AN ABSTRACT OF THE DISSERTATION OF

Weifeng Huang for the degree of Doctor of Philosophy in Mechanical Engineering presented on December 14, 2017.

Title: Applying Multi-Objective Evaluation to Automated Assembly Planning in Early CAD Design Stage

Abstract approved:

____

Christopher Hoyle

Successfully predicting an accurate estimated cost is important in the assembly planning process. When designing an assembly plan, an accurate estimation ensures that the proposed plan can be achieved within the predetermined budget. However, achieving an accurate prediction is a challenge since it requires professional judgment, which is dependent upon the previous experiences of those conducting the estimation. As the scale of the production process increases, making correct estimations for each step in a process is difficult. Although many design tools have been developed to shorten the design process, the focus has shifted to achieving *automated assembly planning* with large scale CAD models in the early design stage. Traditional design tools are less suitable for this task because they require a significant amount of human interaction, and are not very adaptable to 3D CAD models. This PhD dissertation introduces a new *automated design tool* that can help a designer estimate assembly time

and stability for assembly planning process, to support cost estimation and increase the efficiency of the product (or system) design. The first step of this research is to conduct experiments for assembly time and stability evaluation, and this field data is used for model development and validation. Second, a machine learning method is applied to predict the assembly time based on the tessellated model; the results indicate high accuracy compared to the traditional design for assembly (DFA) time estimation method. Third, two novel approaches, a theory-based and a physics-based approach, are created to evaluate assembly stability during the assembly process. Finally, a multi-objective evaluation function that includes assembly time, stability, and predicted uncertainty is applied to the automated assembly planning process. With the implemented tool, engineers can easily evaluate assembly plans that can accommodate actual production environments with lower cost.

Applying Multi-Objective Evaluation to Automated Assembly Planning
In Early CAD Design Stage

by
Weifeng Huang

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented December 14, 2017
Commencement June 2018

Doctor of Philosophy dissertation of Weifeng Huang presented on December 14, 2017

APPROVED:

Major Professor, representing Mechanical Engineering

Head of the School of Mechanical, Industrial and Manufacturing Engineering

---

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Weifeng Huang, Author

ACKNOWLEDGEMENTS

I would not have achieved so much without the support of my family and friends. Especially, my research advisors Dr. Chris Hoyle and Dr. Matthew Campbell; two of my life changers that not only providing me with guidance and patience for my research, but offering me enormous opportunities and trust so I can achieve my ideas. Also, I really appreciate that you always concern about my family and my feature career. It is my honor to work with you and I will never ever forget that.

I would like to express my gratitude to Dr. Onan Demirel for supporting the stability survey in this research. Also, I want to thank Dr. David Kim and Dr. Louis Joseph for serving as my PhD committee members.

Additionally, to my good friend and colleague, Nima Rafibakhsh, I really enjoyed working with you in this project and all of your help will be remembered.

Finally, to my wife, my soulmate, Meiling: The luckiest thing that happens in my life is having you be my side. Your understanding, your support and your love make me become fearless and helped me go through every challenge. I love you. To my parents, I am very thankful for all your supports and cares. Without that, it would be impossible for me to keep pursuing my dream. I also want to thank my sister Biyu and my friend Chao. At last, to my son, Christopher, thanks for all the joy that you bring to me and waking me up very early in the morning every day.

CONTRIBUTION OF AUTHORS

Nima Rafibakhsh was involved in the development of primitive classification, components free direction and assembly planning tree search method.

Yue Liu was involved in the securing time data collection and design of assembly experiments.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF TABLES

CHAPTER 1 : INTRODUCTION

Digital manufacturing has become more prevalent in recent years, enabling assembly processes to be simulated early in the design process so that assembly cost can be estimated before committing to production processes. Making such estimations is meaningful for planning assembly process procedures, especially because up to 85 percent of the manufacturing cost of a product is committed during the early design stage [1,2]. Assembly planning is a critical stage in product design and manufacturing, which consists of multiple tasks including sequence planning, facility planning, and assembly tool and fixture planning [3]. Hence, assembly planning requires precise cost estimation so that potential process designs can be evaluated. Also, as the focus shifts to Automated Assembly Planning (AAP) in industry, many methods for planning and evaluation have been introduced to reduce the burden of manually creating and evaluating assembly plans. However, the challenge of creating a fully automated approach for evaluating assembly cost has not been completely addressed.

The focus of this research is to design a fully automated assembly sequence evaluation method to estimate the assembly time and stability, considering various sources of uncertainty, using a *tessellated* assembly model. A tessellated model represents the surfaces of a solid part by using triangles faces, containing information about the vertexes and edges. Since the tessellated model is independent from the commercial CAD software, the cost of using a tessellated model is reduced significantly over using commercial CAD software. Using the proposed methodology, the production efficiency can be increased by pushing the assembly cost estimation to the early design stage.

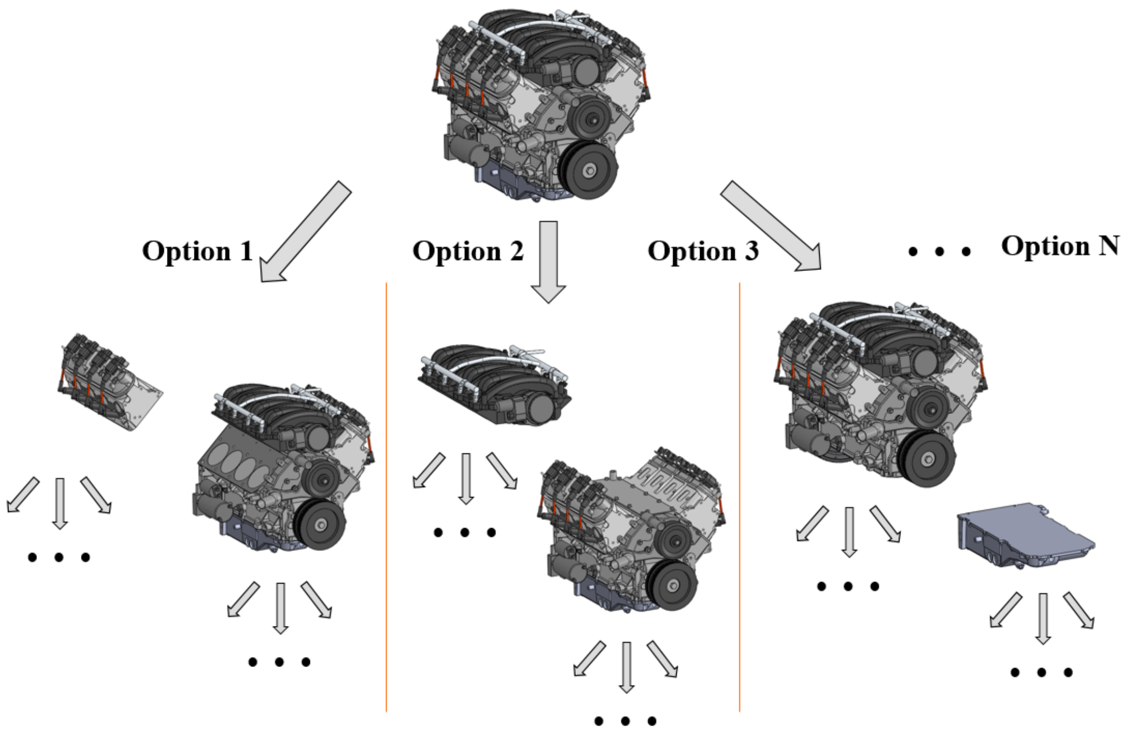


Figure 1.1: An example of applying tree search for disassembling the engine. Meanwhile, the assembly cost for each of the candidate options need to be evaluated automatically

In this research, *assembly time* and *stability* are the two factors that we consider to have the most influence on designing the assembly plan. Hence, automated evaluation approaches are developed to estimate these factors for assembly sequence optimization. The method for generating assembly sequences is called *assembly by disassembly*, with an example of this process illustrated in Figure 1.1. To start the method for a given assembly, an assembly by disassembly approach will generate all valid options for disassembly of the assembly into two subassemblies; this decomposition process continues until the point when no subassembly can be further disassembled. The assembly process can then be obtained as the reverse sequence of the disassembly process. All the options that are generated during this process represent feasible assembly tasks; however, the sequence of tasks will vary by assembly times and the stability of the intermediate subassemblies throughout the process. The scope of this research is to develop an automated method to evaluate the assembly time and stability for the generated tasks for optimization tree search.

The *time prediction model* is designed based on the Design for Assembly (DFA) method, but formulated to work directly with CAD data and using a Gaussian Process (GP) Model to automatically estimate time for a given assembly. The model considers a single assembly operation as comprised of four kinds of motion: *Move*, *Install*, *Secure* and *Rotate*. For each of the motions, the Gaussian Process model is applied to develop a time prediction model. The reason for using a GP model is not only that it is able to estimate accurate assembly times, but it also can provide predictive confidence levels for the estimation which can be used to generate more robust assembly plans. To gather data to build the time estimation model, *experiments* were conducted by assembling real products to gather assembly times for a variety of assembly operations, including move, install, secure and rotate times. The developed GP model is compared to established approaches in the literature to ensure its reliability. An *online* updating approach is also implemented for the GP method, for continuous model time updating and improvement of the prediction model.

