOFTWARE THROUGH SIMIO SIMULATION

IVAN ARTURO RENTERIA MARQUEZ

Master's Program in Manufacturing Engineering

APPROVED:

____
Tzu-Liang (Bill) Tseng, Ph.D., Chair

____
Amit Lopes, Ph.D., Co-chair

____
Yirong Lin, Ph.D.

____
Charles Ambler, Ph.D.
Dean of the Graduate School

Dedication

This work is dedicated to my family.

A LEVELING PRODUCTION STRATEGY TO AUTOMOTIVE ASSEMBLY USING QUEUING SYSTEMS SOFTWARE THROUGH SIMIO SIMULATION

by

IVAN ARTURO RENTERIA MARQUEZ, PhD ECE

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Industrial, Manufacturing and Systems Engineering

THE UNIVERSITY OF TEXAS AT EL PASO

December 2018

Acknowledgements

I would like to thank my master thesis adviser Dr. Tzu-Liang (Bill) Tseng for all his support, both financial and scientific.

Abstract

The automotive manufacturing industry is challenged with an unpredictable costumer demand with considerable fluctuation. In addition, the number of products (vehicles) and the complexity of them are constantly growing. These characteristics make very difficult to plan and schedule production. This thesis presents a methodology to model with a high degree of accuracy the production floor, warehouse and material handling system of an automotive assembly facility through Simio simulation software; resulting in a dynamic model that combines all the critical variables of the system. The presented methodology includes an algorithm developed with the goal of modeling the automotive facility Kanban system. A hypothetical case study of an automotive manufacturing assembly plant is used as an example to show the method. In order to plan the production batch size, a Heijunka analysis was conducted. The presented model could be used by the automotive manufacturing assembly industry as a tool to develop the planning and scheduling strategy.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Simulation of queuing systems (SQS) techniques appeared about 40 years ago, but the implementation of simulations as a strategic technology has been barely used because of the limitation of hardware and simulation software during the last decades. The SQS most used commercial software by industry around the world is Arena simulation software. This software appeared in 2000 and was the best option during the last decade, but this software is limited in terms of the options to develop a model with a particular logic and in terms of the animation options. Nowadays, commercial computers are powerful enough to handle big amounts of data. In addition, new SQS commercial software appeared such as Simio simulation software. This software includes the four discrete modeling paradigms developed for SQS, which are *events, processes*, *objects* and *agent-based* modeling [1].

The purpose of this thesis is to present a methodology to model with a high degree of accuracy an automotive assembly plant, including the production floor, warehouse and material handling system through Simio simulation software. It will be shown how the presented model could be used by the automotive industry as a helpful tool when taking decisions related with planning production.

The motivation of this work is the lack of simulation implementations by most of the actual manufacturing companies. As explained by G. Michalos in [2], manufacturing research is still based mostly in experimentation, but the prognostic for the future includes the use of simulated models during the early phase of the manufacturing plants.

Modern automotive manufacturing enterprises are challenged due to unpredictable costumer demand with considerable fluctuation and a constant growing in complexity and number of products. The personnel in charge of the decision making process needs to face the complexity

of the system (i.e. the manufacturing facility and the supply chain) and the variable customer demand. Due to this constraints, the system must to be flexible to satisfy the product demand [3].

Two lean manufacturing tools that are key elements to have a flexible pull system are Kanban and Heijunka. Because of that, the methodology presented here includes a Kanban and Heijunka implementations.

This thesis is organized as follows. In section 1.1 a literature review of simulations related with the automotive industry, Heijunka, Kanban, planning and scheduling is presented. Chapter 2 presents a description of the automotive manufacturing assembly plant and the simulation approach of the automotive manufacturing assembly plant, Kanban based material handling system and the warehouse of raw material. Finally, Chapter 3 contains the numerical results, conclusions and a summary of the findings.

1.1 Literature review

The automotive manufacturing industry has been moving from a mass production basis to a customization basis. At the very beginning, Henry Ford started with a narrow number of models and colors, but nowadays most of the brands offer a wide number of different options. Due to the complexity of the product (vehicles), this customization of the automobile industry represents a big challenge for automobile manufacturers and the supply chain.

In addition, the customer demand is unpredictable and globalization makes the market very competitive. Thus, the automotive manufacturing industry must have a flexible production system with a quick response capable to react to the continuously variable market.

1.1.1 General automotive simulations

This subsection presents a summary of simulation implementations for the automotive industry. In [4] a data-driven simulation of an automotive manufacturing plant was proposed. The

goal of this work was to automatically generate a simulation of the manufacturing plant, including a feedback of the current state of the real system. Another example of a data-driven simulation for automotive assembly plants can be found in [5]. In this reference, the National Institute of Standards and Technology provides an extensive description of the basic processes in an automotive assembly manufacturing plant.

Patel describes in [6] a discrete event simulation of the finial process system of an automotive plant. The impact of the throughput (i.e. vehicles) as a function of the number of operators, mechanical repair stations and paint repair stations was analyzed with the proposed simulation.

One can find in [7] an interesting application of an automotive assembly workstation simulation that involves a lift assists machine. The goal of this approach was to protect operators predicting ergonomic risks in a new manufacturing area. Ref. [8] presents an implementation of value stream mapping and simulation of an assembly line of clutch discs. The authors reported that a combination of value stream mapping with simulation is a useful tool for the decision-making team of the automotive company.

1.1.2 Heijunka

Heijunka (leveling production) as a lean manufacturing tool has two different but related meanings. One meaning is to level the production by volume, and the second one is to level the production by product type (i.e. mix), refer to [9]. Heijunka is an important tool that helps to eliminate mura (unevenness). At the same time, eliminating mura is the first step to eliminate muri (overburdening people or equipment) and muda (waste), refer to [10]. In other words, Heijunka is the action of leveling the production of a manufacturing system by reducing the ups and downs of work in process (WIP) caused by the fluctuation demand [11].

This lean manufacturing tool is used mostly in high volume production systems. Nevertheless, one could level the production in low volume and high mix using the clustering techniques presented in [12].

A trading off between Heijunka and just-in-sequence is presented in [13]. In this reference, simulation was used to analyze Hejunka as an option for a small motor assembly line. Nevertheless, the changeover time and material handling system were not considered as part of the model. Other simulations of manufacturing systems with Heijunka approaches can be found in [14,15]. In addition, one can find in [16-19] survey papers focused on methodologies to balance production in a variety of manufacturing industries.

1.1.3 Kanban

Kanban (signboard) is a lean manufacturing tool that works as a communication system [20]. This communication system controls the production quantity in every process with the goal of pulling parts (i.e. raw material, WIP, and finished goods) when required [21]. Kanban-systems are usually called pull-systems because material is pulled through the system only when the authorizing signal (Kanban) is received [22].

A Kanban can be a variety of things, like a card, a marked space, an electronic message, etc. One can find in references [3, 19, 22, 23 and 24] implementations of simulations with Kanban-systems. For instance, Sternatz presents in [19] an analysis of the relationship between the line balancing and the material handling system and how the planning strategy needs to consider this interdependency. Reference [24] presents an interesting methodology used to simulate an assembly line in a Kanban system supplied by operators that transport raw materials from the supermarket to the point of use. In this reference, a probability density function was derived to simulate the stochastic demand.

Besides the Kanban signal, one can find in the literature references that analyses the advantages and disadvantages of using kitting and line stocking as part feeding strategies in the automotive assembly industry [25].

1.1.4 Planning and scheduling

In one hand, planning refers to the action of identifying the products that need to be manufactured and the required raw materials. This process is typically determined without a model of the company. In the other hand, scheduling refers to the action of doing a detailed timetable with the products that need to be manufactured, where critical variables such as the required raw materials, on-time delivery, required work force and machines are considered in the generation of the schedule. This process needs to be determined with a detailed model of the company, refer to [26].

The decision-making process in manufacturing systems is divided into four stages, which are design, planning, scheduling and control [27]. Frantzen [28] states that the algorithms and methods developed so far for researchers have little impact in industry, this is because the scenarios used to develop these scheduling methods highly differ from the real industry scenarios. Job shop scheduling problem is the most common scheduling scenario addressed by researchers [29]. Nevertheless, automotive assembly plants do not belong to this category.

The number of references focused in the area of planning and scheduling for the automotive manufacturing assembly industry is limited. For instance, Agnetis [30] proposed a group of dispatching rules in an attempt to solve scheduling problems related with the delivery of raw material in an automotive assembly plant. In [31], a just in time (JIT) scheduling approach focused on the delivery of raw material for automotive assembly lines was investigated. Nevertheless, one can find in the literature some interesting references that deals with problems

related with planning and scheduling for manufacturing companies in a variety of fields. Boysen presents in [32, 33] a summary of planning strategies such as mixed-model sequencing, car sequencing, and level scheduling. In [34] the planning strategy was used as a methodology to prevent ergonomic risks.

Chapter 2: Description of the automotive manufacturing assembly plant and solution approach

The automotive manufacturing assembly plant used as a hypothetical case-study in this manuscript is based on the general manufacturing assembly plant described in [4]. The production floor of this plant is formed by nine sections: four sections of trim, two sections of chassis, one line that assembles the doors, and one final assembly section. Where these sections are composed by 18, 16, 16, 11, 10, 12, 9 and 4 workstations, respectively. This is a flexible manufacturing line that produces several different vehicle models. A total number of 96 workstations forms these nine sections where workers perform the different operations. The WIP is moved by conveyors through the whole system.

The warehouse of raw materials has sixteen vehicles with operators that deliver the material at the point of use. A total number of 202 parts (raw material) is considered into this system. It is assumed that an operator located at the first workstation of the first section of trim is in charge of generating the Kanban signal. The operator receives the production order electronically, and then he calculates the amounts of material and sends an electronic signal to the warehouse. According to the vehicle model being produced at a specific moment in time and the batch size, he needs to specify the quantity and the delivery location of the required raw materials. A schematic of the automotive manufacturing assembly plant is shown in Fig. 2.1.

