CHAPTER III

RESEARCH METHODOLOGY

3.1. Problem Identification

Problem identification is the early step of this study. The problem identification is obtained from the literature review that has been conducted. According to the literature review, the mathematical model is used to problem solving for mixed-model two-sided assembly line balancing, due to the high complexity makes it impossible to be solved to optimality. Furthermore, optimization procedures are presented to solve mixed-model two-sided assembly line balancing (MTALB) problem. Although it finds the better solution using metaheuristic approaches, but these solutions involve complex calculations and thousand constraints which uncertain result. Simulation could be an innovative approach that is used to gets around these limitations. However, it must be proved. Besides, the research from Qi et al. (2015) did the MTALB type II problem with converting data. They convert the data from three process time in automobile product A, B, C to be one process time. It can't be guaranteed that is concrete data. The implementation of MTALB Type II problem without converting the process time data of each product to be one process time is required. It is to ensure the data is suitable to used. This problem identification is organized as question form which will be answered and explained in the discussion and conclusion.

This research was conducted in PT. Toyota Motor Manufacturing Indonesia (PT. TMMIN), one of automobile industry in karawang international industrial city (KIIC), located in Karawang, West Java. The company focuses on assembling car products. The products are made from the request of the costumer, start from production planning until product delivery to customers where the products have detail specification.

3.2. Problem Formulation

The research is focused on improvement the mixed-model two-sided assembly line balancing type II problem in automobile industry. This research will be conducted using simulation approach as solution procedure for MTALB. In addition, the problem test from literature review will be added to valuate the system performance of the proposed assembly line.

3.3. Literature Review

The literature review is used to find out the gap between the previous study and the recent research would be conducted. Literature review is also designed for gaining the state of the art of the study which will support the problem identification. Literature review is divided into two studies, which are Deductive and Inductive study. Deductive study is conducted to obtain the basic concept of theory related with this research, while the Inductive study is conducted to map the related research and see the position of this research compared among previous research.

3.4. Data Collection

Data collection for this research is using secondary data. The secondary data is data that obtained from the literature or indirect observation of conducted by researcher. Data that collected is the database of the company namely PT. Toyota Motor Manufacturing Indonesia. The data obtained will be used for the main calculation and information for this research, The data can be structured as follows:

a. Technical data
b. Organizational data
c. System load data

The following overview is a small selection of data to be collected can be seen in Table 3.1:

Table 3.1 Data collection

Technical data	
Factory structural data	Layout
	Transport functions
	Transport routes
	Areas
	Restrictions
Manufacturing data	Task times
	Cycle time
	Takt Time
	Capacity
Material flow data	Precedence and successor of tasks
	Conveyors
	Capacities
Organizational data	
Working time organization	Break scheme
	Shift scheme
Resource allocation	Worker
	Conveyors
System load data	
Product data	Working plans
Job data	Production orders
	Transportation orders
	Volumes

3.5. Data Processing

Data processing will briefly explain how to build a simulation model of MTALB after data collecting process was performed. The software that will be used to perform data processing is Tecnomatix Plant Simulation by Siemens. Here are the steps to build a simulation model, as follows:

3.5.1 Model

Before building the simulation model, the model of this research should be identified. It is made of composition concepts which are used to help people know, understand, or simulate a subject the model represents. The model of mixed-model two-sided assembly line in this research can be seen in Figure 3.1.

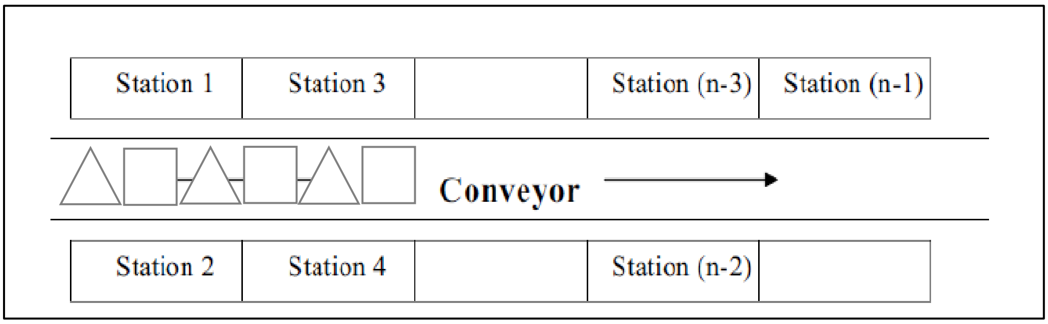
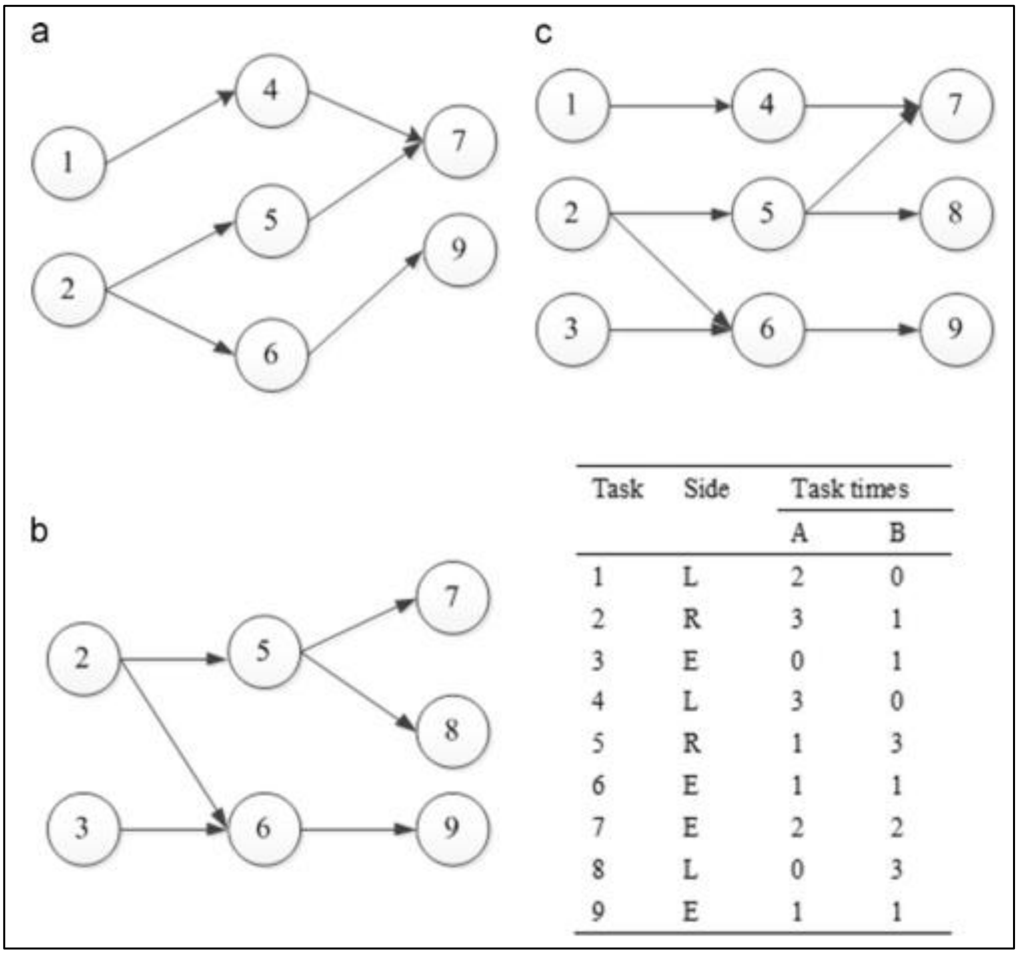