Assembly *stability evaluation* is also integrated into the assembly planning search. In this research, an automated evaluation method that can evaluate the assembly stability for tessellated CAD models is implemented. To conduct such evaluations, first, the physical part constraints are classified for each component in the tessellated model. Subsequently, the assembly is evaluated by both *theory-based* and *simulation-based* stability approaches. The theory-based stability approach involves detecting the part degrees-of-freedom (DOF), as well as tip and slide difficulty. These factors are calculated based on the tessellated part geometry and kinematic constraints. Unlike the theory-based stability based upon static analysis, the simulation-based stability is evaluated based on the simulated kinematic behavior of the assembly during the assembly process, using a physics simulation. By carefully defining the kinematic constraints, the spatial movement of every part within the assembly is simulated by a physics engine. This spatial movement analysis is used to capture the velocity of the assembly parts, which is the basis for evaluating systematic stability. To validate the model-based approaches, an *experiment* is conducted to study human evaluation of assembly stability based on 3D printed assemblies. The experiment results are compared to the two proposed methods to ensure these approaches produce evaluations that are similar to a human to validate the model-based approaches.

Finally, a multi-objective approach that includes *estimated time* and *stability*, while considering model prediction uncertainty, is formulated for assembly planning optimization. This multi-objective approach also contains adjustable weighting parameters for designers to determine the balance between time and stability of the final generated assembly plan.

1.1 MOTIVATION

Assembly cost evaluation is one of the most important factors for achieving optimal assembly planning. During the generation of an assembly plan, each of the candidate subassemblies needs to be evaluated in terms of assembly cost. Based on the estimated cost, designers can decide the sequence of assembly tasks to use in production. To provide this estimation, many design tools [4–10] have been developed to estimate different assembly costs for manual assembly operation.

These costs can be categorized into assembly time, complexity of the assemble action, etc. However, when the scale and the complexity of assembly increases, the cost evaluation becomes more difficult for the designer. For this reason, the AAP method has been developed to help designers generate assembly plans by increasing the automation of the planning process. Since 3D CAD models have been used heavily in AAP, the cost evaluation process becomes more adaptive to the AAP method by introducing CAD-based assembly cost estimation earlier in the design process. For example, Boothroyd Dewhurst FDMA software [11] is one of the most famous estimation methods that can help engineer to determine part assembly time and labor cost based on its CAD dimension and material. In another research approach [12–14], constraints between components in a CAD model are extracted for studying the complexity of assembly task.

Although many CAD tools have been developed to increase the efficiency of assembly cost evaluation, there are still limitations that restrict the broader usage of AAP on complex assemblies. First, human interactions are still required during the evaluation process. To the best of author's knowledge, there is still no design tool that is able to produce fully automated assembly task evaluation with CAD models; this means engineers still need to provide input information about the product design for assembly planning. This is also related to the second limitation: the process of extracting CAD part geometric features is not automatic. In product-based evaluation method, part handling and install difficulty are the two key factors and used for estimating assembly cost. These two factors are highly related to the component geometry; however, the implementation of obtaining such features automatically is complicated, since most of the CAD models are dependent upon commercial kernels such as Parasolid and ACIS. Working on these kernels requires professional knowledge and the costs of these licenses are expensive. Thus, this increases the difficulty of APP development. Finally, most of the assembly cost models are determined based upon real products, and are difficult to modify and adapt to APP implementations. For example, assembly time is one of the most significant costs considered in assembly planning evaluation, since it is directly related to production scheduling. However, most existing models are developed based on physical assembly experiments, and measurements such as motion complexity and product detailed geometric are difficult to obtain from the CAD environment. Hence, for CAD assembly for AAP, new cost evaluation methods are needed.

1.2 OVERVIEW

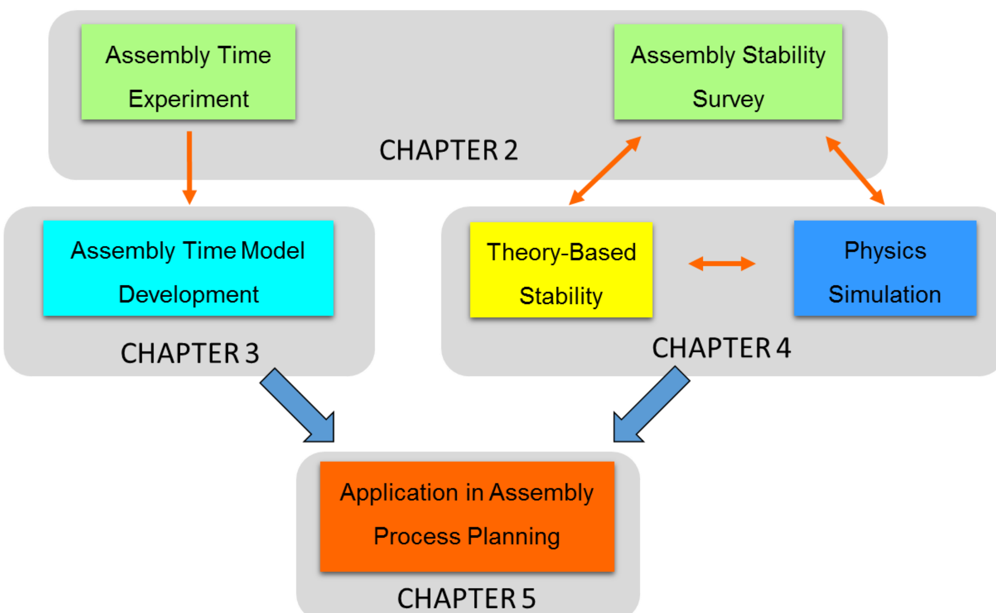


Figure 1.2:Content of the dissertation

This PhD research can be divided into two major themes: automated assembly time estimation and automated assembly stability estimation. The general flow of these two themes throughout the dissertation is shown in Figure 1.2, with their related chapters. First, experiments for collecting field assembly time data and the experiment for evaluating assembly stability are described in Chapter 2. In Chapter 3, the Gaussian Process model is applied to develop the assembly time model with uncertainty prediction by using the field data. This model is also compared with other methods for the purpose of validation. For stability estimation, two different approaches, theory-based and simulation-based stability, are implemented in Chapter 4. Meanwhile, a comparison between the results from these two methods, and the survey data that reflects human perception of assembly stability are included in this chapter. Chapters 3 and Chapter 4 are nearly identical to journal submissions and as such are somewhat redundant with the text in this chapter. Finally, a multi-objective function that includes time, predictive uncertainty and stability is applied to the automated assembly planning process and the results will be discussed in Chapter 5.

CHAPTER 2 : EXPERIMENTS FOR ASSEMBLY TIME AND STABILITY

2.1 INTRODUCTION

To develop a novel assembly cost estimation model based on tessellated model, data that can indicate the relationship between assembly cost and tessellated model features is needed for model development and validation. However, the tessellation-based cost modeling method have not been studied before. Thus, a Lack of empirical data is one of the challenges in this research. Although currently existing methods are able to produce estimated costs such as assembly time and stability, the configuration of these methods is not compatible with the STL and not adaptive to the AAP process. Hence, experiments are required in this study for gathering usable data. There are two experiments conducted in this study: one for predicting time and one for predicting stability. In the assembly time experiment, we consider each of the assembly operations includes move, install, secure and rotate actions. The move action represents the transportation of a part or subassembly between different stations. After transportation, install action is used to describe the installation of the subassembly. Meanwhile, secure action is required if there are any fasteners within subassembly; at the end, if the finished subassembly has poor accessibly for the next operation, rotation of the product is needed for preparing the next assembly operation. The time for each of these actions is measured individually for action time model developments and the assembly time for each assembly task can be calculated by the summation of the predictive action times. The input variables that correlate to the action times are selected based on their measurability on real components and tessellated models. Good measurability is a key factor in this experiment and it can ensure that the variables can be physically measured. Meanwhile, they can also be extracted from the tessellated model automatically while the automated evaluation is applied. Design of experiments (DOE) is included in this experiment to optimize the experimental setup and ensure the field data is sufficient enough for developing an accurate prediction model. Three different products are assembled in this experiment. Meanwhile, the action time is collected for model development.

Different from the assembly time experiment that the collected data is used for surrogate model development, the result from the user survey about assembly stability evaluation is used to validate the computational method for estimating the tessellated model stability. This survey includes two kinds of evaluation: evaluation of external stability and evaluation of internal stability. Meanwhile, each of these evaluation is based on 3D printed assemblies and their images.

2.2 LITERATURE REVIEW

2.2.1 Assembly Time Experiment

Experiments are important for time prediction model development. Many approaches for time estimation have been developed and derived from the Predetermined Motion Time Systems (PMTS) and Design for Assembly (DFA). Both of these approaches require assembly experiments for collecting human subject data to study the effect of the assembly motion and component geometric complexity on assembly cost. PMTS is a motion based method that divides the entire assembly operation into basic human movements and classifies each of them based on the movement, such as grasp, place, reach, etc. [15]. By assigning time to each of the movements, the assembly time can be calculated. Methods-time measurement (MTM) is one of the earliest methods for time prediction based on experiment, developed in 1948 [4].The development of MTM is based on studying films of assembly operations which were performed by qualified workers on a shop floor. Later, the Maynard Operation Sequence Technique (MOST) was developed with higher efficiency for time estimation [5]. Different than the MTM method in which the experiment is focused on capturing the time for each of the single motions that the worker preforms during assembly task, MOST is more focused on sequence operation models which include general move, controlled move, tool use and manual crane[5]. For each of these sequential operations, experiments for setting standard times are also conducted under different movements.