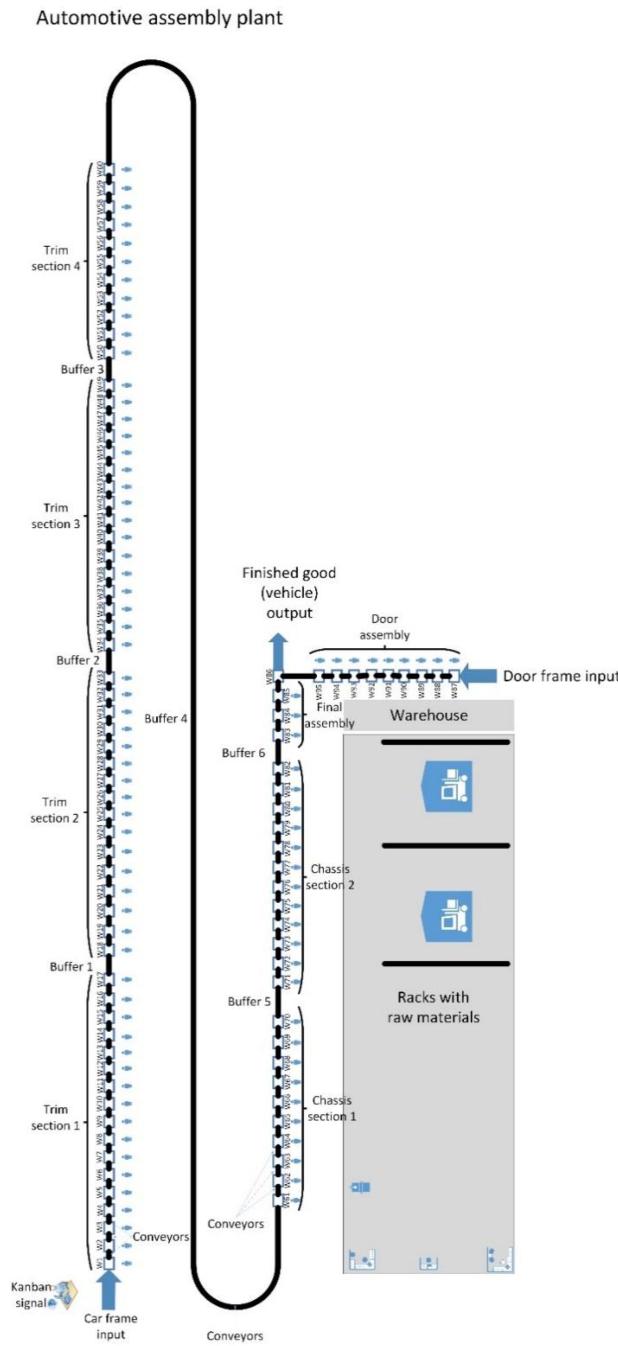

Figure 2.1: Automotive manufacturing assembly plant.

2.1 The solution approach. Simulation of the automotive manufacturing assembly plant.

An important step a simulation engineer needs to do when working with SQS is to choose the proper software to do the task. Simio simulation software was chosen for this work because it offers the four discrete modeling paradigms developed for SQS with a focus on *intelligent objects* [1]. This software provides a new object based paradigm that has been successfully implemented in systems like airports, hospitals, ports, mining, call centers, supply chains and manufacturing. In addition, this software can be coupled with other platforms such as MATLAB [35] to add custom heuristic optimization algorithms such as the one presented in [36].

Subsequently, the logical model being simulated needs to be defined. The defined system is composed by 96 workstations that forms the trim section 1, trim section 2, trim section 3, trim section 4, chassis section 1, chassis section 2, door assembly and final assembly sections.

The WIP is moved through the assembly process using conveyor belts. Inside the sections, the buffer capacity is defined as zero, which means that the workstations will be blocked if the workstation doing the next operation is still processing WIP. Six buffers are defined after each section. Failures of equipment and times to repair it are considered in the simulation model.

An operator is in charge of monitoring the amount of raw material, calculating and sending the Kanban signal requesting the raw materials to the warehouse. This is a flexible manufacturing line that produces two different vehicle models. Two different vehicle models are assembled in this manufacturing line with a changeover time after each vehicle model change. Fig. 2.2 shows a simplified diagram of this logic model.

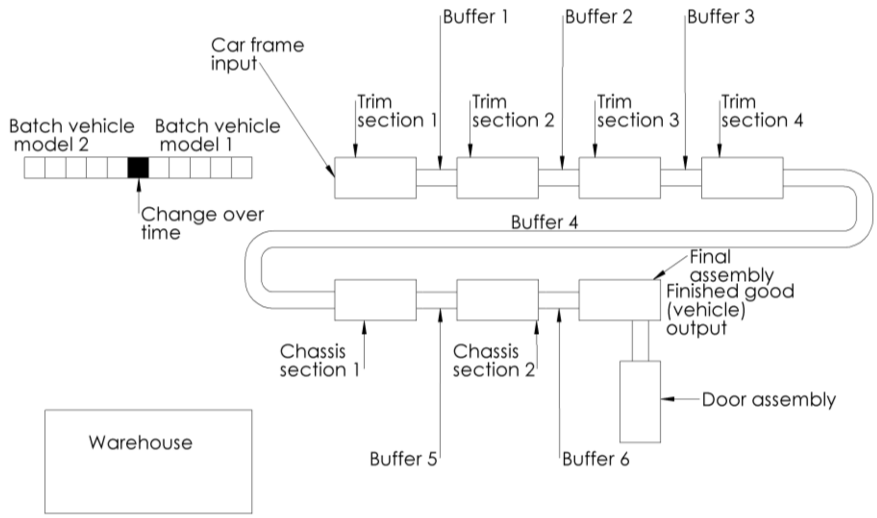

Figure 2.2: Logical simulation model.

2.2 Production floor. The simulation approach.

After the logical model was defined, one can program the simulation in the commercial software. Simio simulation software offers two object options to simulate the physical workstations: *server* and *workstation*. The Simio *server* object could be used to represent a service operation or a machine. The *workstatio*n object offers more options; with this object one can model setups, processing and teardown operations. In addition, one can call secondary resources (e.g. workers and raw materials). Since the *workstatio*n object offers more options, the workstations of the manufacturing plant are modeled with these objects.

A total of 96 *workstations* objects are used to form the trim, chassis, door, and final assembly sections. Each *workstation* object has three queues used to contain the units waiting to be processed (input buffer), the units being processed (processing buffer), and the units waiting to move to the next process (output buffer).

The *ranking rule*, *processing time*, *buffer logic*, *failure type*, *uptime between failures*, and *time to repair* options of the *workstation* objects where defined accordingly, refer to Fig. 2.3. In

this case the *ranking rule* is used to prioritize the *entities* with the smallest value. The *processing time* defines the time that takes to process the *entity* at a specific *workstation*. *Buffer logic* defines the capacity of the three queues of each *workstation*. *Failure type* is a tool used to select the method to define the time between failures (i.e. *calendar time based*, *event count based*, *no failures*, *processing count based* and *processing time based*). The *uptime between failures* is used to define the time between failures and *time to repair* is an option used to define the time required to repair a failure when it occurs.

The WIP produced by the *workstations* is moved using *conveyor* objects. The *travel capacity*, *entity ranking rule* and the desired *speed* of the *conveyor* objects where defined accordingly, refer to Fig. 2.4. The *travel capacity* defines the maximum number of entities allowed to be moved by the *conveyor* at the same time. *Entity ranking rule* is used to prioritize entities with the smallest value, and the desired *speed* is used to define the speed of the entities being move by the *conveyor*.

A *source* object was defined to create the *entities* that represent the orders of automobiles being produced by production floor. Since it is intended to simulate two different vehicle models, two different entities were defined: vehicle model one and model two. This means that this *source* produces two types of *entity* objects. The flow through floor of these two vehicle models is different. Because of that, a *sequence table* was defined for each *entity* type. These *sequence tables* were specified at the *entity* level. In order to avoid jumps between orders of the different models, a specific priority value was assigned to the different orders.

A total of seven buffers are defined to store the WIP. These *buffers* are located between the different sections forming the assembly plant. No *buffers* are defined inside of the different *workstations*. As an example imagine two internal *workstations* of a section, if the first

workstation finishes the assembly of a vehicle but the next *workstation* is still working with an assembly, the *conveyor* will not move and the *workstation* will be blocked until the second *workstation* releases its WIP.

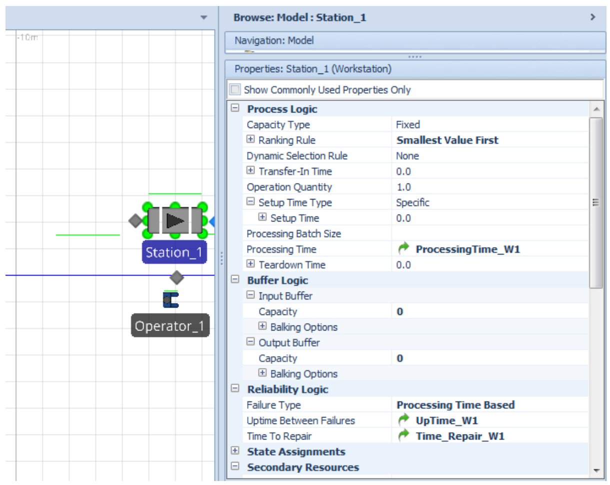

Figure 2.3: Properties of *workstation* object.

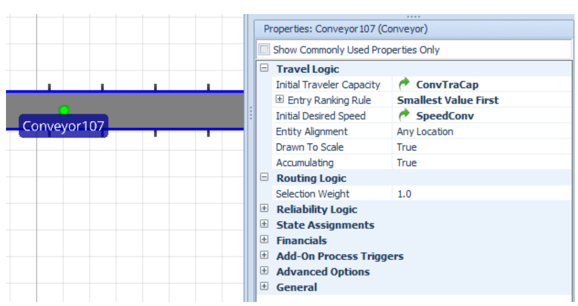

Figure 2.4: Properties of *conveyor* object.

2.3 Kanban based material handling system and warehouse of raw material. The simulation approach.

As explained in section 2.1, an operator located at the first workstation of the first section is in charge of generating the Kanban signal that request the raw materials to the warehouse. In order to simulate the operator generating the Kanban signal, an algorithm was developed and embedded into the simulation using the *Processes* tool offered by Simio. In [1], *Processes* are defined as a flowchart that uses steps to change the states of the simulation elements.

In other words, the decisions made by the operator in charge of the Kanban signal are programed with the presented algorithm. One can summarize the operations of this person (i. e. logical model) as follows:

1) Check for the number of orders of the current batch waiting to start the production process.

2) Decide if the number of orders of the current batch is smaller than the specified threshold value. Where the threshold value is the minimal number of reaming frames waiting to start the assembly process used as a reference for the operator as the reorder point.

3) Decide which vehicle model needs to be produced next, and generate the Kanban signal that asks for the frames of these vehicles. After that, a priority value is assigned to this batch of frames.