Figure 3.1 Mixed-model two-sided assembly line model

Source: Mercan (2012)

In mixed model, each model has its own predecessor relationship which can be described by precedence graph. All of the precedence graphs of the model can be combined into a single predecessor diagram, called a joint precedence graph. Figure 3.2 shows the example of joint precedence graph, the task time and direction of two models. Each cycle on each model represents a task and an arrow connecting their predecessor. Each task in the joint precedence graph has different task times but identical operating directions for different models.



Task	Side	Task times	
		A	B
1	L	2	0
2	R	3	1
3	E	0	1
4	L	3	0
5	R	1	3
6	E	1	1
7	E	2	2
8	L	0	3
9	E	1	1

Figure 3.2 Precedence graph, task time and the direction from two models

Source: Yuan et al. (2015)

Furthermore, some data needed for being processed or calculated. The data will be input of the simulation. The notations used in mathematical formulations are given as follows:

Indices:

i = task

j = mated-station

m = product model

k = side of the line

$$= k \begin{cases} 1 \; indicates\ a\ left\ side\ station \\ 2 \; indicates\ a\ right\ side\ station \end{cases}$$

NS = number-station

Parameters:

I = set of tasks in the combined precedence diagram; I = {1, 2, ..., i, ..., nt}

J = set of mated-stations; J = {1, 2, ..., j, ..., nms}

M = set of product models; M= {1, 2, ..., m, .. ., npm}

The data are overall proportion of the number of units of model m being produced (q_m) is computed by Equation (1). The forecasted demand, over the given planning horizon (P), for model m ($m \in M$) is Dm. Then, the cycle time (CT) of the MTALB problem is calculated by Equation (2) as maximum constraint and the researcher added the CT minimum in that equation.

Where:

$$q_m = \frac{D_m}{\sum_{m \in M} D_m} \qquad \forall m \in M \tag{1}$$

$$CT_{min} = \frac{q_m \sum_{m \in M} T_{im}}{\sum_{m \in M} D_m} \leq CT \leq CT_{max} = \frac{P}{\sum_{m \in M} D_m} \tag{2}$$

The objective is MTALB type II problem. However, it cannot distinguish between different solutions sometimes. So the weighted line efficiency (WLE) and weighted smoothness index (WSI) are considered as problem test result. The formulations from Özcan & Toklu (2009) which are calculated as follows:

$$WLE = \left(\frac{\sum_{m \in M} qm \left(\sum_{i \in I} t_{im} \right)}{CT.NS} \right) \cdot 100 \tag{3}$$

Equality of workloads between stations is also considered as objective due to the nature of the mixed-model two-sided assembly line problem. Equalization of workloads such as minimizing weighted smoothness index between stations aims to distribute waiting time between stations evenly. Equation (4) is used to calculate the WSI between stations:

$$WSI = \sqrt{\left(\frac{\sum_{m\in M} qm \cdot \left(\sum_{j\in J} \sum_{k=1,2} {}_{m}WL_{j}^{k} - WL_{max}\right)^{2}}{NS}\right)} \quad (4)$$

Where WL_{max} is maximum station time. By the maximization WLE, it aims to reduce the length of line and also by minimizing WSI is to reduce minimizing the difference of workload between station and to distribute workload between station as balanced as possible.

3.5.2 Simulation Procedure

The procedure of simulation of this study consists of five phases. It starts with the preliminary study phase then the simulation study phase, experimentation, analysis of results, and finally discussion, conclusion and recommendation. The beginning phase it starts with a problem statement and setting of objectives which indicate the questions that are to be answered by the simulation study. After that, the actual production system is converted into a model in order to avoid waste in terms of time or monetary, it is beneficial to create a simple plane at first stage and improve it continuously. In this phase, the basic model such as workstation, buffer, and material handling is constructed then coded into a computer recognizable form utilizing Tecnomatix Plant Simulation software. At the same time, the important data such as arrival time, set up time and processing time are collected. Next is to verify the model which is extremely important in determining whether the computer implementation of the model is correct. Verification is to ensure that each element in the model has been coded and behave in the right way. Then the output from the simulation is validated by comparing to the actual system. This is to make sure that the model is a representative copy of the real system under investigation. Once validated, the tool of assembly line balancing (optimization tool) is developed to describe the objective function. In this case, the objective function is minimize cycle time for given a number of workstation which called MTALBP Type II. Once the results are analyzed and satisfactory, the findings are presented and documented, and can be implemented in the real system.

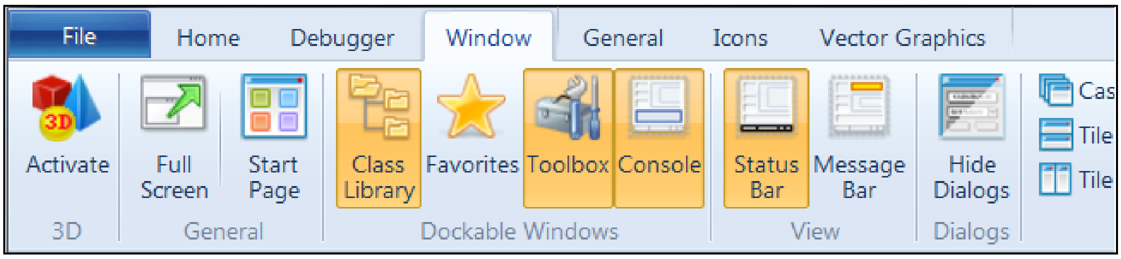
3.5.3 Simulation Model of MTALB Using Tecnomatix Plant Simulation Software