The Design for Assembly (DFA) method is a product based method used for evaluating an assembly process. One of the earliest DFA method is developed by Boothroyd, Dewhurst and Knight [10] and it uses physical features of the assembly parts for assembly time prediction. To develop the relationship between assembly time and component geometric features, different assembly processes are filmed so the operation time can be obtained in a more accurate way. After gathering assembly times, regressions models for predicting handling, install and secure time are developed with the associated component geometric.

Due to the increasing usage of CAD models in assembly planning, new DFA methods are also developed that use 3D CAD models for assembly planning. Meanwhile, different experiments are also conducted for collecting usable data for DFA with 3D CAD model. In Owensby's study[13], assembly experiment is conducted with Solidworks software to simulate the assembly process. In this experiment, participants conduct assembly operations by adding mates between 3D components and the assembly time is recorded. At the end, the experimental data is used for developing the model to represent the relationship between mating complexity and assemble time. In Cho's study[16], a time estimation method which combines motion-based and product-based method is developed to evaluate the assembly process during product design stage. Experiments are conducted to simulate the assembly line in the factory with typical layout factors setup for assembling the assigned commercial product. All of these studies provides good examples of conducting experiments for assembly time acquisition; however, their applications on tessellation-based assembly planning are not mentioned. Thus, it requires new experiment to obtain data that can adapt to the automated assembly planning process with tessellated model.

2.2.2 Assembly Stability Experiment

Unlike the process of assembly time estimation, which requires experiments for assembly time acquisition, the stability estimation models are more dependent on the physical connection between connected components in the assembly. For example, many study [17–19] have been using predetermined stability index (SI) for stability evaluation. In general, SI is calculated based on the degree-of-freedom and types of connection such as attach, force-fit, screw, connectors and so on. Although such methods can accurately describe the physical connection between the component without experimentation, the amount of human interaction is not suitable for automated assembly planning.

More recently, virtual reality (VR) techniques have been applied to simulate the assembly process. During simulation experiment, stability can be also evaluated in a more dynamic process with different simulated environments. In Aleotti and Caselli's study[20], a physics-based simulation is conducted with human subjects to study the stability during product disassembly process. The participants performed different disassembly tasks that are based on the designed sequences and the stability level of each of these tasks is evaluated.

2.3 EXPERIMENT FOR ASSEMBLY TIME

In this section, move, install, secure and rotate actions are introduced to represent the motions that an assembler needs to perform during the assembly operation. Such operations are tested in both workbench and modular assembly stations to simulate the assembly environments. New model variables that are used for estimating assembly time are introduced in this section. Compared to the transitional time estimation model parameters, the introduced variables are more suitable with 3D CAD models. Design of experiments (DOE) methods are used here as guidance for designing the experimental tasks. Finally, the collected data for each of the actions is processed and the correlations between model variables and responded time and data variations are discussed.

2.3.1 Assembly Action Times

In Boothroyd, Dewhurst and Knight's study [10], the manual assembly actions for small parts include tool acquisition, part handling, and insertion. Based on this study, we introduce four assembly operations, move, install, secure and rotate, for time estimation as shown in Figure 2.1. The reason for dividing an assembly operation into these four actions is to estimate each as separate phenomena which may or may not be included in every assembly step. For example, if a subassembly does not contain any fastener, the prediction of secure time can be excluded from the overall assembly time. With this flexibility, the purposed time estimation method can be adaptive to different manufacturing environment and produce accurate predictions.

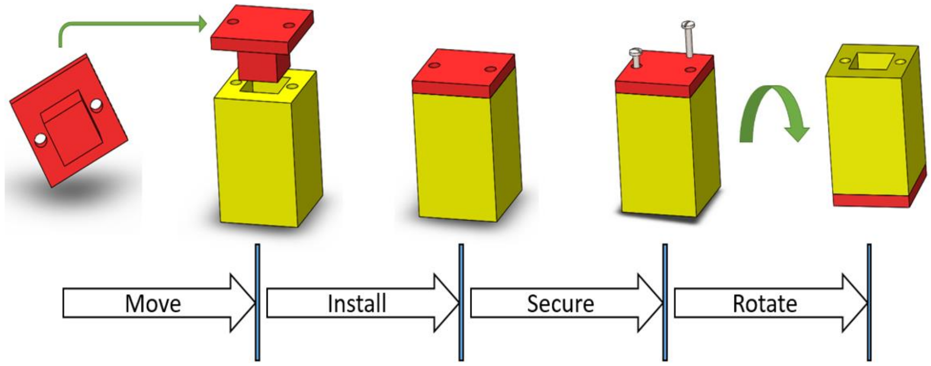
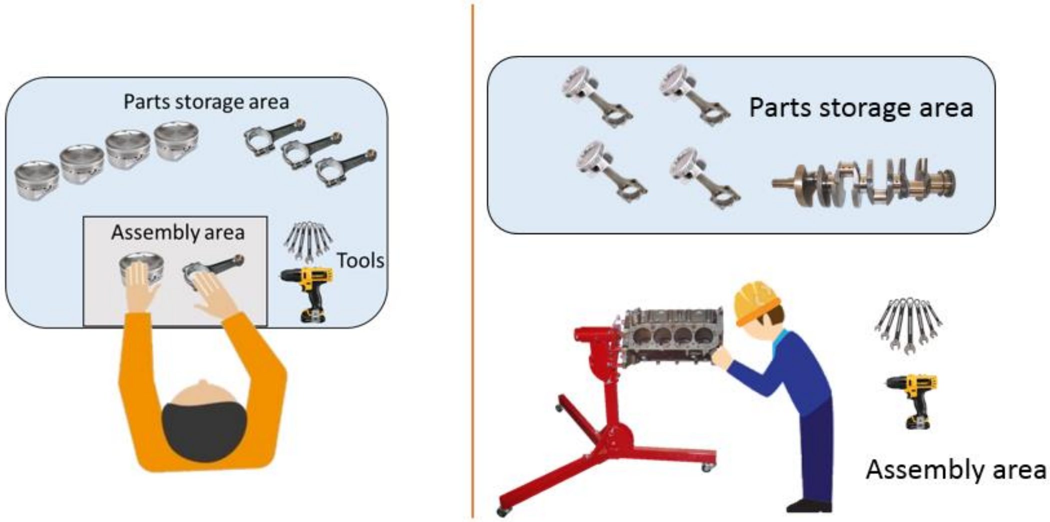


Figure 2.1: Four assembly action: Move, Install, Secure and Rotate

To achieve more realistic results, we apply two basic assembly station setups in this experiment: workbench assembly station and modular assembly station. The general illustration of these two setups is shown in Figure 2.2. The use of these depend on the scale of the product that is to be assembled. For an assembly with reasonable amount of components that are less than 5 pounds or 12 inches[10], workbench assembly station is applied . In this setup, all the components are placed on the table within reachable distance and the participant can finish all of the assembly task in the same spot. When the weight and volume of the components become larger, a modular assembly station is used and the worker needs to perform major body movements such as turning, bending and walking to complete the assembly operation. By considering both, the collected field data is able to cover different scales of assembly tasks. In addition, moving time, installation time, securing time and rotation time are independently captured in this experiment and models are developed for each.



(a) (b)

Figure 2.2: Two assembly environment setups, (a) workbench assembly station and (b) modular assembly station, are used in this experiment to simulate assembly task that without and with worker major motion (Images [21–23] from internet are used in this figure)

Moving time is the time starting from when a human operator starts to acquire a part to the part is arrived at the pre-install position. In this research, we consider the components are only transferred by hand without any usage of mechanical tools. Also, only one component or subassembly is acquired at each assembly step. Furthermore, we also assume that when a part is in the pre-install position, it is already oriented in the proper position for insertion. A moving operation includes grasping, transporting and rotation. Hence the moving time is dominated by the physical size, weight, and the travel distance of the moving part. The model input parameters for predicting moving time are included in Table 2.1.

Install time is defined as the time it takes to install a part from the pre-install position to its final location. The installation includes aligning and insertion actions. To capture the effect of alignment difficulty, alignment features are introduced in the prediction model. Alignment features are a subset of the connecting surfaces between a pair of mating parts in the assembly; the details of how they are measured will be discussed later. Also, it is assumed that the clearances between all of the connected parts are uniform since the tessellated model does not contain any information of clearances.

Securing time is the time needed to secure or fasten two or more parts after installation. There are several kinds of methods to secure parts, such as snap fits, tight fits, welding, etc. Due to the particularity and variety of securing methods, the combination among various fastening method would influence the securing time. Since threaded-fasteners are commonly used in industry for securing parts, fastener securing is considered as the only securing method in this study.

Rotation time is the time that worker spends on rotating an assembly to the orientation where the assembly is ready for the next assembly operation. This rotation occurs to the fixed or reference part in the subsequent install step. It is important to include rotation because many assembly tasks require parts to be installed from different directions and the reference assembly may need to be rotated to ensure the new part can reach its target location. Also, rotating an assembly can provide workers higher accessibility and visibility for installation. Like the secure operation, rotation may not be necessary if a new part can reach its target location unobstructed.