4) Calculate the quantity of material required to assembly the next batch of vehicles and send the signal to the warehouse. Where the calculation of the Kanban signal is a function of the replenishment time, production rate and the container size. The equation used to calculate the amount of raw material is presented below.

In addition, the presented algorithm is used to simulate the changeovers suffered after each change of vehicle model.

The developed algorithm is shown in Fig. 2.5. This algorithm was defined in Fig. 2.5 for two vehicle models, but it can be extended to be used for any number of vehicle models. The steps of this algorithm are:

Search. First, a *Search* step is called, this instruction returns the remaining *entities* (number of frames) in queue waiting to start the assembly process.

Decide. Is the number of entities in the first queue equal to the threshold value? A *Decide* step determines if the number of entities at the previously mentioned queue is equal to the threshold value. If this step is false then the algorithm ends and if this step is true then a second *Decide* step is called.

Decide. Was the vehicle model two produced the last time? This *Decide* step decides which model needs to be produced next. If vehicle model one was produced the last time then this step is false, and the branch of the algorithm used to produce vehicle model two is activated. If the step is true then a third *Decide* step is called.

Decide. Is the maximum number of batches for vehicle model one reached? The maximum number of batches for vehicle model one was previously defined using an *Integer property*. If the maximum number of batches for vehicle model one was reached then the algorithm ends. If the maximum number of batches for vehicle model one is still not reached then a create step is called.

Create. This *create* step is used to generate the batch of *entities* that represent the new frames of vehicle model one entering the system.

Assign. The *state variable* named *ModelEntity.Priority* (default variable provided by the software) is used to assign a priority value to the current created batch.

Transfer. This step is used to define the location of the entities created by the *Create* step.

Delay. This *Delay* step is used to define the change-over of the production line when a change of product occurs.

Fire. This *Fire* step is used to generate an *event* that makes the simulated Kanban operator to calculate the amount of raw materials, then he sends a request to the warehouse to have ready the raw materials.

One could embedded the following Kanban equation into the algorithm:

$$kanban = \frac{(Replenishment\ Time\ x\ Production\ Rate)\ (1+Alpha)}{(Container\ Size)} \quad (1)$$

where $Alpha$ is the safety factor, refer to [20]. In the warehouse, the operators of the sixteen vehicles pick up the raw materials and distribute them at the points of use.

Fire. This *Fire* step is used to generate an *event* that sends a request to the sixteen operators of the warehouse to collect the raw materials and deliver them at the point of use. The operators receive the material quantity and the delivery locations.

Assign. This *Assign* step is used to update a custom defined variable that is used to keep track of the last generated model.

Assign. This *Assign* step is used to update the value of the *ModelEntity.Priority* variable. The algorithm ends after the execution of this step.

In order to simulate the action of the Kanban operator checking for the level of raw material at the point of use, elements *Source*, *Entity* and *Sink* where added. The frequency of the generated *entities* represents the frequency at which the operators checks for the raw material level. Every time an entity enters the *Sink*, an *Event* is generated, and this event triggers the Kanban based material handling algorithm.

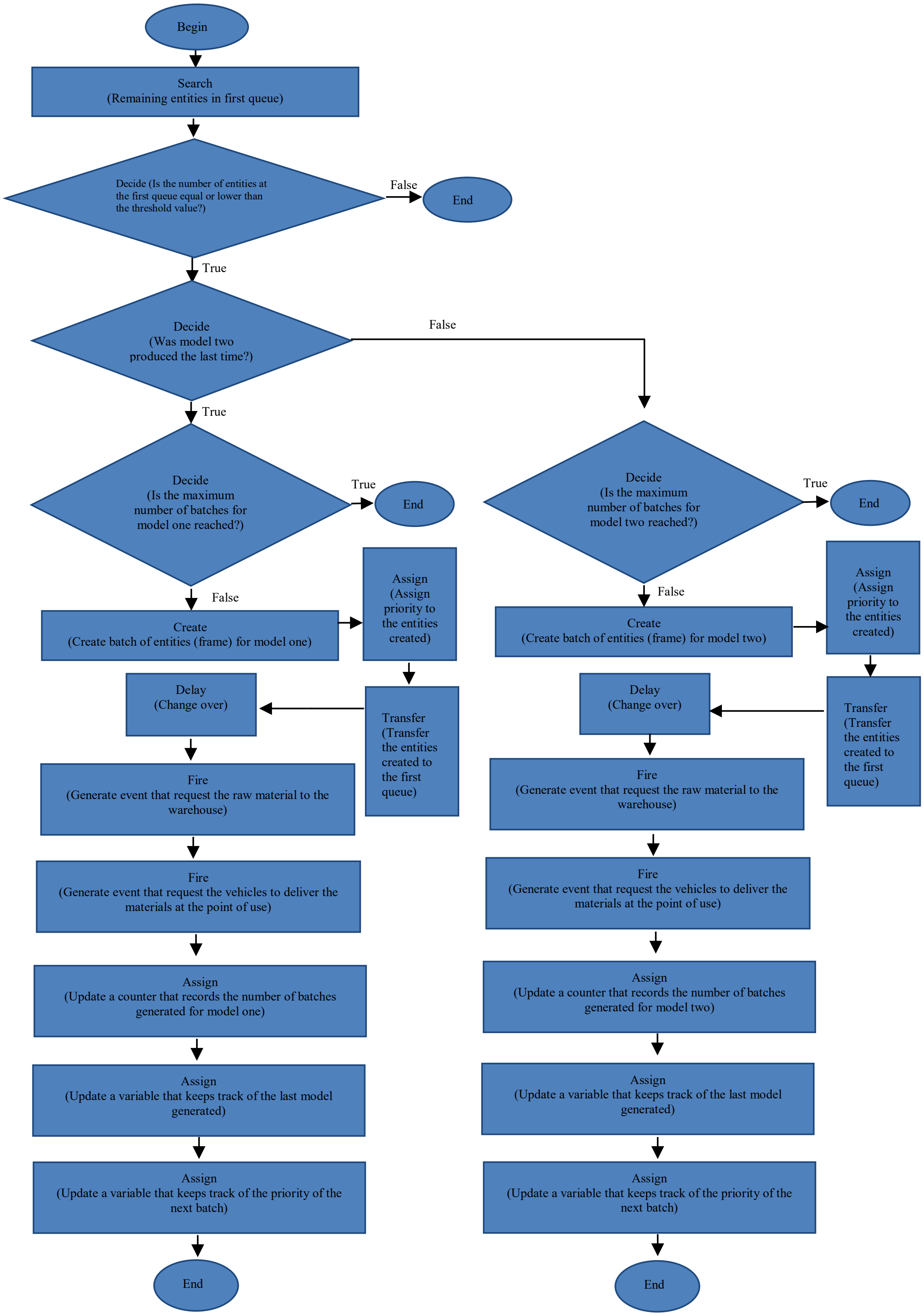

Figure 2.5: Kanban based material handling algorithm.

The simulation presented here contains 204 *Sources*. *Sources* one and two are used to produce the frames of the vehicles that enter the trim shop and the frames of the doors that enter the door assembly shop, respectively. The remaining *Sources* are used to create the raw materials that came from the warehouse and will be used to assemble the vehicles and doors.

After the logic of the automotive manufacturing assembly plant was completely built in Simio, 3D animation was added into the model. Animation is a tool that helps designers to troubleshoot the logic of the model. In addition, 3D animation is becoming an excellent method to communicate the simulation results to the stakeholders. A picture of the 3D animation built for the automotive assembly plant is shown in Fig. 2.6.

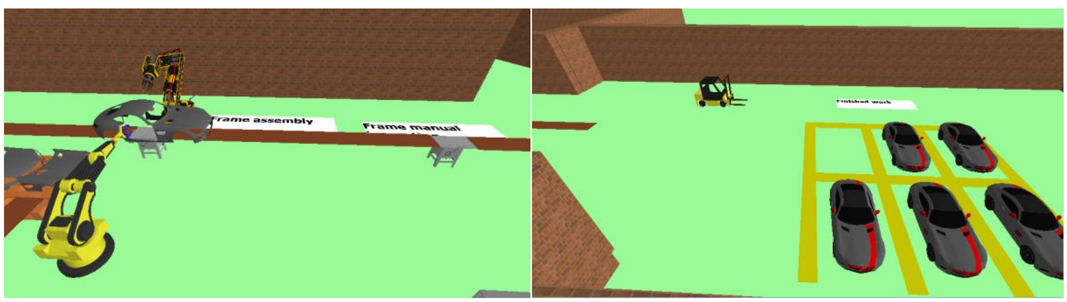

Figure 2.6: 3D view of the generated automotive manufacturing assembly plant.

It is important to mention that the automotive manufacturing facility is simulated and modeled as a pull system, because the production is pulled from the client through the whole value stream. In a similar way, the warehouse waits until it gets the Kanban signal generated by production floor to collect the raw materials and to deliver them at the point of use.

In order to record the WIP, an *Add-On Process Step* called *Excel Write* is used to connect Simio with Microsoft Excel, refer to Fig. 2.7. The *Excel Connect*, *Worksheet*, *Row*, *Starting Column* and *Items* are defined accordingly, refer to Fig. 2.8. In order to calculate the total WIP, an *Integer State Variable* was defined to record the total WIP cumulated in the virtual production

floor. In order to obtain the simulated time the default Simio variable *TimeNow* was recorded. The total Lead Time was recorded in a similar fashion.

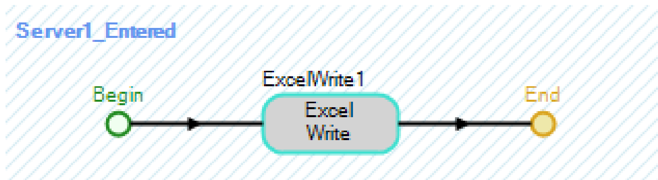

Figure 2.7: *ExcelWrite* Step.

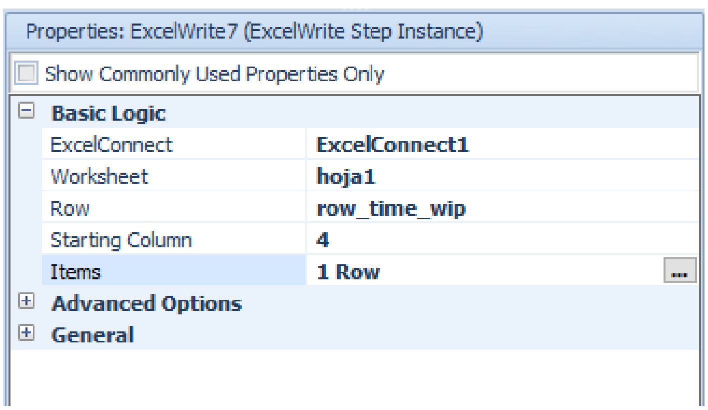

Figure 2.8: Properties of *ExcelWrite* Step.