There are many options in market when a simulation model is prompted. Therefore, it is important to choose the most appropriate one. Tecnomatix Plant Simulation is the simulation software that can play an important role in flexibility and capability for modeling. Assembly line especially MTALB is discussed with Tecnomatix Plant Simulation Software. Firstly, the method of assembly line modeling is proposed. A simulation model will be built and tested. The result of modeling has to be included in the model documentation to make further changes of the simulation model possible. In practice, this step is often neglected, so that models due to the lack of documentation of functionality cannot be used. Therefore, there is a needing for commenting the models and the source code during programming. In this way the explanation of the functionality is still available after programming is finished. Assembly line modeling is still not giving the optimal solution to solve the problem. Furthermore, assembly line simulation optimization is researched by Hu et al. (2014). Secondly, assembly line simulation optimization is researched. The simulation optimization technique is general embed in the process of simulation. It is the process finding the best input variable value from feasible domain. The result is output be the optimal solution or satisfied solution. The goal is to consume the least resource while get the most information in simulation experiment and let the user decide easily. Likewise, Bangsow (2010) makes the case that Tecnomatix Plant Simulation The software has many assembly line optimization tools, one of the best is Tecnomatix programming method. It is object-oriented simulation control language. Also, It provides assignments and settings of structures that can be described with programming languages or SimTalk. In this case, programming languages can expand how to model, control user simulations and can even program the given equations. Finally, assembly line modeling and simulation optimization are both described by an instance. Therefore, the research about optimization and improvement of the assembly line can satisfy the enterprise’s requirement of improving and optimizing assembly line based on the existing assembly line, and it can greatly improve the efficiency of the enterprise. In order to get the best way, Huang et al. (2012) described the process of assembly line modeling using this software as follows:

3.5.3.1 Assembly Line Modeling Basis

Assembly line modeling needs to plan organization structure. It is very important for modeling especially when assembly line is more complex as well as the situation of requiring a different person to complete different parts. If the structure of the organization is well organized, so that is easy to find the object in the modeling process. It is not only easy to manage but also can improve the efficiency of the modeling effectively with the good organizational structure. The organizational structure of assembly line usually has an assembly workstation, laying-off workstation, mobile equipment, transportation equipment and lifting equipment, etc. Object library must correspond to simulation software library after planning organization.

Tecnomatix Plant Simulation provides objects library of the basic modeling, Including logistics object, information flow object, the user interface object and moving object. Logistics object have the ability to change the moving objects parameters of the object, including workstation and laying-off workstation. Information flow object can control and record the data of simulation. The most of the information flow objects are presented in the form of a table, including variables, table, storage file, etc. The user interface object is the method of the communication between simulation model and the user. It is not only can provide system simulation of relevant information for users, but also can be a control simulation tool for users. The moving object refers to the object that physical location is not limited to one place, such as the assembly section of the assembly line and vehicle. The system model can be built according to actual production with inheriting and duplicating objects. Practical application is also convenient. The corresponding object model can be generated with the mouse click and the drag of corresponding object icon according to the needs. The object model in menu item “window” of this software can be shown in Figure 3.4.



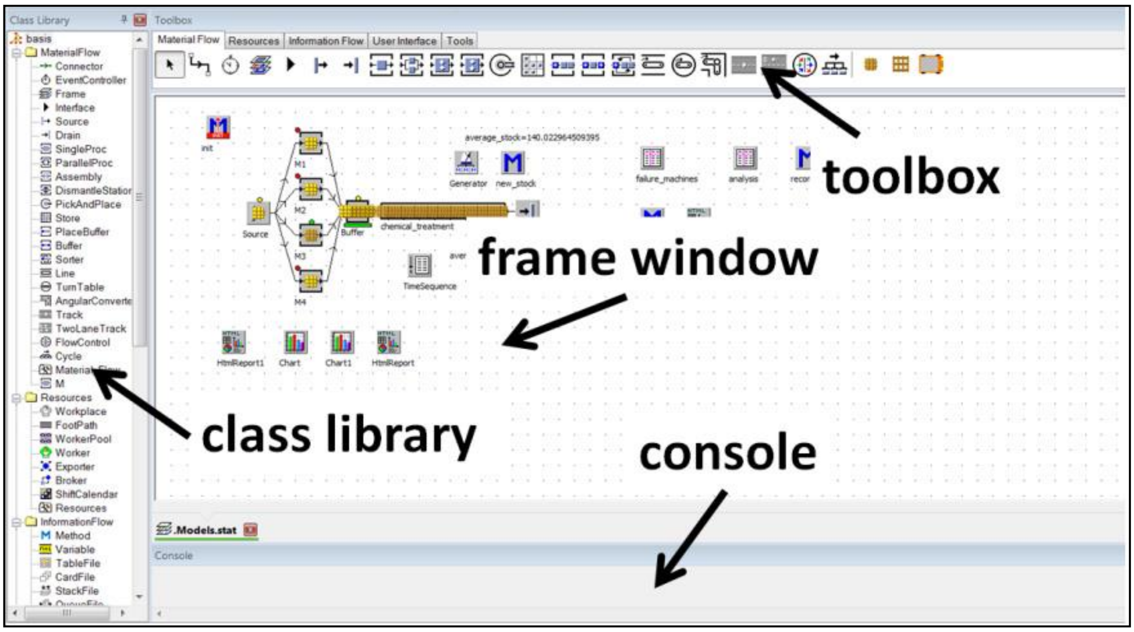


Figure 3.3 Menu item “Window”

Source: Plant Simulation (2012)

In menu item “Window” of Tecnomatix Plant Simulation, there are three important elements in menu “Dockable Window” as follows:

a. The Class Library: In the class library can be found all the objects needed for simulation. The user can create personal folders, get and duplicate classes, create frames or fill the objects from other simulation models.

b. The Console: Console provides information during simulation such as error messages when simulation runs. We can use the print command to output messages for the console. If we do not want a console, we can hide it.

c. The Toolbox: The toolbox provides quick access to classes in the class library. The user can easily create personal tabs inside the toolbox and fill with the personal objects that user's need.