2.3.2 Variables Selections

After defining the move, install, secure and rotation actions, features that are correlated to each of these actions are selected as the input variables for developing prediction models. While including more related features can result more realistic estimation, the quantity of the features needs to be controlled since inputs must be derived automatically from the tessellated models. Meanwhile, as the number of input features increases, the cost and time for conducting experiments and collecting data will also increase significantly. For the experiments in this study, all of the data collecting tasks are done manually. So the number of input parameters is limited to a reasonable amount for experiment cost reduction propose. Approaches for obtaining the variables from tessellated models are described for providing a better understanding of how the physical measurement data is transformed into computational model inputs.

2.3.2.1 Moving Distance

For the moving time model, the moving distance is the distance between the original position of the part and its pre-insert position. This parameter can indicate the distances that the worker needs to move to acquire components for installing. Also, these distances vary in different layout designs. For the workbench assembly station in Figure 2.2(a), the part moving distance is between the parts storage area and the assembly area which is within the table and this layout does not require major body motions, workers can finish assembling in the same station without moving. For modular assembly station in Figure 2.2(b), workers need to travel to the parts storage for acquiring parts for assembly and moving distance becomes longer. Hence, introducing moving distance can capture whether a moving task involves major body motion or not, and this is important for predicting the moving time. Furthermore, once the factory layout is designed in the CAD environment, the moving distance can be automatically extracted by using the locations of different assembly section.

2.3.2.2 OBB volume and Maximum and Minimum OBB face areas

The Oriented Bounding Box (OBB) is defined as the smallest box that encloses the assembly part, introduced by Gottschalk [24]. The volume, maximum and minimum areas of the OBB are used as input parameters for move, install and rotate time models.

There are three reasons that OBB volume is selected instead of the exact part volume. First, during assembly operations, the contact points between component and hand have a higher chance to be located on the boundary surfaces of the OBB. For example, two parts with their OBBs are shown in Figure 2.3. When assembling these two components, the contact points are more likely to be on the outer surfaces. Hence, using the OBB volume can approximate the handing difficulty than using the actual part volume. Second, OBB is obtained automatically with the TVGL C# library [25] which can simplify the implementation of extracting inputs from the tessellated model for the cost evaluation. At last, the OBB for real components can be easily measured and generalized by three measurements: length, width and height. This is also important in the physical measurement of these parameters during experiments for data gathering.

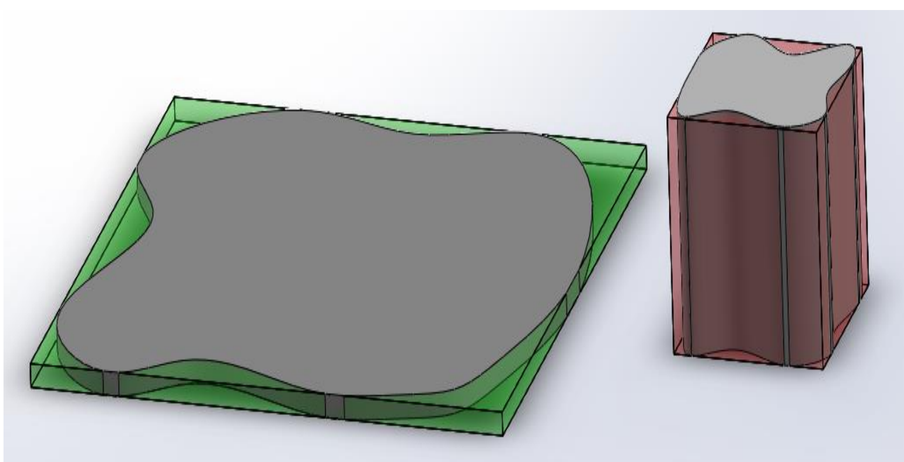


Figure 2.3. Two OBBs have the same volume but different handling difficulty

When parts that have that similar OBB volumes, the lengths, widths and heights for those OBBs can be different and can also affect the handling difficulty. For example, in Figure 2.3, the OBBs for two different parts have the same volume and placed on ground. Compared to the part that has a red OBB, the part with a green OBB is harder to handle since it is thinner and more difficult to be picked up. Two variables, maximum and minimum OBB face areas, are used to capture this difference and are defined below:

$$A_{max} = l_{max} * l_{mid} \tag{1}$$

$$A_{min} = l_{min} * l_{mid} \tag{2}$$

where l_{max} l_{mid} and l_{min} are the maximum medium and the minimum lengths of the sides of an OBB. With larger A_{max} and smaller A_{min}, a part is more likely to be thinner or skinnier and the handling difficulty increases. Instead of using three lengths of the sides, these two areas of an OBB are used for building models while the effect of the length, width and height of the part can still be captured. Meanwhile, l_{max} l_{mid} and l_{min} are also easily measured in the physical component.

2.3.2.3 Overlap Convex Hull Distance

The Overlap Convex Hull Distance (OCHD) is another input parameter for installation time model. It is used to estimate the insertion time by approximating the depth that a part needs to be inserted into another part. Instead of extracting the exact insertion distance from the tessellated model which requires a complicated implementation, using the OCHD results in more efficient calculation and minimum loss of information. An example of OCHD is shown in Figure 2.4. The OCHD is obtained from tessellated models by using MIConvexhull [26] library. The MIConvexhull is an open source C# plug that can generate a convex hull for a given 2D, 3D, and higher dimension model.

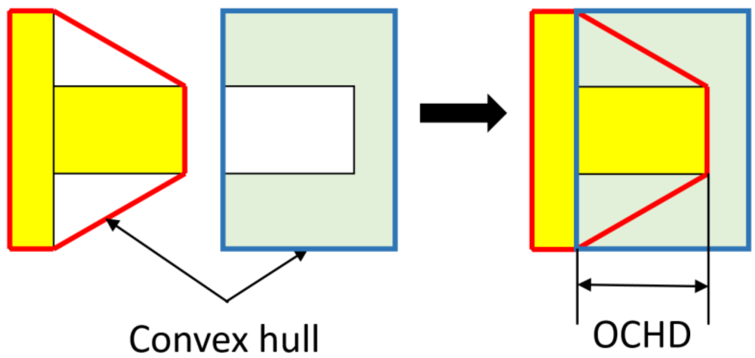
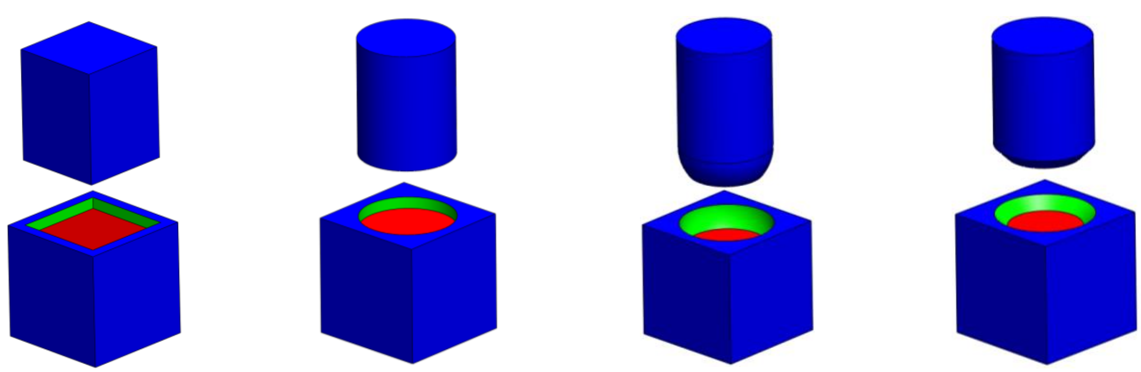


Figure 2.4: Convex hulls of two components and the Overlapping Convex Hull Distance (OCHD) after they are installed

The insertion distance for a part can be easily obtained in the experiment by measuring the travel distance between its pre-install and post-install position. This parameter is also used for estimating securing time by approximating the depth that a threaded-fastener needs to be inserted into a part before it got tighten. Usually, this parameter is represented by the thread length; however, the thread length does not accurately represent the actual insertion distance for fastening in some situations. Meanwhile, the actual insertion distance inside the part is very difficult to collect from experiments. To reduce time and cost, approximately half of the thread length is estimated as the insertion distance for situations where thread length does not represent insertion distance.

2.3.2.4 Number of Alignment features

In this study, alignment features are defined as the surfaces that facilitate the alignment action. During installation, the alignment difficulty between two parts is related to the connected surface geometries and it has significant effect on the install time. Hence, the number of alignment features is a suitable parameter for developing the install time prediction model and it is readily extracted from the CAD model. In Rafibakhsh et al.'s study [27], the surface geometries of a part is classified into four categories: flat, cone, sphere and cylinder. Also, the primitive surface classification is able to obtain these four surface features form a tessellated model automatically. Based on this implementation, alignment features can be identified from a tessellated assembly and they are a subset of the connected primitives which are between two or more connected parts. For a connected primitive surface, it is considered as an alignment feature if not all of the normal vectors on it are parallel to the insertion direction. For example, in Figure 2.5, all the connecting primitive surfaces after two parts are installed are highlighted with green and red colors. Since no normal vector on the green surface is parallel to the insertion direction, they can affect the insertion difficulty and are all considered as alignment features. Also, the number of alignment features of the cuboid connection in Figure 2.5 (a) is equal to four since the four perpendicular green flat faces can affect the alignment. For the cylinder, sphere, and cone, they are equal to one. Some of the alignment features, such as cone and sphere, can make part alignment easier than other feature types. In Boothroyd and Dewhurst’s study [10], a part that has this kind of feature is called a self-aligning part and can improve the installation time. Ideally, including all of these alignment information in the model can increase the accuracy, but it also increases the number of inputs. Therefore, the quantity of these features is used as model input to reduce model inputs dimension. These features can be also obtained in the experiment by observing the physical contact between parts.