Chapter 3: Numerical results and conclusions

In this section, the proposed model is used to simulate the entire assembly manufacturing plant with the goal of analyzing and leveling the production of two different vehicle models. It will be shown how the proposed model enables "what-if" analyses for improved planning that will bring down the WIP and lead time. Therefore, the probability of production delays will be reduced.

The processing times for both vehicle models are considered to be the same for common workstations, except for the first workstation located at trim section one and the first workstation located at chassis section one. The processing times for vehicle model two at those two workstations are eight times longer than the processing times for vehicle model one.

In order to determine the proper warm-up period, an arbitrary mix of 30 vehicles (i.e. 30 vehicles for model one followed by a change-over delay, then 30 vehicles for model two) was run for 30 days. It was observed that steady-state is reached after 1.55 hours. Hence, a warm-up period of 2 hours was defined. The warm-up period allows the statistical accumulators of the software (Simio) to be cleared during the specified time [37]. Fig. 3.1 shows a graph of the first 50 hours of the simulation time versus the WIP, which was used to determine the transient and steady-state periods of the system.

After the warm-up period was defined, an experiment was run to determine the proper number of replications. Table 3.1 shows the average WIP, average lead time, and percentage of utilization for stations 1, 18, 34, 50, 61, 71, and 83. Ten scenarios were considered with one to ten replications. Figs. 3.2 and 3.3 show the SMORE plots generated with the ten scenarios for the average lead time and the average WIP, respectively. Since convergence is achieved between

scenario 9 and 10, a number of ten replications was selected for this analysis. Simio SMORE plot are a combination of a box plot, a histogram, and a dot plot, refer to [1].

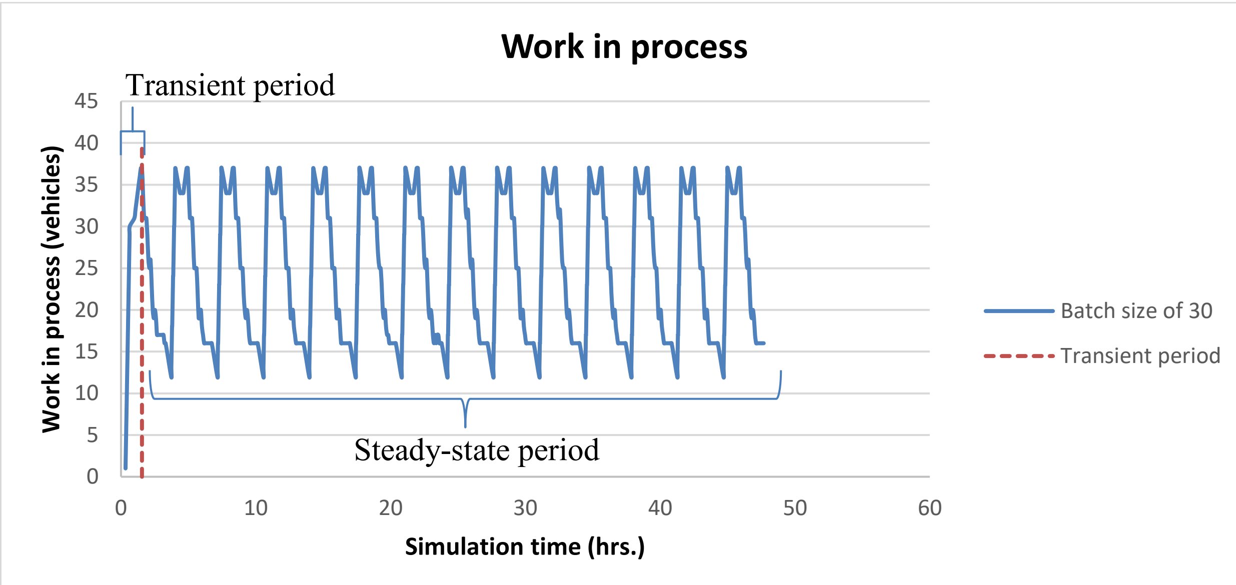

Figure 3.1: Transient and steady-state period analysis for the average WIP.

Table 3.1: Results of the experiment.

Number of replications	WIP average (vehicles)	Lead time average (hours)	Utilization Station 1 (percentage)	Utilization Station 18 (percentage)	Utilization Station 34 (percentage)	Utilization Station 50 (percentage)	Utilization Station 61 (percentage)	Utilization Station 71 (percentage)	Utilization Station 83 (percentage)
1	24.4827	2.61297	1.70928	8.792772	11.7205	8.7934	5.86428	2.93473	5.85879
2	24.4783	2.6131	1.70867	8.78732	11.7214	8.79161	5.86321	2.93	5.85309
3	24.4755	2.61322	1.70907	8.78902	11.7192	8.79085	5.86147	2.931	5.85182
4	24.4791	2.61332	1.70968	8.79184	11.72	8.79158	8.86216	2.93157	5.85594
5	24.4792	2.61324	1.70942	8.79185	11.7198	8.79155	5.86265	2.93185	5.85772
6	24.4774	2.61334	1.70919	8.78929	11.7206	8.79088	5.86265	2.932	5.85818
7	24.4779	2.61328	1.70893	8.78743	11.721	8.79058	5.86306	2.93176	5.8573
8	24.4785	2.6132	1.70863	8.78797	11.7223	8.79018	5.86312	2.93188	5.8579
9	24.4789	2.61319	1.70837	8.78829	11.7228	8.79046	5.86309	2.9316	5.85769
10	24.4787	2.61314	1.70837	8.78877	11.7222	8.79036	5.86295	5.85739	5.85739

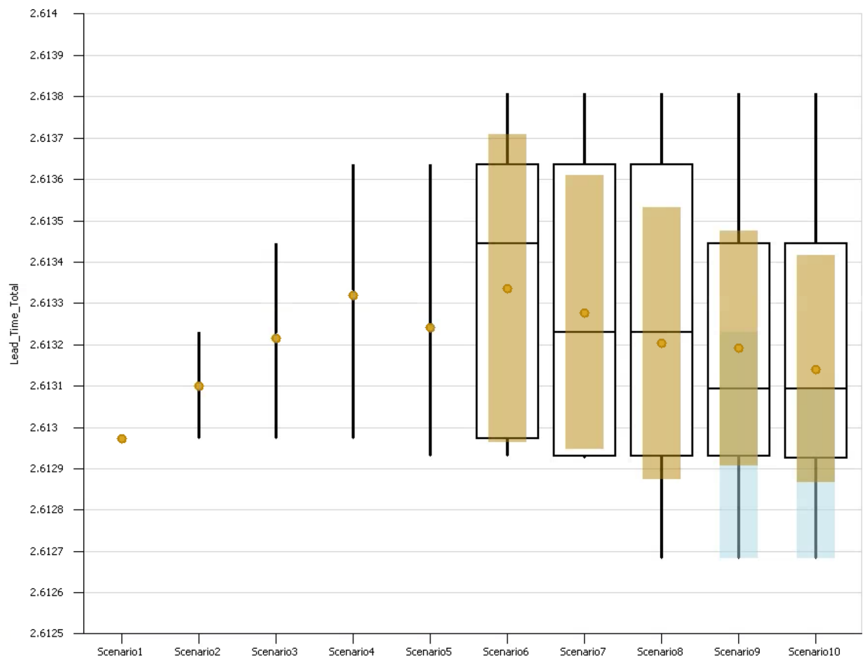

Figure 3.2: Average lead time SMORE plot.

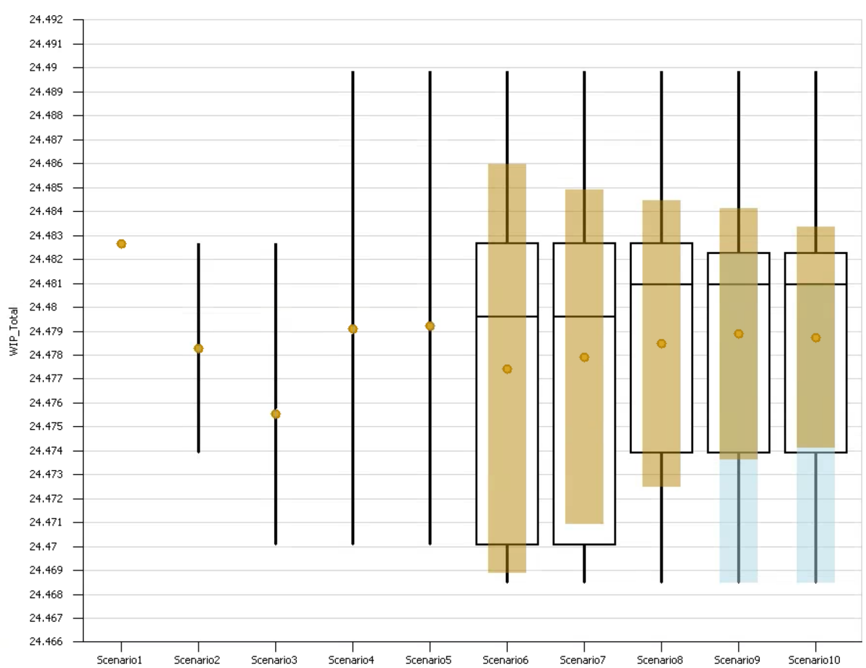

Figure 3.3: Average WIP SMORE plot.

3.1 A Heijunka (leveling production) analysis.

As mentioned before, the word Heijunka has two different but related meanings. One is to level the production by volume, and the other one is to level the production by product type. Hence, the production needs to be leveled by volume first. The assumed historical monthly demand for each model during four months is 8000, 6000, 7500 and 7300 units, respectively. Then, the average of these demands will give one the amount of vehicles to produce the next month (i.e. 7200 units for each model).

In order to see how much the WIP could increase, the capacity of the buffers in the model is defined as infinity for this study. The simulation runs for thirty days (720 hours). First, a batch size of 7200 units was scheduled for both vehicle models considering a change-over between the models of 20 minutes. Graphs of the WIP and the lead time are shown in Figs. 3.4 and 3.5, respectively.

One can see in Fig. 3.4 a linear grow of the WIP from time zero until 76 hours with a maximum value of 4761 units being processed. After that, the WIP linearly drops until it reaches a WIP of fourteen units. In the other hand, one can observe in Fig. 3.5 a linear grow of the lead time which never stops, reaching a maximum value of 582 hours. Two different slops of the lead time are shown, which belong to the two different vehicle models. One can see from both graphs that the system is unstable for this batch size.