The object libraries of the basic modeling that will be needed for the case study are described, as follows:

a. Logistics object: It has the ability to change the moving objects parameters of the object, including source, operator/machine, assembly workstation and drain as output.

b. Information flow object: This object is important to control and record the data of simulation, and most of the information flow objects are presented in the form of table, including the object of variables, table, storage file, trigger and circulation.

c. The user interface object: It is the method of the communication between simulation model and the user, which not only can provide system simulation of relevant information for users, but also can be a control simulation tool for users. The researcher is also use object "method" to control the assembly line when product has been available on workstation.

d. The moving object: It refers to the object that physical location is not limited to one place, such as the assembly parts of the assembly line, the vehicle and tray of transport parts. In this case, MU's object as products will be produced with different model type. The conveyor will be transporter for its products. Due to different products have different process, modeling assembly line according to the process. Under the condition of meeting the requirement of the process models according to the actual situation of the assembly lines. The type of MU's objects can be divided into colors. In addition, the parts that will be assembled have also different color. It is depending on the products. The purpose is to figure out the different model when process production in progress.

e. Testing problem: It is to figure out the assembly line performance. Based on the literature of MTALBP, actually there are two test problems are WSI and WLE. However, the software provides many tools to measure the performance. The researcher gives the throughput and total waiting time as additional for test problem.

Furthermore, the system model can be built according to actual production with inheriting and duplicating objects. The corresponding object model can be generated with the mouse click and the drag of corresponding object icon. In this step, extraction process information then arrangement order modeling into information flow object. The next step is create the object such as logistic object, moving object and testing problem that corresponding with actual production. The last is control the system by the user interface object.

3.5.3.2 The Design Method of Assembly Line Modeling

Assembly line modeling is the important role to implement of the real system. Many types of assembly line that must be selected. The software should be support with the assembly line type. In this research, the types of assembly line have different products and process, modeling the assembly line according to the process. Under the condition of meeting the requirement of the process models according to the actual situation of the assembly lines. It needs to base on the information resources. Firstly, the information of assembly line should be collected into the resource database of the assembly line. Furthermore, the information will be classified in the aspect of organization structure according to function. The classification of this treatment is easy for a corresponding actual assembly line in modeling.

Modeling is the process of constantly extracting Plant Simulation object library according to the information resource of the assembly line (Bangsow, 2016). In modeling process, each different model also has different condition. In addition, on both sides of the station side by side in parallel also arranged according to the type of task. It is because regardless of which kind of assembly lines for the products that assembled on two sides and different products have different processes. Therefore, the production process library restricts the arrangement and order of modeling in assembly line. Here is the specific design flow chart of assembly line by using the software can be seen in Figure 3.5.

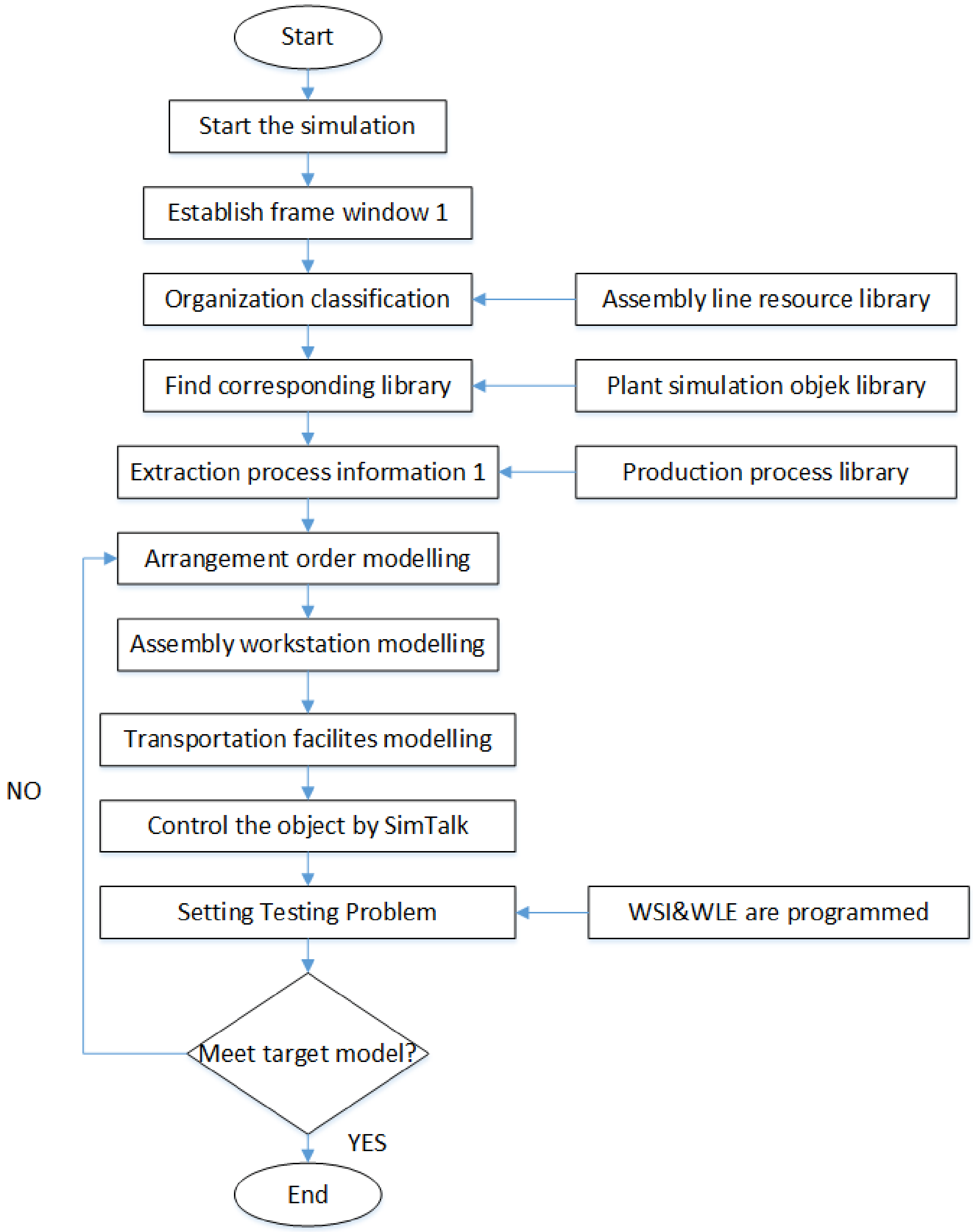


Figure 3.4 Flowchart of assembly line modeling design method

Source: Huang et al. (2012)

3.5.3.3 Modeling the Assembly Line Balancing Method

The operation of the model simulation can only provide feasible schemes under certain conditions and does not provide the optimal solution to the problem. The simulation assembly line balancing techniques that are generally implanted in the simulation process. It is the process of finding the best input variable value from a feasible domain making the output to be the optimal solution or satisfactory. The goal is to determine the most optimal variable decision when getting the most information in the simulation experiment. Therefore the user can decide easily (WANG & SHI 2013). Tecnomatix Plant Simulation has many optimization tools to create the assembly line balancing method, the researcher use programming language (SimTalk). It is one of the best optimization procedure. The purpose is to implement an objective function in this research by using this method. This can be an option to optimize and control the allocation of objects in the system. It is manifested in the following way:

A. Optimization Tool Equipment

The optimization tool used to represent problem solving is the "Method" object. This object provides settings of structures that can be described with programming languages. The procedure can expand how to model and control user simulations. Each object has built-in properties that provide many useful features when a model requires more detailed or completely different properties. Especially if the user wants to create a program to input data, calculate formulas, control objects and make optimizations. In the optimization of assembly line, this method will be described to balance the assembly line.