(a) (b) (c) (d)

Figure 2.5：Four alignment features, flat (a), cylinder (b), sphere (c) and cone (d), are highlighted with green color

2.3.2.5 Bolt thread number, tool effect and nut

Thread number is an important input parameter for the securing model and it is proportional to the number of turns that the human or tool needed preform for fastening. With a larger thread count, it takes more turns to secure the fastener. Meanwhile, the algorithm for detecting the bolt thread from tessellated model has been implemented in our previous work [28], so the number of threads can be obtained automatically.

Tool effect is a binary variable that represents two fastening methods, powered screwdriver and manual tool (such as a screwdriver, wrench or torque wrench). This parameter is defined as 0 for manual tools and 1 for powered screwdriver. There is also an input to distinguish screws with nuts, coded as 1 and screws without nuts, coded as 0. This parameter is included because screws with nuts would require relatively more time and more tools than the alternative.

2.3.2.6 Rotation angle

The rotation angle is defined as the angle between the original and rotated footprint face normal vectors. In this study, it is assumed that all of the assembly tasks are produced on a flat surface. Hence, if an assembly is rotated before next assembly task, the original footprint face that this assembly shares with the table will rotate too. By measuring the angle between the two normal vectors on the original and rotated footprint faces, we can estimate the rotation difficulty.

2.3.2.7 Variables Extraction from the Assembly

After the model input variables are selected, they are assigned to four predictive action time models and measured in the physical experiment. The parameters for all four models are shown in Table 2.1. The oriented bounding box (OBB) features are included in all of the four models since they are directly related to the handling difficulty during these operations. This is also true for the component weight except for the secure action since the fasteners are relatively small and the effect of their weight is ignored in this study.

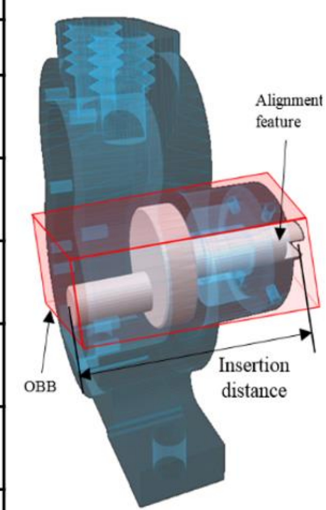
Table 2.1: Input features for four assembly actions.

Symbol	**Move**	**Install**	**Secure**	**Rotate**
x_1	Weight	Weight	Maximum OBB face area	Weight
x_2	OBB volume	OBB volume	Minimum OBB face area	OBB volume
x_3	Maximum OBB face area	Maximum OBB face area	Number of bolt threads	Maximum OBB face area
x_4	Minimum OBB face area	Minimum OBB face area	Insertion distance	Minimum OBB face area
x_5	Moving distance	Insertion distance	Nut (Covariance)	Rotate angle
x_6		Number of alignment features	Tool effect (Covariance)	

An example of the how these parameters are presented in a CAD assembly model that include a pump body and a shaft is shown in Table 2.2 with all of the parameters that are related to the shaft installation time. The OBB is formed by the shaft and the insertion distance is equal to the length of the OBB which is same as OCHD in this case. Meanwhile, the number of alignment features is equal to one since there is only one coaxial cylinder feature shared by the body and shaft. The shaft moving distance between its storage location and the pre-install position is based on the assembly station layout design and can be extracted. In this study, it is set to 500 mm. The weight information is easily obtained by using weight scale for the experiment. However, it is not directly provided for tessellated models since such models lack density information. Hence, we assume aluminum's density for all the unspecified parts and a more accurate weight can be obtained once the specific materials are defined. The description of the secure time model variables is omitted here since the measurements for these in tessellated model and experiment is straightforward.

Table 2.2: Input variables for assembly task 3

Input variables		Actions	Related part(s)
Weight (g)	17.58	move, install	Part 10
OBB volume (mm³)	103680	move, install	Part 10
Maximum OBB face area (mm²)	2880	move, install	Part 10
Minimum OBB face area (mm²)	1296	move, install	Part 10
Moving distance (mm)	500	Move	Part 10
Insertion distance (mm)	80.2	install	Part 10, 1, 4, 5
Number of alignment features	1	install	Part 10, 1, 4, 5



In this example, we can see that the selected model variables are automatically extracted from the tessellated model for installation time evaluation. They can also be measured manually in the experiments. After confirming model variables, the design of experiments method is applied for field data gathering.

2.3.3 Design of Experiments

In recent years, design of experiments has been applied to optimize and predict multiple variable systems in various fields [29,30] . By using design of experiments statistical approach, we can avoid studying all possible factor combinations, resulting in a minimum number of experiments required to analyze the relationships between the response and multiple-variables.

Response surface methodology (RSM) is a statistical analysis for model building and predicting. By carefully designing experiments, RSM can identify the relationship between response and predictors or input variables [31]. Therefore, the goal of RSM in this study is to find approximation models to predict the time for moving, installation, securing and rotation actions. The input variables for the four models have been defined in the previous section. Also, the Box-Behnken Design (BBD) for RSM has been used to specify experimental design to perform a response surface regression for each of the models.

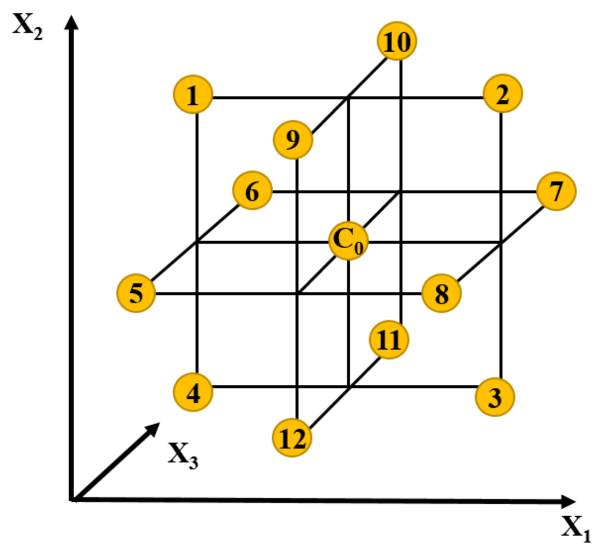


Figure 2.6: Three-levels BBD design schematic for three factors

To reduce the experiment cost and time, the BBD is used to optimize the response of the input parameters. An example schematic of BBD for three factors is shown in Figure 2.6. C_o is the central point. Multiple tests are conducted in this point to ensure the repeatability of the data. The other points (labelled 1 to 12) indicate the location of experiment runs for a model that has three input parameters. Compared to a three-level full factorial design which requires number of experiments is equal to 27, BBD requires significantly fewer experiment runs.

The controllable variables for each of the time models in Table 2.1 are chosen and assigned three range levels except for the two binary variables: the nut factor and the tool factor. Three coded range levels are defined to describe the different levels for each factor: low (-1), center (0) and high (1). The required combinations of all variables for all four experiments in Box-Behnken design are shown in Table 2.3. In this table, each ±1 indicates two runs: high level (+1) and low level (-1), while the other factors are kept at the designed level. BBD requires a minimum 54 experiments for install, 46 for move and rotate and 28 for secure. Also, replications on experimental runs are included to decrease the experimental error.

Table 2.3: Coded factor levels for Box-Behnken designs for four, five and six factors

Install							Move and rotate						Secure				
	x_1	x_2	x_3	x_4	x_5	x_6		x_1	x_2	x_3	x_4	x_5		x_1	x_2	x_3	x_4
Run 1 - 8	±1	±1	0	±1	0	0	Run 1 - 4	±1	±1	0	0	0	Run 1 - 4	±1	±1	0	0
Run 9- 16	0	±1	±1	0	±1	0	Run 5- 8	0	0	±1	±1	0	Run 5- 8	0	0	±1	±1
Run 17 - 24	0	0	±1	±1	0	±1	Run 9 - 12	0	±1	0	0	±1	Run 9 - 12	±1	0	0	±1
Run 25 - 32	±1	0	0	±1	±1	0	Run13 - 16	±1	0	±1	0	0	Run13 - 16	0	±1	±1	0
Run 33 - 40	0	±1	0	0	±1	±1	Run 17 - 20	0	0	0	±1	±1	Run 17 - 20	±1	0	±1	0
Run 41- 48	±1	0	±1	0	0	±1	Run 21- 24	0	±1	±1	0	0	Run 21- 24	0	±1	0	±1
Run 49- 54	0	0	0	0	0	0	Run 25- 28	±1	0	0	±1	0	Run 25- 28	0	0	0	0
							Run 29 - 32	0	0	±1	0	±1					
							Run 33 - 36	±1	0	0	0	±1					
							Run 37- 40	0	±1	0	±1	0					
							Run 41- 46	0	0	0	0	0					

2.3.4 Products Description

In this experiment, a chainsaw, lawnmower engine and airplane seat are selected to satisfy the Box-Behnken design. The assembly task is repeated by five participants, and the operation processes times were recorded by breaking the video into frames to minimize measurement error.