Due to the fact that just leveling production by volume is not enough to achieve the expected demand, the production needs to be leveled by product type or mix. The goal is to stabilize the unstable system obtained with the previous run. The system was run with batch sizes 20, 30, 40, 50, 80, 90, 100, 150 and 200 units. For example, a batch size of 20 units means that 20 units of vehicle model one are manufactured followed by a change-over time and a batch of

20 units of vehicle model two, until the maximum time or the maximum number of units is reached (i.e. 720 hours or 14400 units). The obtained results are shown in Figs. 3.6 to 3.23. Fig. 3.24 and 3.25 show a summary with these results.

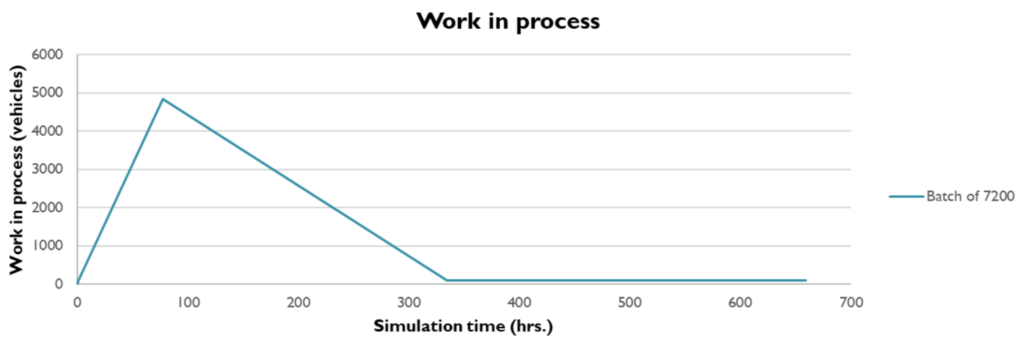

Figure 3.4: WIP for a batch size of 7200 vehicles.

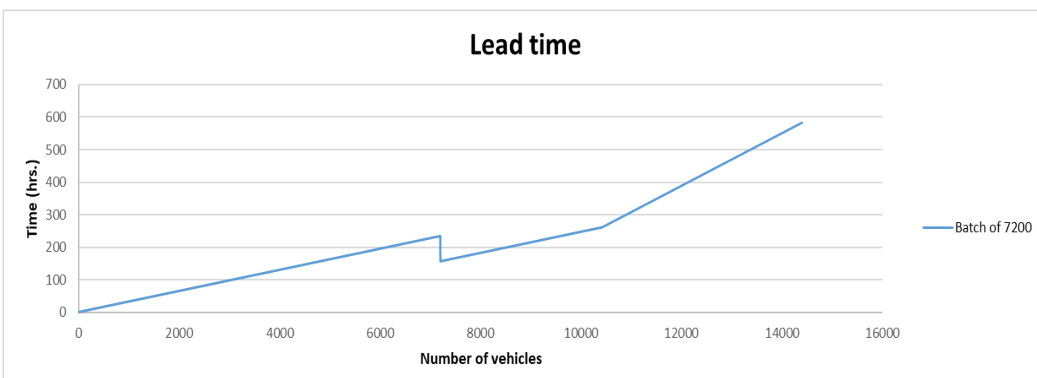

Figure 3.5: Lead time for a batch size of 7200 vehicles.

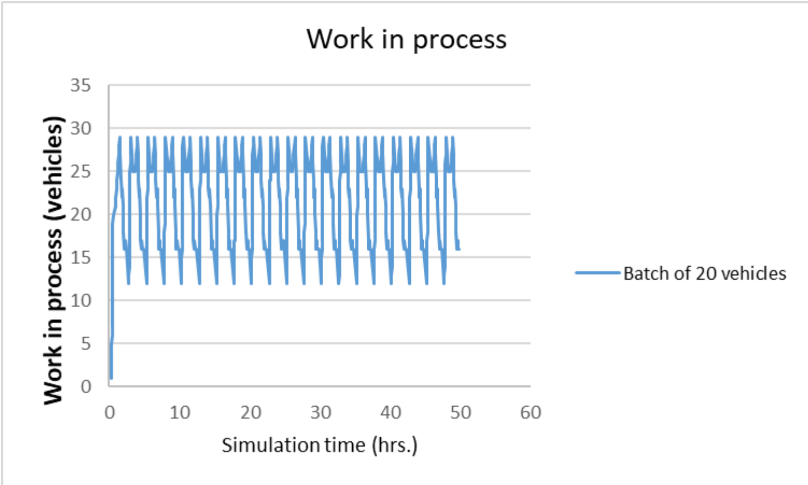

Figure 3.6: WIP for batch sizes of 20 vehicles.

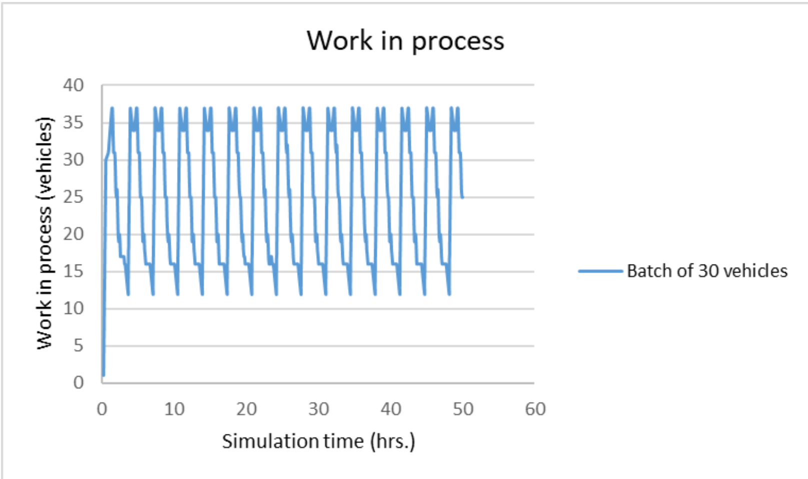

Figure 3.7: WIP for batch sizes of 30 vehicles.

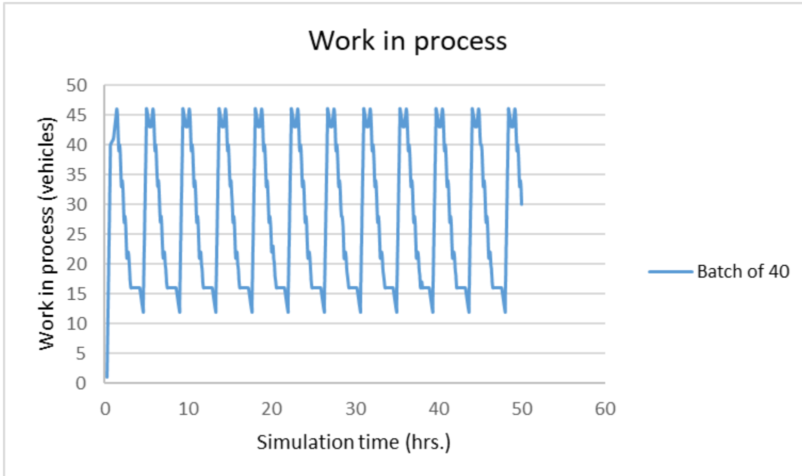

Figure 3.8: WIP for batch sizes of 40 vehicles.

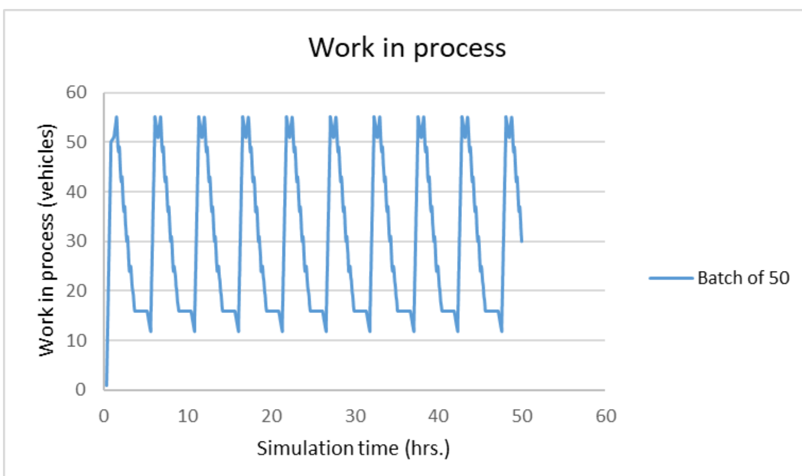

Figure 3.9: WIP for batch sizes of 50 vehicles.

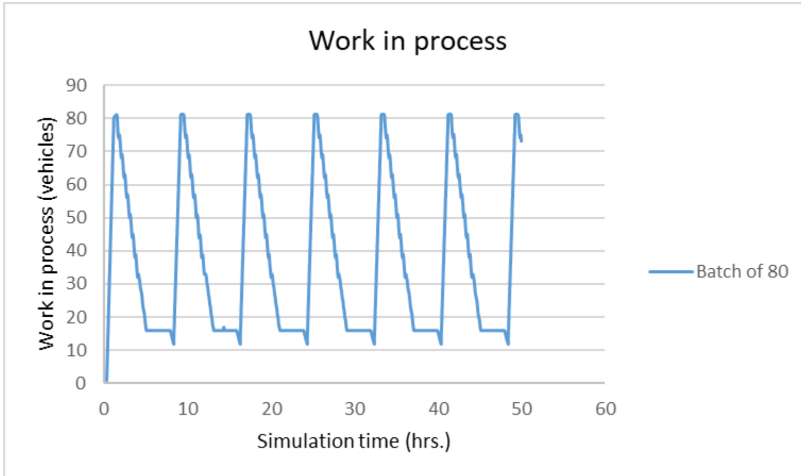

Figure 3.10: WIP for batch sizes of 80 vehicles.