B. Design the Assembly Line Balancing Method Using Optimization Tool

The researcher uses the optimization program adopted from Bangsow (2016). The purpose is to represent the MTALB Type II problem. The function of this program is to allocate tasks base on the possible tasks that can be assigned to each workstation dynamically, flexible and optimally. The tasks will be allocated cannot break the constraints such as precedence constraint and assignment restrictions. The optimization program which described in the object "Method" can be seen in Figure 3.6 (details of the program code attached).

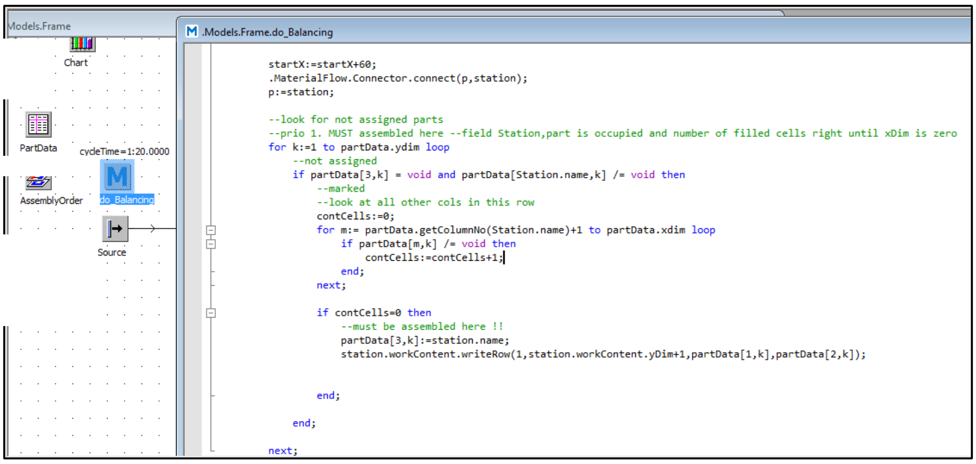
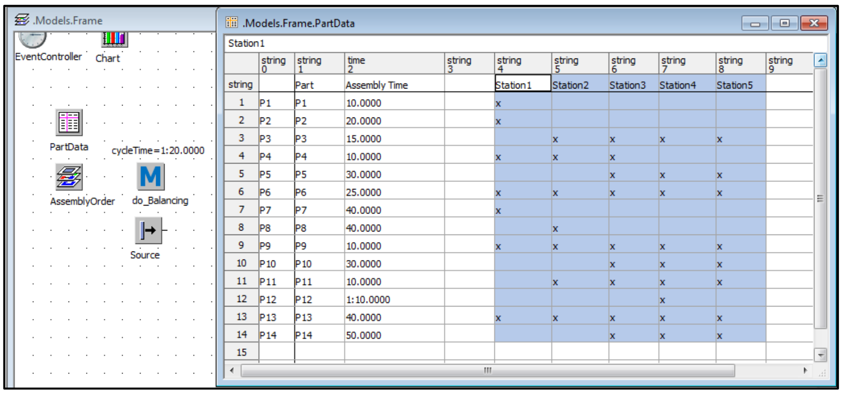


Figure 3.5 Optimization program of MTALB problem in object “Method”

In this procedure, work contents as task times that will be assigned into workstation. The task will be assigned based on cell assignments in the table. The function is to describe the task assignment that will be assigned to the right or left side in each station. It can represented by the mark symbol (x) in each workstation. The assignment restrictions are considered by transformed into the table. After that, the joint precedence diagram as precedence constraint is transformed to be the object which called assembly graph. Thus, work contents, cell assignments and assembly graph can be integrated with describe the program. The program code can be described technically by the following stages:

1. In the process of running the program, firstly, the old version of "Single.Proc" object as workstation and "Drain" object as output are removed first automatically.
2. All possible elements must be available in the table such as task, time and the number of alternative tasks that enter into the station. The following steps of the process of entering tasks automatically into the station, as follows:

- Assign the task to station alternatives can be marked with the symbol (x) in the "PartData" table. The symbol aims to consider the assignment constraint such as precedence constraints and assignment restrictions. It also can be describe right or left side assembly line. Examples of assignment process by using the symbol can be seen in Figure 3.7.



	string 0	string 1	time 2	string 3	string 4	string 5	string 6	string 7	string 8	string 9
string		Part	Assembly Time		Station1	Station2	Station3	Station4	Station5	
1	P1	P1	10.0000		x					
2	P2	P2	20.0000		x					
3	P3	P3	15.0000			x	x	x	x	
4	P4	P4	10.0000		x	x	x			
5	P5	P5	30.0000				x	x	x	
6	P6	P6	25.0000		x	x	x	x	x	
7	P7	P7	40.0000		x					
8	P8	P8	40.0000			x				
9	P9	P9	10.0000		x	x	x	x	x	
10	P10	P10	30.0000				x	x	x	
11	P11	P11	10.0000			x	x	x	x	
12	P12	P12	1:10.0000					x		
13	P13	P13	40.0000		x	x	x	x	x	
14	P14	P14	50.0000				x	x	x	
15										

Figure 3.6 The symbol (x) in table object "PartData".

- The program will check all the existing part positions in the table to match with the part in the "assembly graph" frame as precedence diagram in the simulation. If something is not appropriate, either from the wrong part name, mismatch and an inappropriate workflow, then automatically the program will notice that an error occurred. In addition, the program can ensure that all predecessors are assembled.

3. During line balancing process, the program gives command to look for task times with the smallest number of available task alternatives in the table. The next process is maximizing the possibility of task time in the remaining cycle time. It can be assigned flexibly according to the symbol (x) that has been marked in cell assignment of the table.
4. The stations that have been balanced, the station name will be copy into the "PartData" table column number four. After the station names are copied, the program will set to create an object "SingleProc" or station according to the available name then from all stations connected to the object "connector".
5. The task time will fill automatically into the object through the existing work content table on the "SingleProc" object. Examples of work content in that object can be seen in Figure 3.8.