The chainsaw is mostly comprised of small and lightweight parts. Twenty-two of them are plastic components, and only the chain supporter and the motor are made of metal. This product also contains 16 threaded fasteners divided into 5 different types. There are two reasons for choosing this product. First, compared to other regular products, the parts of this assembly have more complex alignment features that make the installation more difficult. Second, the surface profile of each of the parts is complicated. Since OBB is selected as an input for evaluating the handling difficulty, the actual surface profile is neglected. So using this product can test if it is suitable to select OBB as input parameter versus the actual profile.

The second assembly is a lawnmower engine and it has 21 unique parts, most of them are made of metal. Seven fasteners from three different types are used to fasten this product. Half of the parts have small volumes so the overall handing difficulty is higher than the other regular components. Meanwhile this product also contains heavy parts with irregular shape such as the crankshaft, engine body. Thus, it provides significantly different data in comparison to the chainsaw.

The third assembly is an airplane seat that contains 32 parts and 48 bolts. It has the largest size of these three assemblies. Different from the other two products, assembling this product requires major motions. The worker needs to walk to the different places to acquire the parts. Using this assembly can obtain the data related to multiple assembly stations since the parts or sub-assemblies are far from the assembly station and require longer moving times.

2.3.5 Experiment Coordination

Eight graduate mechanical engineering students participated in the experiment in which at least two products were randomly assigned to each of them for assembling. At the end, each of the assembly tests was done and repeated by five different individuals. Before time recording, each of the participants was trained by teaching the standard operations for reducing the assembly time and variance. Later, they practiced three times each to be familiar with the assembly procedure. After training, we expected that the assembly speed would be uniform, and the data will be closer to real manufacturing scenarios.

Figure 2.7: Three assembly layouts for product assembly experiments: (a) lawnmower engine, (b) chain saw, (c) airplane seat

We simulated the workbench assembly station and modular assembly station for assembling the three products in this experiment and the assembly layouts are shown in Figure 2.7. Workbench assembly layout is used for the chain saw and the engine assembly. All of the components in these two stations are placed in the locations that are easily reachable for the participants so that they can finish the assembly task with minimum motions. Compared to these two assemblies, the scale of the airplane seat is significantly larger and it results in a different working spaces. Hence, a modular assembly station is used for studying the assembly time that involve major body motions. For the airplane seat, the nine subassemblies were placed around the operator to simulate a flexible assembly layout. To complete the assembly task, the participants need to preform turning, lifting, bending and walking motion to transport the subassemblies to the central assembly area. During the assembly procedure, only one component or fastener could be taken to the assembly area so the moving time for each component would not be interrupted. Also, if a securing operation existed for a given set of parts, it was required immediately after the final part of that set was installed. This can ensure the next operation will not be affected by any loose parts. All the components are placed in the designed locations where the moving distances are different for maintaining the Box-Behnken design. More detail about the application of Box-Behnken design in this assembly experiment can be found in our other research [32].

In the experiment, all of the components are labeled with numbers for indicating the assembly sequence. The participants only acquire and assembly one component at a time and follow the Moving, Installing, Securing, and Rotating action sequence so the times for these actions can be recorded separately. For time recording, first, we define the moving time starts once the participant's hand touches the part, and it ends when the part arrives at the assembly area (Figure 2.7) where it is ready to be installed. Then the installation time is recorded as the end of moving time until to the installation is finished. For securing, the time starts once the fastener is contacted and it ends when the securing tool is placed down. At last, the rotating time starts when the participant's hands touch the subassembly and ends until the rotation action finished.

To minimize the error during time recording, cameras were used to record every procedure. After that, each of the videos were broken down into frames and the time was read from them. The accuracy of the recorded time is ± 0.033 seconds due to frame rate of the camera is 30 frames per second. By applying this method, all the four action times are obtained for constructing prediction model.

Finally, each product was assembled by five participants and only the best times for each participant are recorded. That means, for each type of actions, five data points were collected from five participants. The reason for using the best time instead of the mean time in this study is to better meet the times achieved by professional workers in production lines. By using the best assembly time, we can predict results that are more applicable to real production. All of the collected data is analyzed in the next section and the time prediction models development with these experimental data will be discussed in the later chapter.

2.3.6 Time Experiment Results

2.3.6.1 Correlations between Selected Input Variables and Experimental Data

After experiential data is collected, it is important to check the correlation between each of the selected model inputs and the experimental time for determining if they are strongly correlated. For each of the actions, the average action time for each of tasks is plotted in Table 2.4 with its corresponding input parameter.

Trend lines are used to visualize the general trend between the time and input with the correlation score which is calculated by the equation shown below:

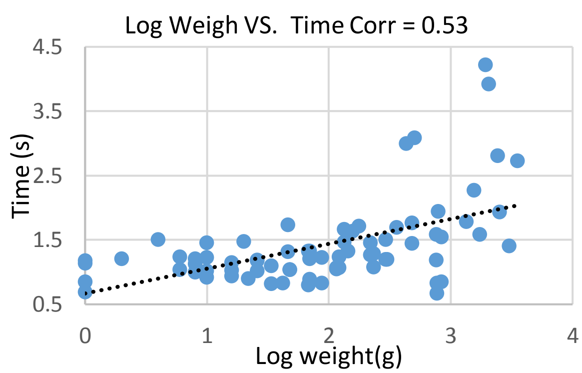
$$Cor(X,Y) = \frac{\sum(x-\overline{x})(y-\overline{y})}{\sqrt{\sum(x-\overline{x})^2 \sum(y-\overline{y})^2}} \tag{3}$$

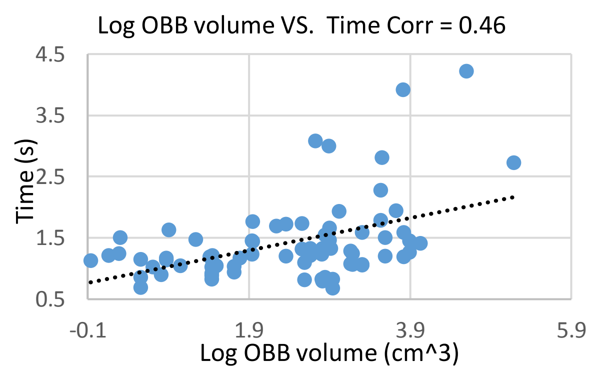
In this equation, X is the measured dataset of the model variable and Y is the corresponding action time. The correlation score is from 1, which stands for perfect positive correlation to -1, which stands for perfect negative correlation. Although there is no well-defined rule to interpret this score, in general, correlations are above 0.4 to be relatively strong; correlations between 0.2 and 0.4 are moderate, and those below 0.2 are considered weak. Based on this interpretation, the variables that we selected for the moving model have reasonable correlations with the experimental moving time while most of which are above 0.4. The correlations are slightly weaker in for the install data which are between 0.4 and 0.2 in general.

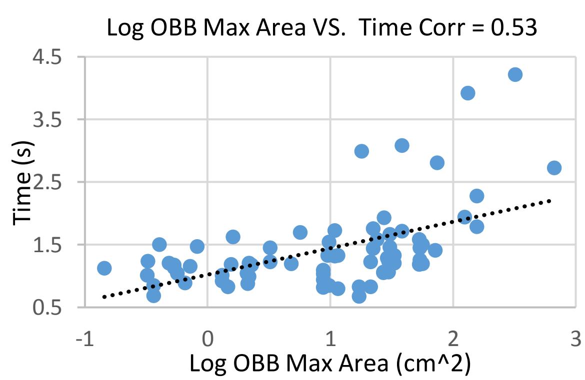
For secure action, the data of number of threads is divided into two groups based on whether the hand tool or powered tool is used in the securing action. The reason for this separation is because the powered tool can increase the fastening speed and reduce the securing time significantly compared to manual tool. So dividing this data can result in better understanding of the effect of the thread number. The correlation between time and thread amount is 0.19 for using powered and 0.41 for using manual tool. This is reasonable that when the thread number increases, assembler need to perform more fastening action with manual tool. Since this fastening action is not continuous (hands need to be twisted back for each tightening action) compared to the powered tool, the increase number of thread will have more effect on the secure time. Meanwhile, two OBB areas variable are strongly correlated to the securing time by having the scores of 0.55 and 0.49. Finally, the selected variables have very strong correlations with the rotate time which are between 0.5 and 0.8 except no correlation is found when it comes to rotation angle. One reason for this to happen is that the assemblies we used in the experiment are relatively small and are easy to rotate. Hence, the rotation angles did not make large impact on the rotation time. However, we believe stronger correlations can be found when the rotation actions are performed on larger and heavier products that require low rotational speed and major body motion.