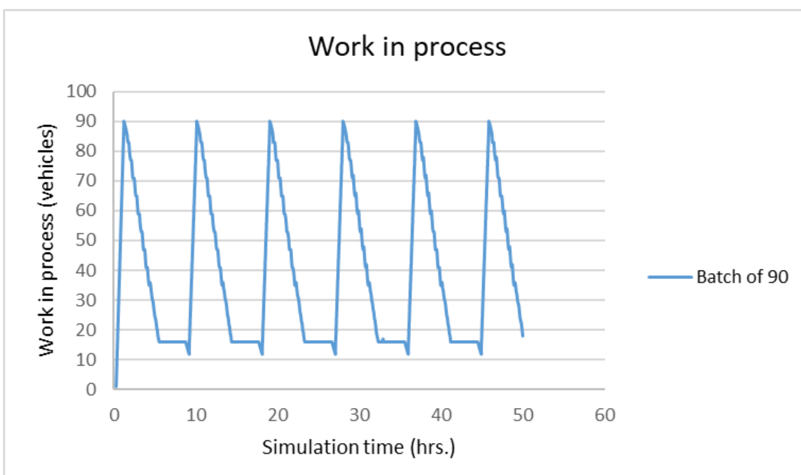

Figure 3.11: WIP for batch sizes of 90 vehicles.

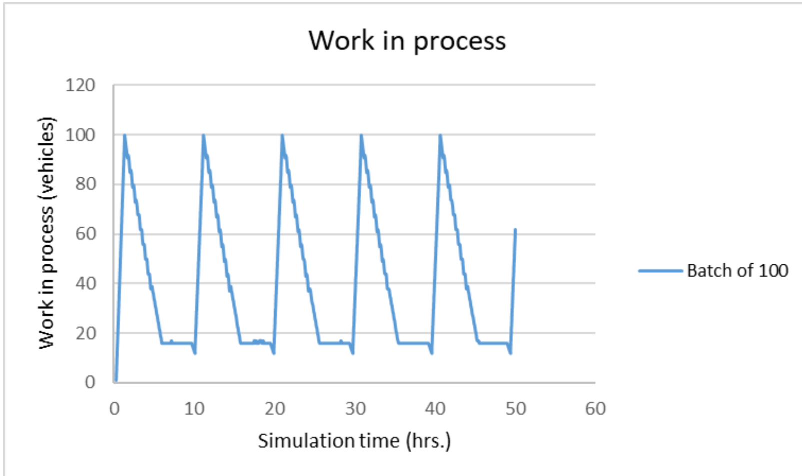

Figure 3.12: WIP for batch sizes of 100 vehicles.

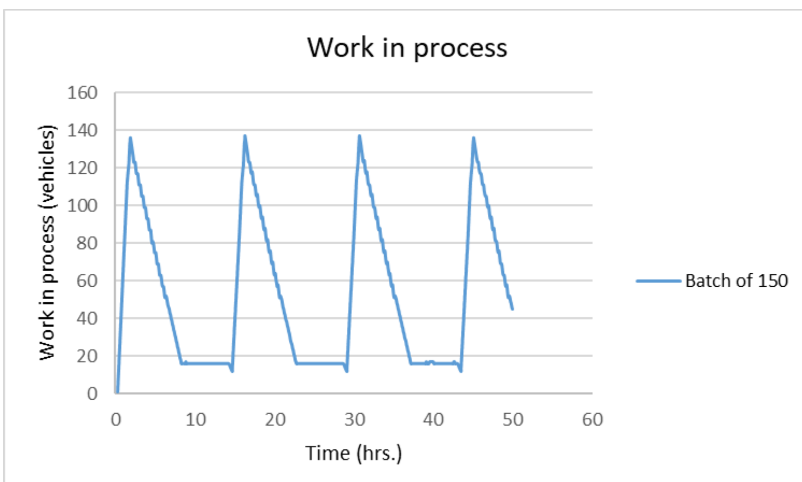

Figure 3.13: WIP for batch sizes of 150 vehicles.

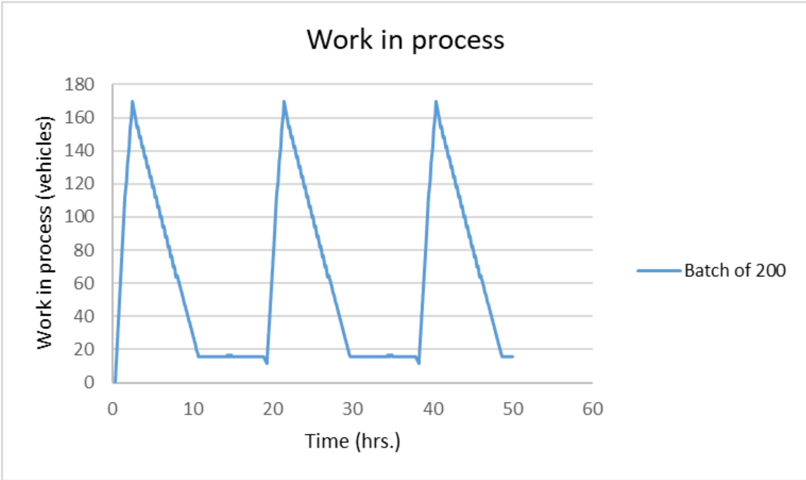

Figure 3.14: WIP for batch sizes of 200 vehicles.

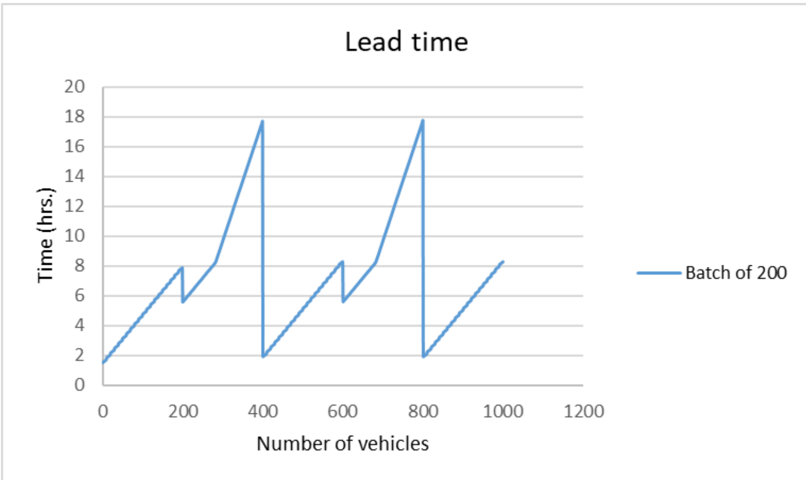

Figure 3.15: Lead times for batch size of 200 vehicles.

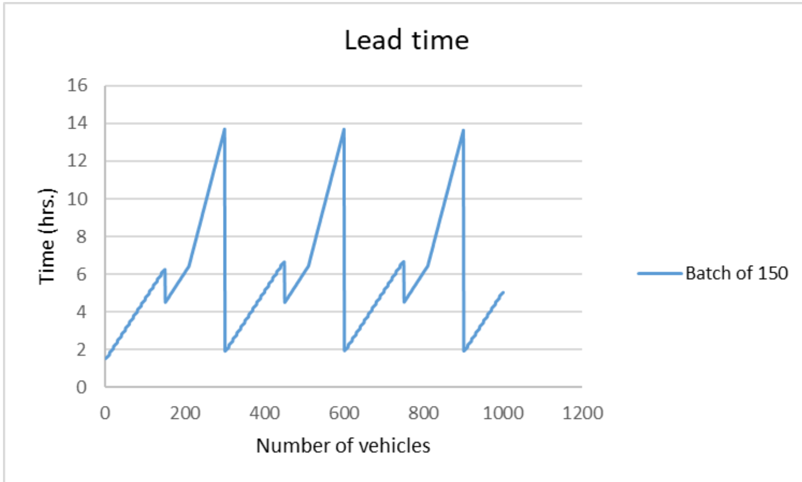

Figure 3.16: Lead times for batch size of 150 vehicles.

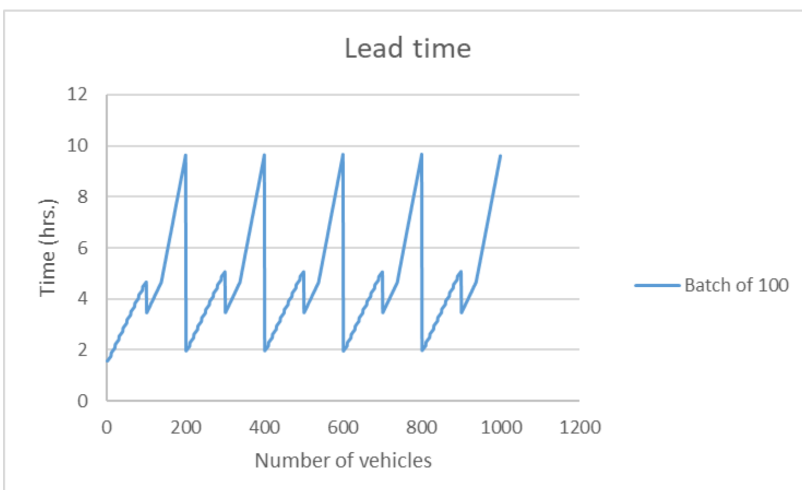

Figure 3.17: Lead times for batch size of 100 vehicles.

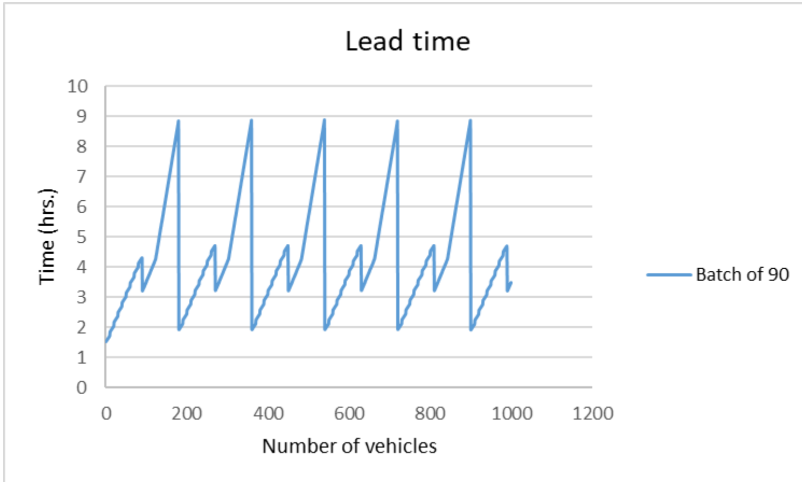

Figure 3.18: Lead times for batch size of 90 vehicles.