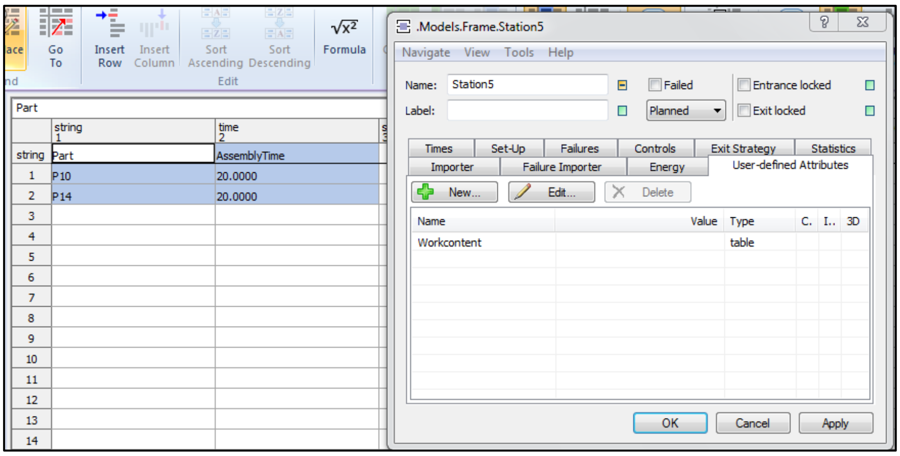


Figure 3.7 Work content

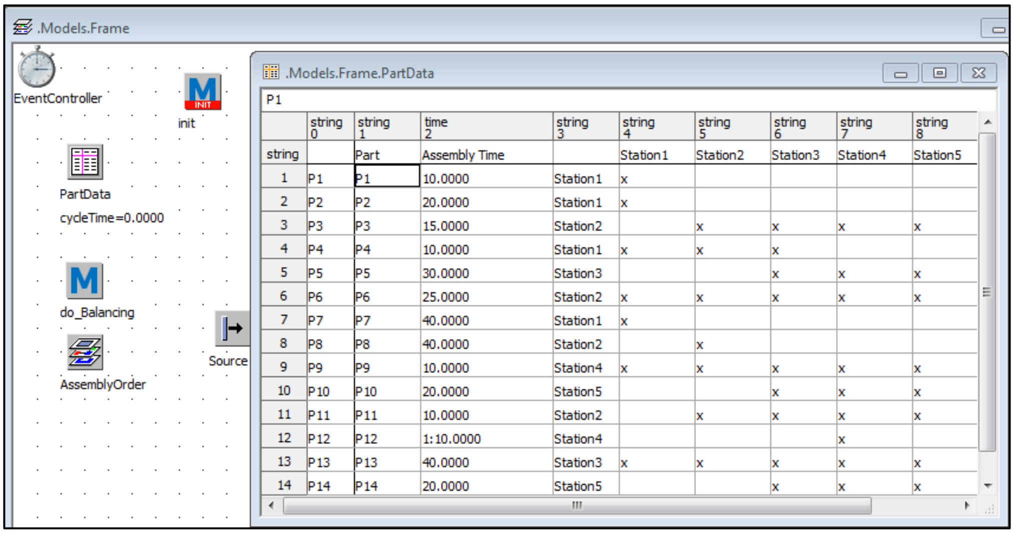
6. Connecting the last station with object "Drain" as output.

C. Assembly line balancing procedure

Here are how to apply the optimization method to the balance of mixed-model two-sided assembly line has several stages:

a. Input data

In this simulation, the researcher uses object "table" to enter the data. The part and task time can be inserted into the table. Assignment process can be done with the mark (x) in the cell assignment. Example table object "PartData" can be seen in Figure 3.9.



string	string 0	string 1	time 2	string 3	string 4	string 5	string 6	string 7	string 8
string		Part	Assembly Time		Station1	Station2	Station3	Station4	Station5
1	P1	P1	10.0000	Station1	x				
2	P2	P2	20.0000	Station1	x				
3	P3	P3	15.0000	Station2		x	x	x	x
4	P4	P4	10.0000	Station1	x	x	x		
5	P5	P5	30.0000	Station3			x	x	x
6	P6	P6	25.0000	Station2	x	x	x	x	x
7	P7	P7	40.0000	Station1	x				
8	P8	P8	40.0000	Station2		x			
9	P9	P9	10.0000	Station4	x	x	x	x	x
10	P10	P10	20.0000	Station5			x	x	x
11	P11	P11	10.0000	Station2		x	x	x	x
12	P12	P12	1:10.0000	Station4				x	
13	P13	P13	40.0000	Station3	x	x	x	x	x
14	P14	P14	20.0000	Station5			x	x	x

Figure 3.8 Table object "PartData"

After the tasks that have been entered in the table above, then created a sequence of assembly or workflow with predecessor and successor on the task by using object "SingleProc". The users are required to create new frames first in the Class library window. The assembly sequence commonly called assembly order can be seen in Figure 3.10.

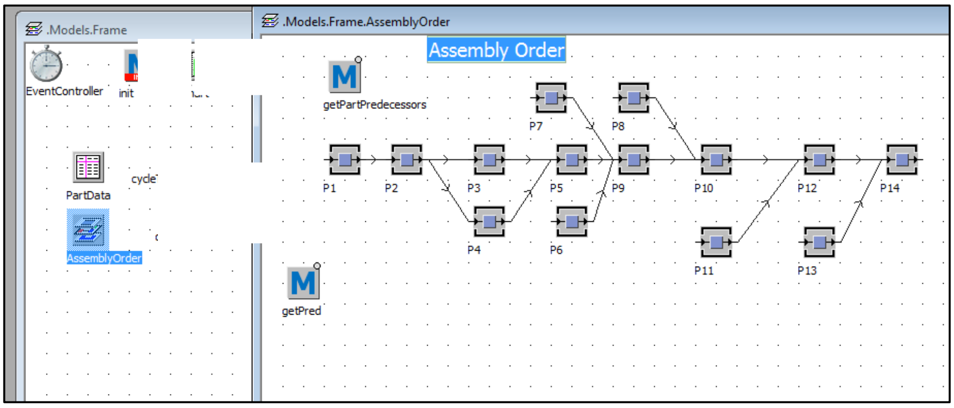


Figure 3.9 Assembly order

After that, the object "Method" is created in the AssemblyOrder frame. The purpose of the program is that the sequence of tasks in the AssemblyOrder frame with tasks that exist on the "PartData" table object can be connected. The program works to get or read all predecessors on assigned tasks and can check the same or conflict tasks if errors occur when creating workflows. The following examples of programs that appropriate with the goals can be seen in Figure. 3.11.