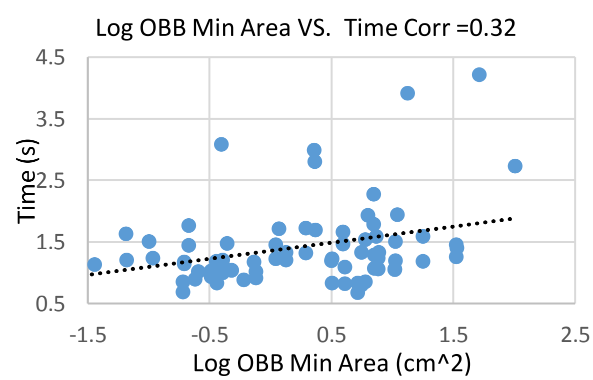
Table 2.4: Comparison Between Collected Action Time and Model Input Parameters

Move Data









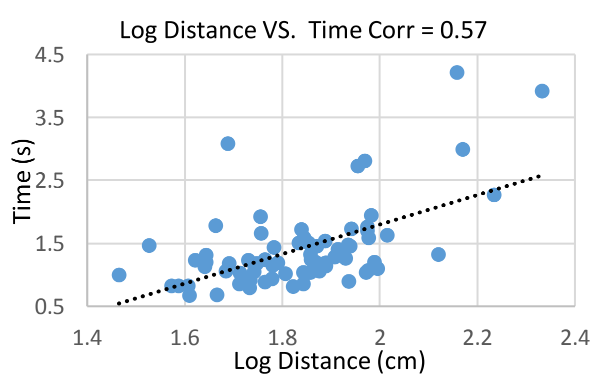


Table 2.4: Comparison Between Collected Action Time and Model Input Parameters (Continued)

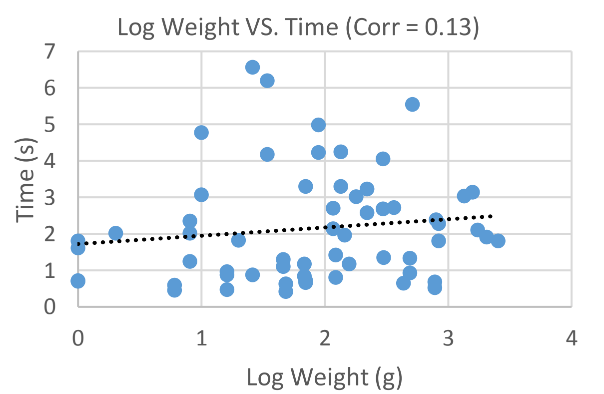
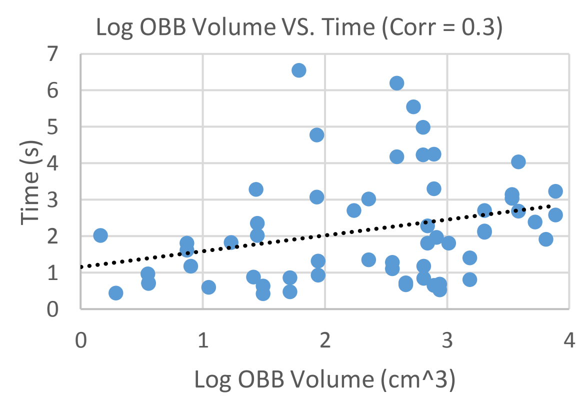
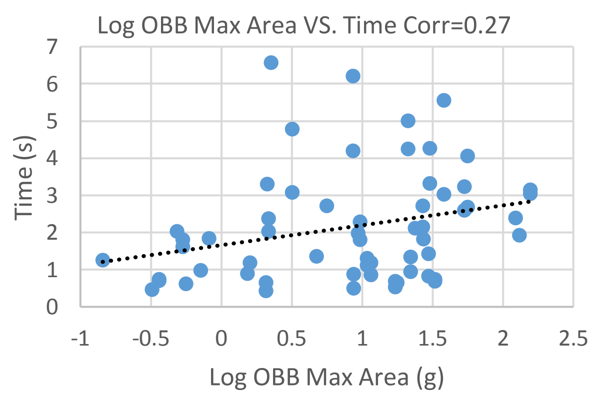
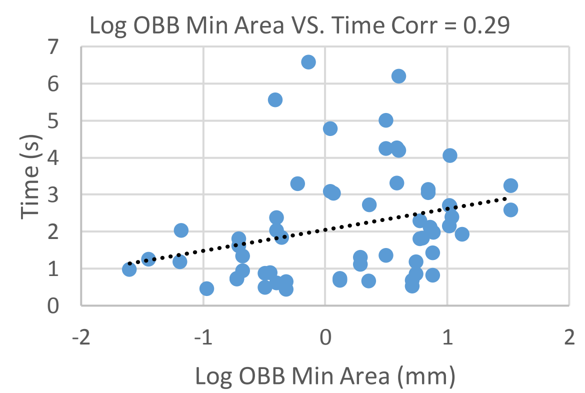
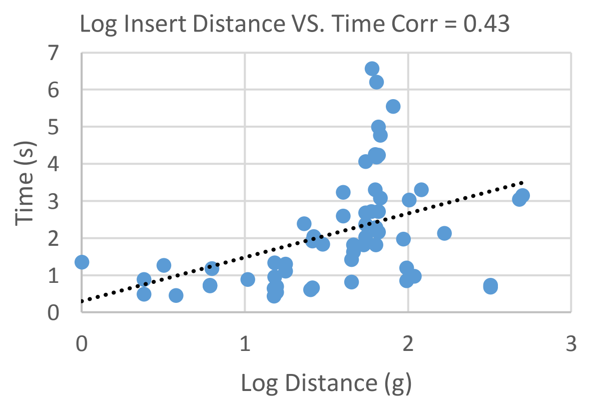
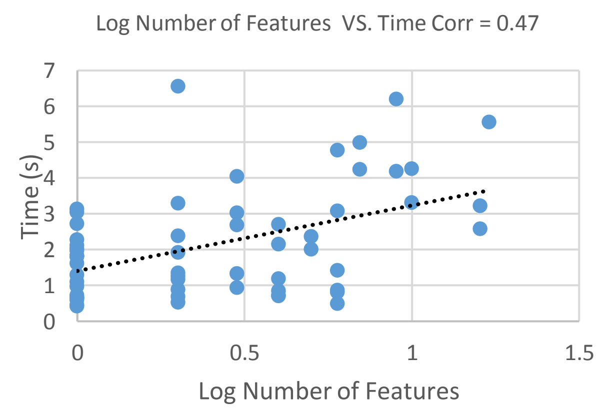
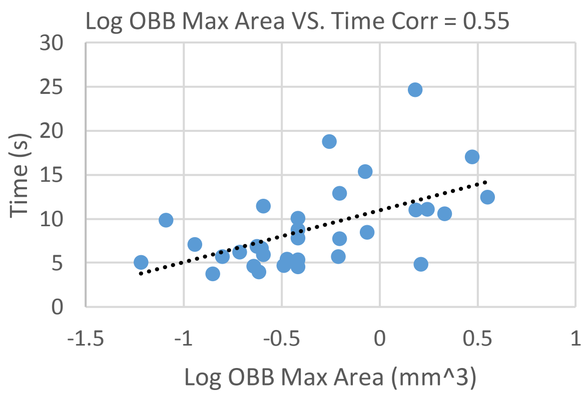
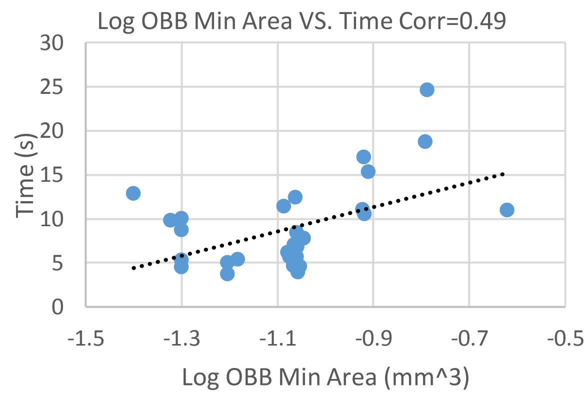
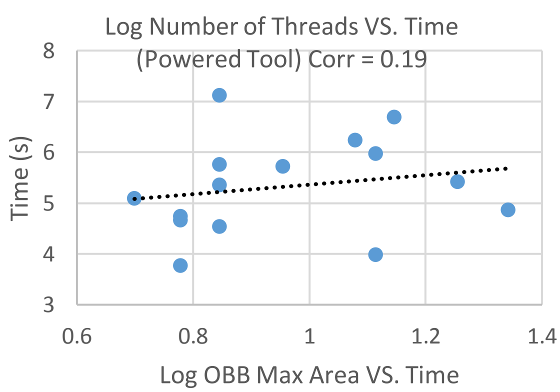
Install Data	
	
	
	

Table 2.4: Comparison Between Collected Action Time and Model Input Parameters (Continued)