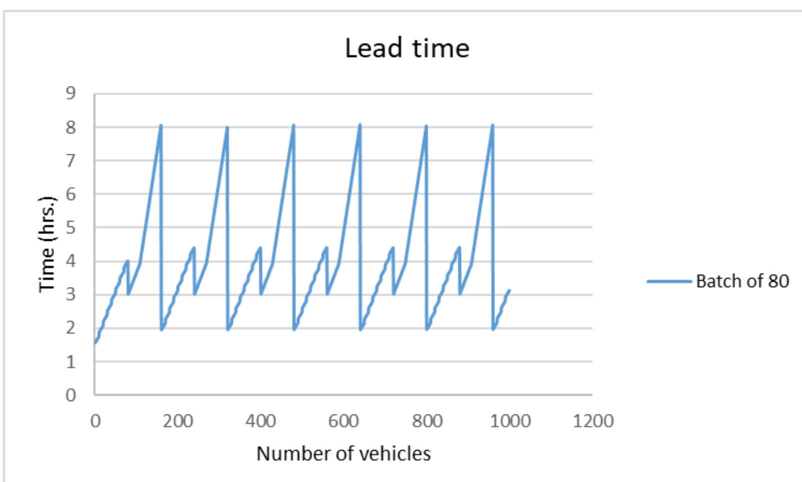

Figure 3.19: Lead times for batch size of 80 vehicles.

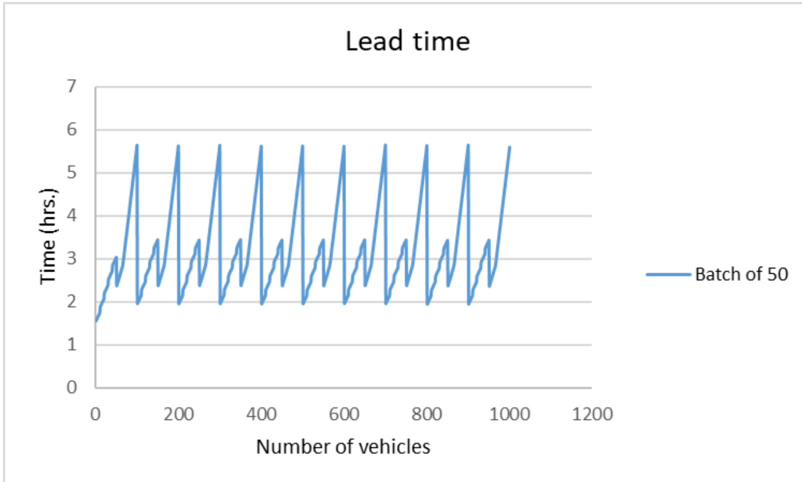

Figure 3.20: Lead times for batch size of 50 vehicles.

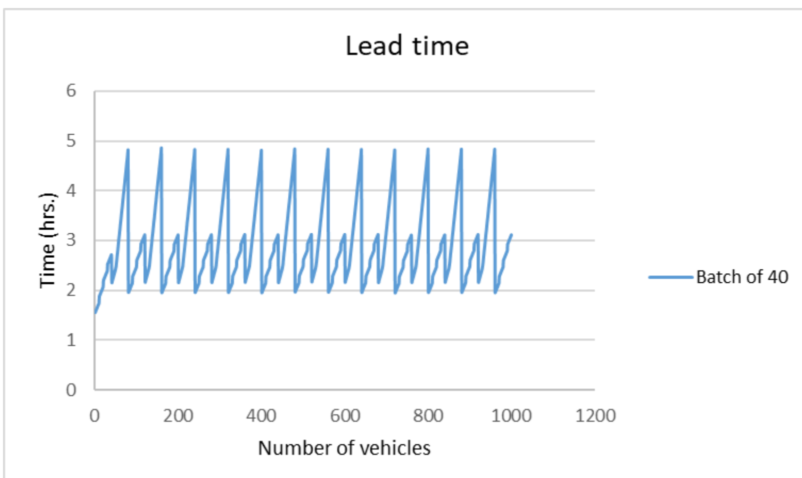

Figure 3.21: Lead times for batch size of 40 vehicles.

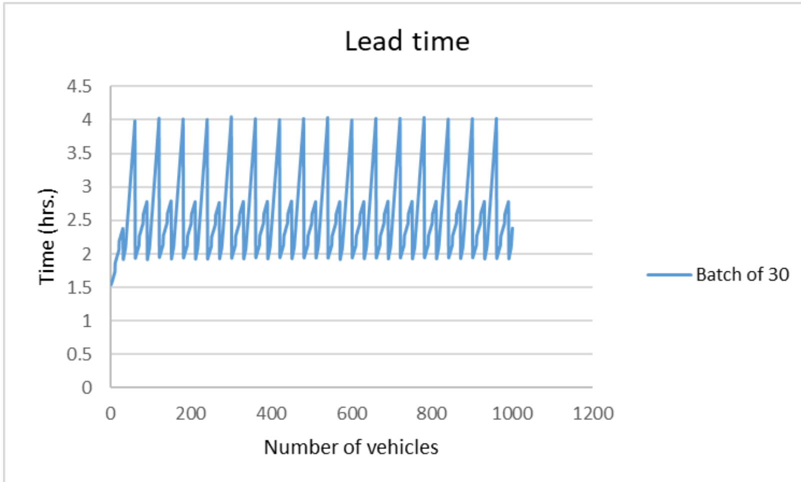

Figure 3.22: Lead times for batch size of 30 vehicles.

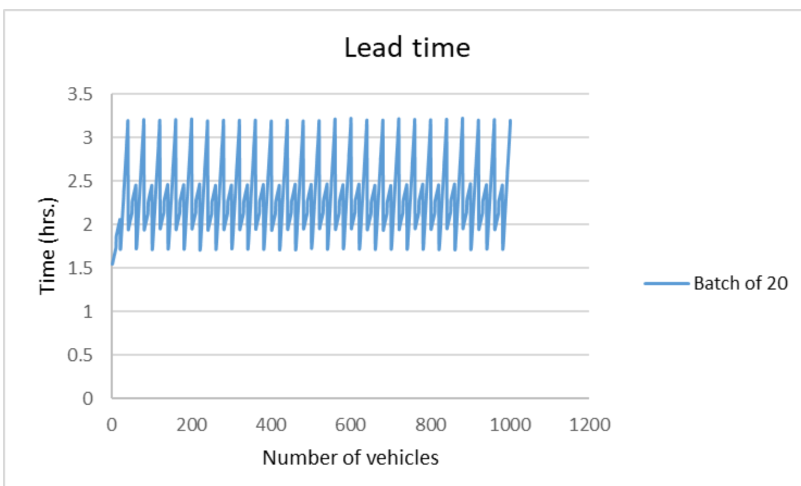

Figure 3.23: Lead times for batch size of 20 vehicles.

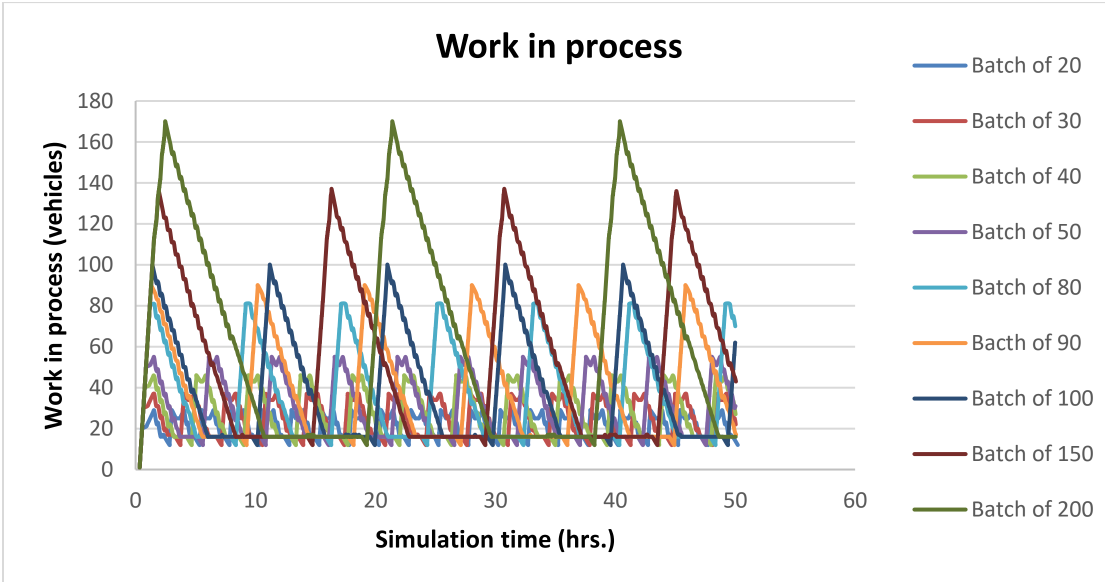

Figure 3.24: WIP for batch sizes of 20, 30, 40, 50, 80, 90, 100, 150 and 200 vehicles.

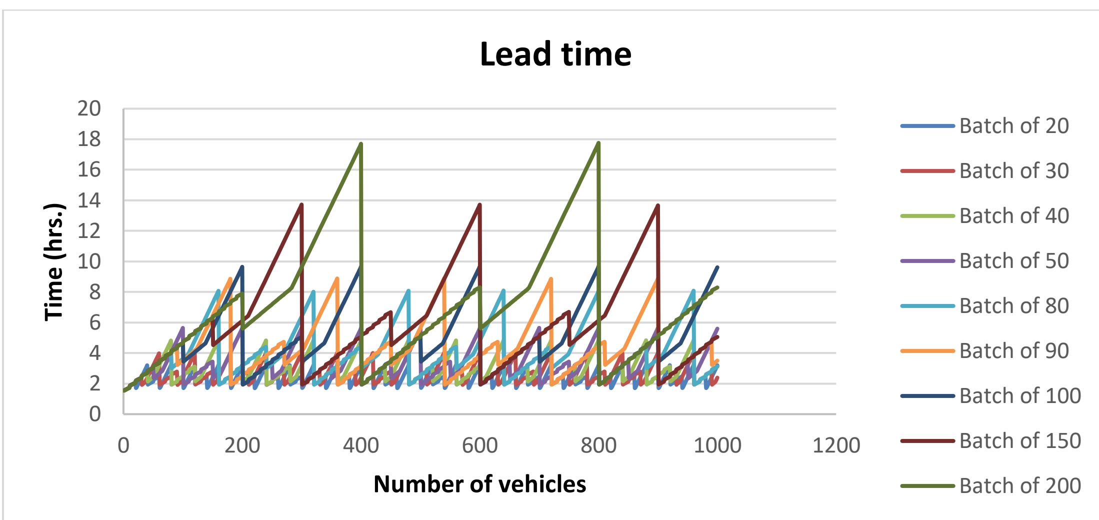

Figure 3.25: Lead times for batch sizes of 20, 30, 40, 50, 80, 90, 100, 150 and 200 vehicles.