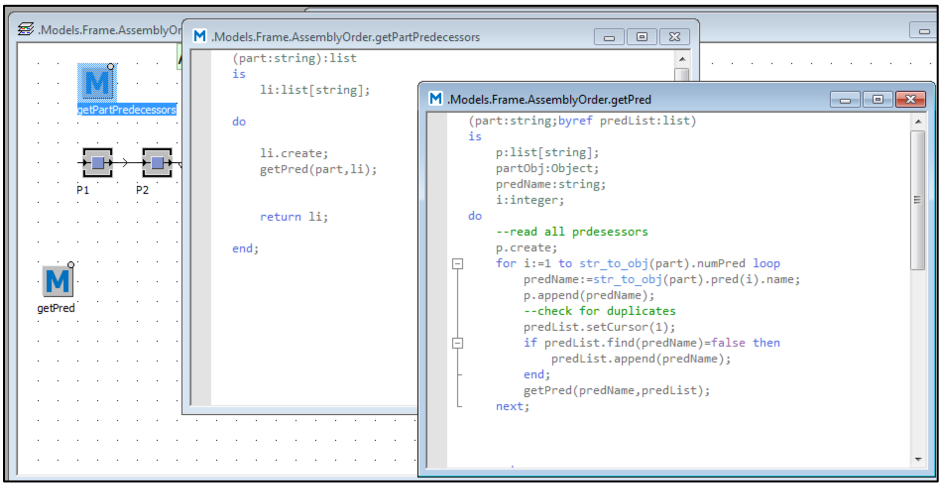


Figure 3.10 The method program on frame object “AssemblyOrder”

b. Process of line balancing

Balancing process can be done if the objects have been met requirements such as entering data tasks, workflow, task time and especially line balancing optimization program. Before the optimization process, determine the cycle time than right click on the object method "Do_Balancing" after that click run. An example of running the line balancing method can be seen in Figure 3.12.

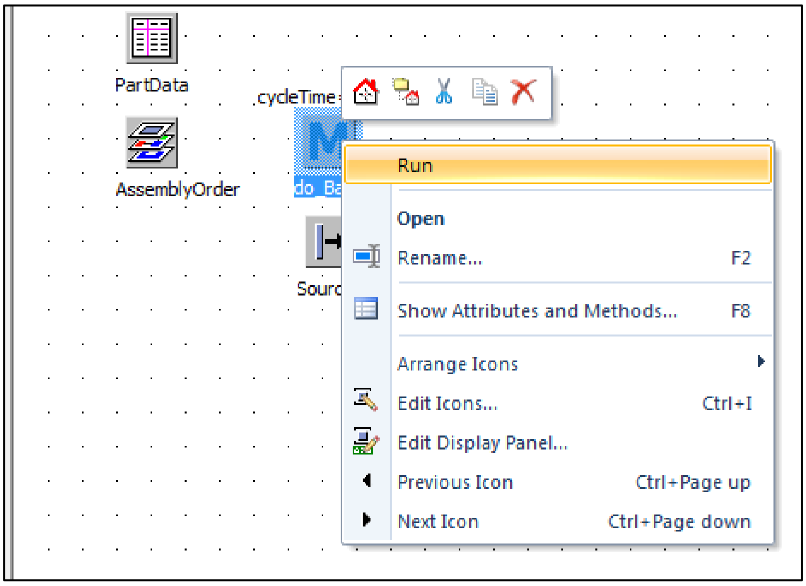


Figure 3.11 Running the line balancing method

The condition of assembly line after running the line balancing method can be seen in Figure 3.13.

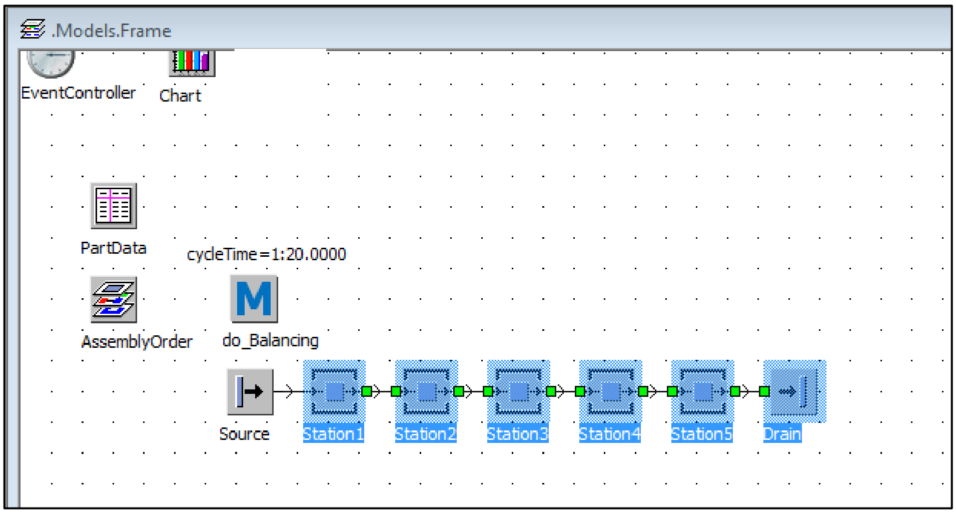


Figure 3.12 The new station after balanced

In the figure shows the new station objects and the number of time stations that have been adjusted to a certain cycle time. Line balancing results can be viewed using the "Chart" object. The function of this object is to show the graph at the time of each station when the production process is running. Simulation results on the "Chart" object can be seen in Figure 3.14. On the graph shows the amount of time in each station is 1 minute 20 seconds. These results show that the assembly line is already balanced.

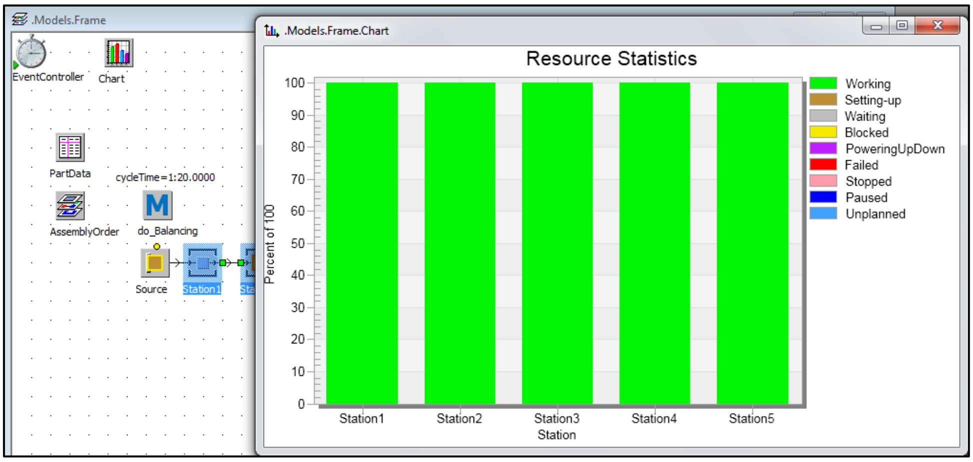


Figure 3.13 Simulation Result on object "Chart"