One can see in Figs. 3.6 and 3.25 that the uncontrolled linear grow disappear for both, the WIP and the lead time. It was observed that the lower the batch size, the lower the WIP in the system. Similarly, the lower the batch size, the lower the lead time.

In table 3.2 a summary with the results obtained with the different batch sizes is presented. One can see that the system is no able to produce the expected demand with batch sizes of 20, 30, 40, 50, 60, and 80 units. The system is able to produce the expected demand with batch sizes of 90, 100, 150, 200 and 7200 units. Nevertheless, the higher the batch size, the higher the WIP. Thus, for this particular case a batch size is 90 units is proposed. With this batch size, the average lead time and maximum WIP for vehicle model one and two are 3.33 hours, 90 units, 5.54 hours and 35 units, respectively.

3.2 Conclusions.

This thesis successfully shows the capabilities of new commercial simulation software (Simio simulation software) and the potential benefits for the automotive industry. It is described in detail a methodology to model production floor, warehouse and the material handling system of an automotive manufacturing assembly plant. This method allows a convenient interaction of these three sections of the factory into a unique model, resulting in a model that combines all the critical variables of the system.

Two lean manufacturing tools where implemented into this model: Kanban and Heijunka. In order to model the Kanban system, an algorithm was developed and programmed using the *Processes* tool offered by Simio. In order to model Heijunka, different batch sizes were tested in the developed model. It was observed that the WIP and the lead time grow up as a function of the

batch size. It suggests that the analyzed system becomes unstable if the batch size is not properly chosen.

The results suggest that a batch of 90 vehicles is the option that allows the current system to produce the required demand on time with the minimum amount of WIP and lead time. The proposed methodology allows one to perform valuable "what-if" analyses. This strategy can help planners and schedulers to take more accurate and faster production decisions.

Table 3.2: Lead time, units manufactured and WIP for both vehicle models.

	Batch size	**Lead time (average hrs.)**	**Lead time (maximum hrs.)**	**Units (vehicles) manufactured**	**WIP (max)**
Model 1	20	2.20	2.48	5760	20
	30	2.36	2.80	6330	30
	40	2.52	3.12	6644	40
	50	2.68	3.44	6859	50
	60	2.84	3.76	7020	60
	80	3.17	4.42	7200	80
	90	3.33	4.75	7200	90
	100	3.49	5.06	7200	100
	150	4.30	6.68	7200	136
	200	5.10	8.30	7200	170
	7200	118.17	234.80	7200	4848
Model 2	20	2.43	3.21	5756	16
	30	2.86	4.03	6317	19
	40	3.30	4.84	6640	22
	50	3.74	5.64	6850	24
	60	4.19	6.45	6999	27
	80	5.09	8.09	7198	32
	90	5.54	8.89	7200	35
	100	5.99	9.72	7200	38
	150	8.25	13.75	7200	51
	200	10.51	17.78	7200	65
	7200	327.05	582.85	7200	1942

References

[1] W. D. Kelton, J. S. Smith, D. T. Sturrock, “Simio and simulation: Modeling, analysis, applications”, Pittsburg Pennsylvania, Simio LLC, 2014.

[2] G. Michalos et al, “Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach”, *CIRP journal of manufacturing science and technology*, vol. 2, no. 2, pp. 81-91, January 2010.

[3] Q. Hao, W. Shen, “Implementing a hybrid simulation model for a Kanban-based material handling system”, *Robotics and computer-integrated manufacturing*, vol. 24, no. 5, pp. 635-646, October 2008.

[4] J. Wang et al, “Data driven production modeling and simulation of complex automobile general assembly plant”, *Computers in industry*, vol. 62, no. 7, pp. 765-775, September 2011.

[5] D. Kibira, C. R. McLean, “Generic simulation of automotive assembly for interoperability testing”, *Winter 2007 simulation conf.*, Washington, DC, 2007, pp. 1035-1043.

[6] V. Patel et al, “Discrete event simulation in automotive final process system”, *winter 2002 simulation conf.*, San Diego, CA, pp. 1030-1034.

[7] Q. Yu et al, “Productivity simulation with promodel for an automotive assembly workstation involving a lift assist device”, *winter 2006 simulation conf.*, Monterey, CA, pp. 1935-1939.

[8] P. F. Andrade et al, “Value stream mapping and lean simulation: a case study in the automotive industry”, *Int. journal of advanced manufacturing technology*, vol. 85, no. 1-4, pp. 547-555.

[9] D. T. Jones, (2006). Heijunka: Leveling production. [Online]. Available: http://advancedmanufacturing.org/heijunka-leveling-production.

[10] J. K. Liker, “The Toyota way”, McGraw-Hill, New York, NY, 2004.

[11] C. Lippolt and K. Furmans, “Sizing of Heijunka-controlled Production Systems with Unreliable Production Processes,” *IFIP-The international Federation of Information Processing*, Boston, USA, 2008.

[12] F. Bohnen et. al, “Leveling of low volume and high mix production based on a Group Technology approach,” *CIRP Journal of Manufacturing Science and Technology*, vol. 4, no. 3, pp. 247-251, July 2011.

[13] A. Huttmeir et al, “Trading off between heijunka and just-in-sequence”, *International journal of production economics*” vol. 118, no. 2, pp. 501-507, January 2009.

[14] P. Korytkowski et al, “Multivariate simulation analysis of production leveling (heijunka) - a case study”, *7th IFAC conference on manufacturing modeling, management, and control*, Saint Petersburg, Russia, June 2013.

[15] F. Grimaud et al, “Exponential smoothing for multi-product lot-sizing with heijunka and varying demand”, *Management and production engineering review*, vol. 5, no. 2, pp. 20-26, June 2014.

[16] E. Erel and S. Sarin, “A survey of the assembly line balancing procedures,” *Production Planning and Control*, vol. 9, no. 5, pp. 414-434, Nov. 2010.

[17] O. Battaia and A. Dolgui, “A taxonomy of line balancing problems and their solution approaches,” *Int. J. Production Economics*, vol. 142, pp. 259-277, April 2013.

[18] C. Becker and A. Scholl, “A survey on problems and methods in generalized assembly line balancing,” *European Journal of Operational Research*, vol. 168, no. 3, pp. 694-715, Feb. 2006.

[19] J. Sternatz, “The joint line balancing and material supply problem,” *Int. J. Production Economics*, vol. 159, pp. 304-318, Jan. 2015.

[20] L. Wilson, “How to implement lean manufacturing”, McGraw Hill, 2010.

[21] J. Matzka et al, "Buffer sizing of a Heijunka Kanban system," *Journal of Intelligent Manufacturing*, vol. 23, no. 1, pp. 49-60, September 2009.

[22] R. Marek et. al, "Understanding the fundamentals of Kanban and CONWIP pull systems using simulation," *Proceedings of the 2001 Winter Simulation Conference,* Arlington, USA, August 2002.

[23] P. Köchel and U. Nieländer, "Kanban optimization by simulation and evolution," *Production Planning and Control*, vol. 13, no. 8, pp. 725-734, November 2010.

[24] F. Lolli et al, "A simulative approach for evaluating alternative feeding scenarios in a Kanban system", *International Journal of Production Research*, vol. 54, no. 14, pp. 4228-4239, Nov. 2015.

[25] V. Limere et al, "Optimising part feeding in the automotive assembly industry: deciding between kitting and line stocking," *International Journal of Production Research*, vol. 50, pp. 4046-4060, Jan. 2011.

[26] C. Dennis Pegden, "Deliver on your promise. How simulation-based scheduling will change your business", ed. Simio LLC, Pittsburg Pennsylvania, 2017.

[27] M. H. Kim, Y. Kim, "Simulation-based real-time scheduling in a flexible manufacturing system," *Journal of Manufacturing Systems*, vol. 13, no. 2, pp. 85-93, 1994.

[28] M. Frantzen et. al, "A simulation-based scheduling system for real-time optimization and decision making support," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 4, pp. 696-705, August 2011.

[29] L. Shen, S. Dauzere-Peres, J. S. Neufled, "Solving the flexible job shop scheduling problem with sequence-dependent setup times," *European Journal of Operational Research*, vol. 265, no. 2, pp. 503-516, March 2018.

[30] A. Agnetis et al, "Scheduling of flexible flow lines in an automobile assembly plant," *European Journal of operational research*, vol. 97, no. 2, pp. 348-362, March 1997.

[31] S. Emde et al, "Scheduling in-house transport vehicles of feed parts to automotive assembly lines," *European Journal of operational research*, vol. 260, no. 1, pp. 255-267, July 2017.

[32] N. Boysen et al, "Sequencing mixed-model assembly lines: Survey, classification and model critique" *European Journal of Operational Research*, vol. 192, no. 2, pp. 349-373, Jan. 2009.

[33] N. Boysen et al, "Production planning of mized-model assembly lines: overview and extensions," *Production Planning and Control*, vol. 20, no. 5, pp. 455-471, Oct. 2010.

[34] D. Battini et al, "Preventing ergonomic risk with integrated planning on the assembly line balancing parts feeding," *International Journal of Production Research*, vol. 55, no. 24, pp. 7452-7472, Aug. 2017.

[35] M. Dehghanimohammadabadi and T. Keyser, "Intelligent simulation: Integration of SIMIO and MATLAB to deploy decision support systems to simulation environment," Simulation Modelling Practice and Theory, vol. 71, pp. 45-60, Feb. 2017.

[36] T. Tseng et al, "Optimization of wind turbine placement layout on non-flat terrains," *International Journal of Industrial Engineering*, vol. 21, no. 6, pp. 384-395, June 2014.

[37] W. D. Kelton, R. P. Sadowski, N. B. Zupick, "Simulation with Arena", New York, Ney York, Mc Graw Hill, 2015.

Vita

Ivan Arturo Renteria Marquez was born in Ciudad Juárez, Chihuahua. He graduated from Colegio de Bachilleres plantel número 7, Ciudad Juárez, Chihuahua, in the spring of 2005 and he entered the Mechatronic engineering program at Tecnológico de Monterrey with an academic scholarship. In the spring of 2011 he graduated with distinction. In the spring of 2012 he entered the PhD fast track program of the University of Texas at El Paso. In the spring of 2016 he obtained his Philosophy Degree with specialty in Electrical engineering.

Permanent address: Wallington

El Paso, Texas, 79902

This thesis was typed by Ivan Arturo Renteria Marquez.