3.5.3.4 The Programmed Object "Method"

In simulaton model, there are elements that will be programmed to control the objects and calculate the measurement results of some equations that are described. The elements will be programmed using the software as follows (code of the program in detail attached):

a. The setting of parts that have been assembled

   MU's part in Object "source" will exit if the product in the next station available and ready to be assembled, the setting can use the program as follows: (Attached).

b. The listing summary of station times

   To see the station times at each station, it is necessary to set the program time so that the task time can be summarized in a single "Table" object.

c. Tasks placement

Time stations that have been summarized into one table then placed to the table on each station through the program.

d. The overall proportion of units produced (qm)

Programmed from Equation (1).

e. Cycle time constraint

Programmed from Equation (1).

f. Calculation of task time total

The task time on each model is summed to get the total number of tasks in each model.

g. Waiting time

Programmed based on the calculation of cycle time minus each station time.

h. Weighted line efficiency

The simulation result uses the parameter of WLE calculation from Equation (4).

i. Weighted smoothness index

The simulation result uses the parameter of WSE calculation from Equation (4).

3.6. Discussion

After the system had been build, then discussion will be conducted to discuss the problem formulation and the result of data processing. Discussion will provide a performance analysis of current situation and proposed model the MTALB.

3.7. Conclusion and Recommendation

This section will briefly answer the question stated in problem formulation formulated in chapter 1. Besides, this section also give some recommendations that might be used to further research.

3.8. Research Flowchart

The research diagram is used to solve problems. Research diagram explain the steps of conducting research from the beginning until final result. The research diagram can be seen in Figure 3.15, as follows:

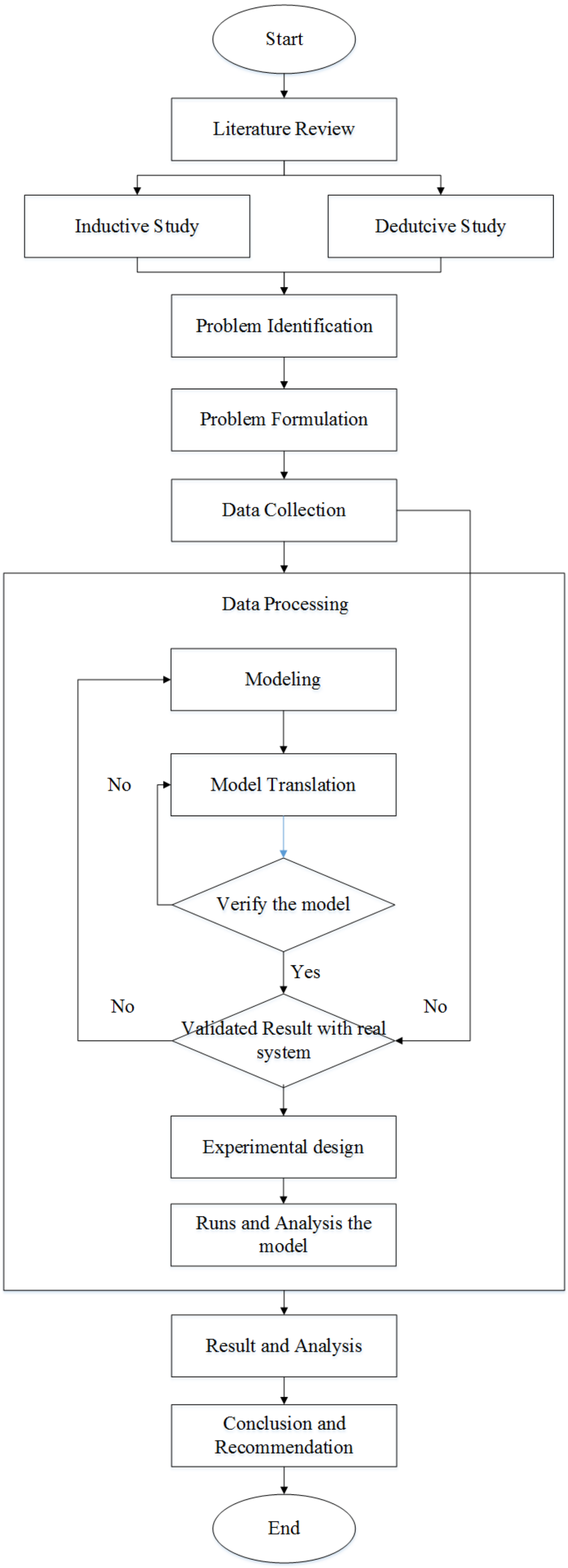


Figure 3.14 Flowchart process of research

Based on the figure above, the outline of creating a simulation model consists of nine steps as follows:

1. Problem Formulation: In order to begin a simulation, it is crucial to define the problem in detail and to formulate in a way that also the reader can understand.
2. Data Collection: If the aim is to get accurate results at the end, then we have to collect data in detail and use them correctly as an input. Both at the beginning and during creating the model, it is efficient to get data as much as possible. For instance, for a manufacturing process, the data can be cycle times, failure frequencies, number of defected products, some probability equations and so forth.
3. Defining objectives and building a plan: The objectives should be well defined before beginning the simulation. In addition, the plan which can guide us during the project also must be built.
4. Creating the Model: In order to avoid wastes in terms of time or monetary, it is beneficial to create a simple plan at first stage and improve it continuously.
5. Verification and Validation: In brief, validation puts forward a question. Is it the right model? It investigates if the created model compatible with real process. In contrast, verification is interested if we build the model in the right way.
6. Experimental Design: It is also important to try different scenarios and to observe the length and number of runs in a model. The more simulation run brings better accuracy.
7. Production Runs and Analysis the model: Different analysis has to be made during simulation runs in order to see the model performance. Different statistical methods and theories are used in this step.
8. Result and Analysis: The result is obtained from simulation model. Furthermore, it can be analyze how the performance is after get the result.

3.9. Expected Result

By conducting the research, following results are expected:

a. A final assembly process can be obtained from the model built by using Tecnomatix Plant Simulation tool that is applied, where the performance of assembly line is better compared to the previous.

b. The proposed optimally of simulation can be beneficial for the company as it has better performance and more feasible, as it a concern to objectives directly rather than one.