Secure Data







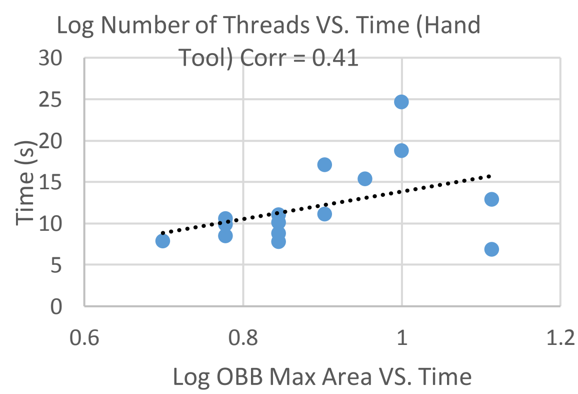
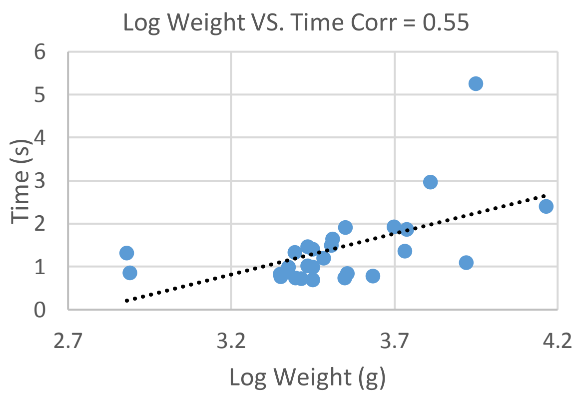
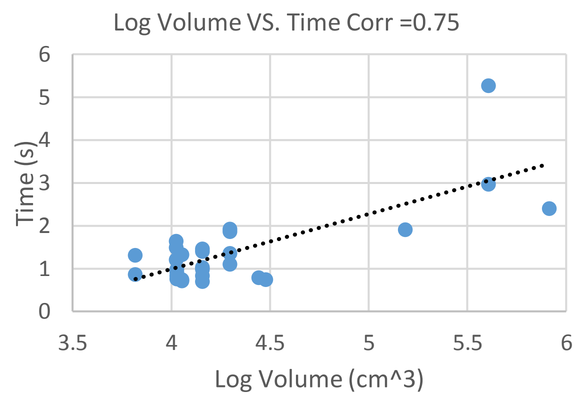
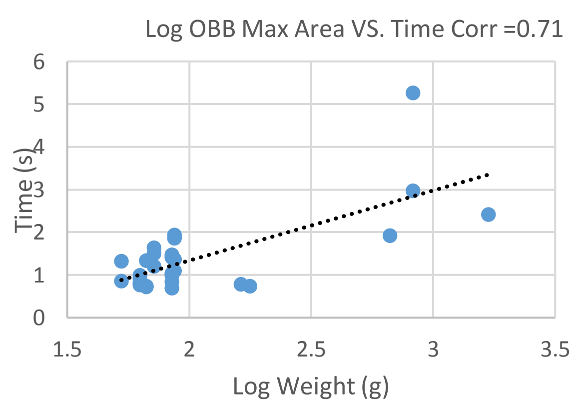
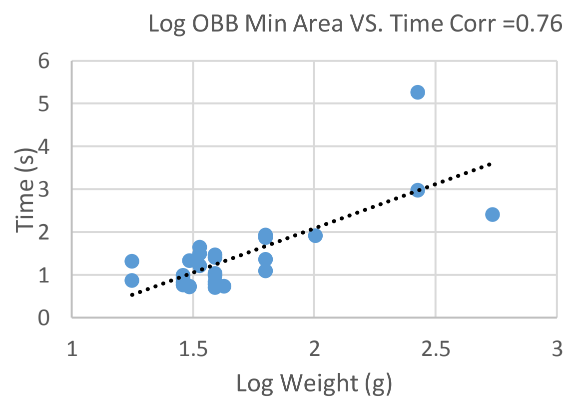
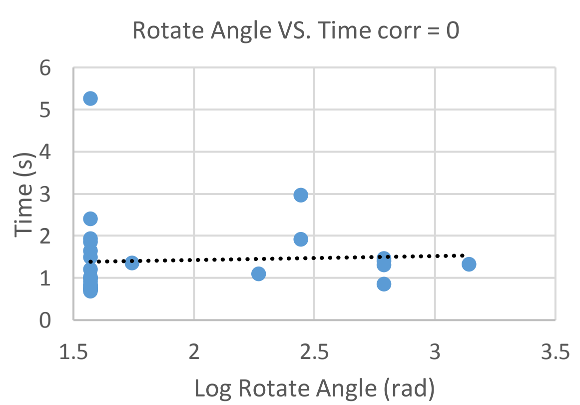


Table 2.4: Comparison Between Collected Action Time and Model Input Parameters (Continued)

<table>
<tr><th colspan="2">Rotate</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td></td></tr>
</table>

2.3.6.2 Discussion of Final Data

After analyzing the correlations between feature variables and the collected average action time, all of the collected time for the move, install, secure and rotate operation are represented by the boxplots in Figure 2.8 to show the distribution of all of the participants' performance in this study in each of the tasks. Based on these plots, some of the general findings are addressed here. First, the person to person variations increased with the action difficulty. For more complex assembly tasks, they require more actions time and the performances of different participants becomes more inconsistent. For example, from tasks 66 to 74 in the moving time plot and from tasks 25 to 28 in the rotate time plot are about moving and rotation of the large airplane frame which the participants need to conduct major body motions to finish the assembly tasks. The average standard deviations of these two ranges of action times is 1.3s and 2.1s, and they are much higher than the average standard deviations, 0.3s and 0.2s, from the rest of the data that the assemblies are in smaller scales. The same conclusions can be applied to the install and secure data. Second, compared to rest of the action times, the secure data has the highest average action time which is 8.94s and the highest standard deviations which is 2.45s. This is due to the high complexity of the securing action. Unlike the other three actions, the secure process includes fasteners and tool acquisition, fastener alignment and manual tightening. Variance is introduced to each of these actions when it is performed. Third, person to person variances are large in some of the tasks which can be reduced by having more well-trained participants. Although the participants practiced the assembly tasks before recording, the assembly times between people are still inconsistent. Such limitations can affect the accuracy of applying the prediction model to estimate industrial production time. However, the prediction model that we used in the research has ability to be updated based on new user input. Hence, the collected data is still valuable for developing the initial prediction model.

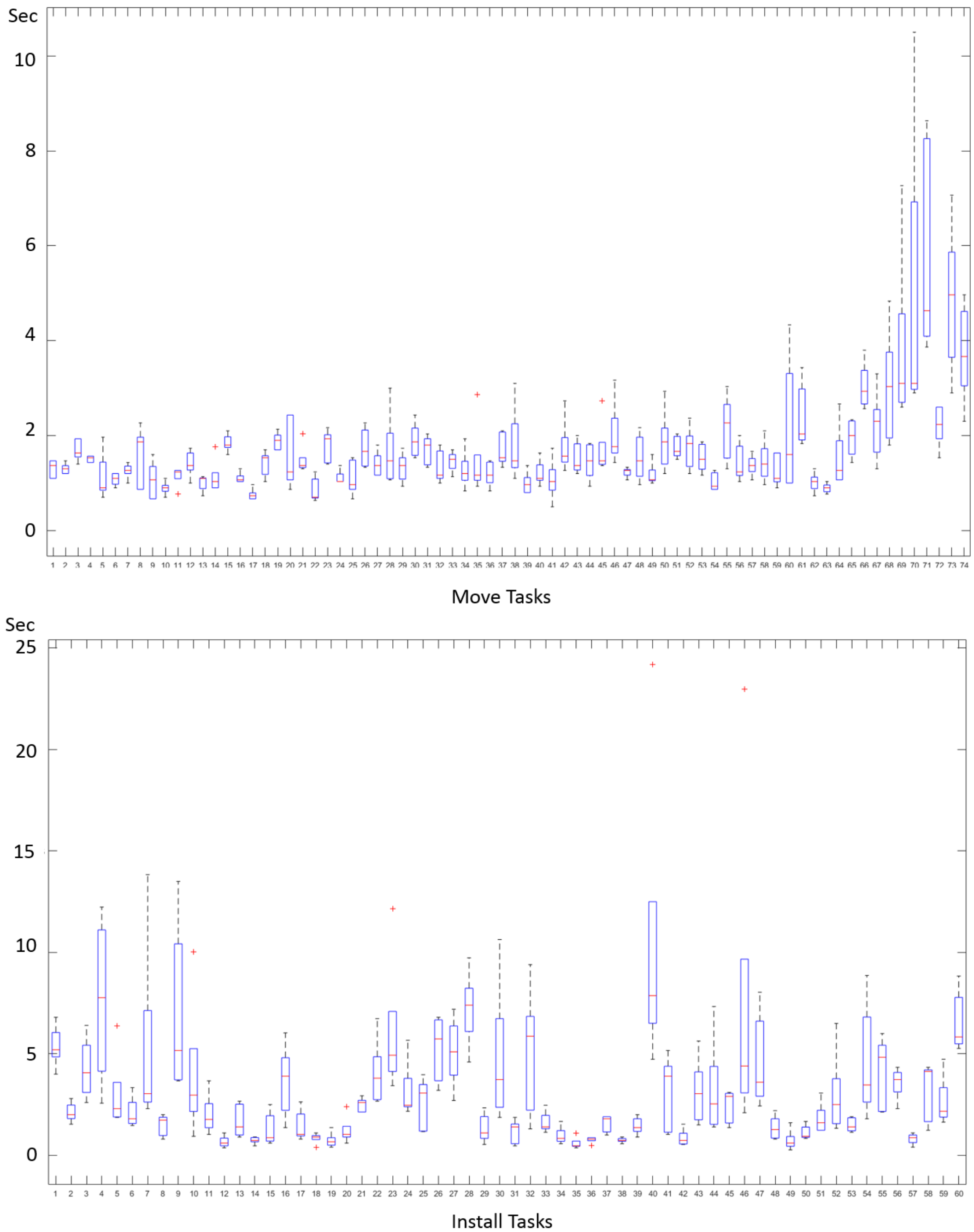
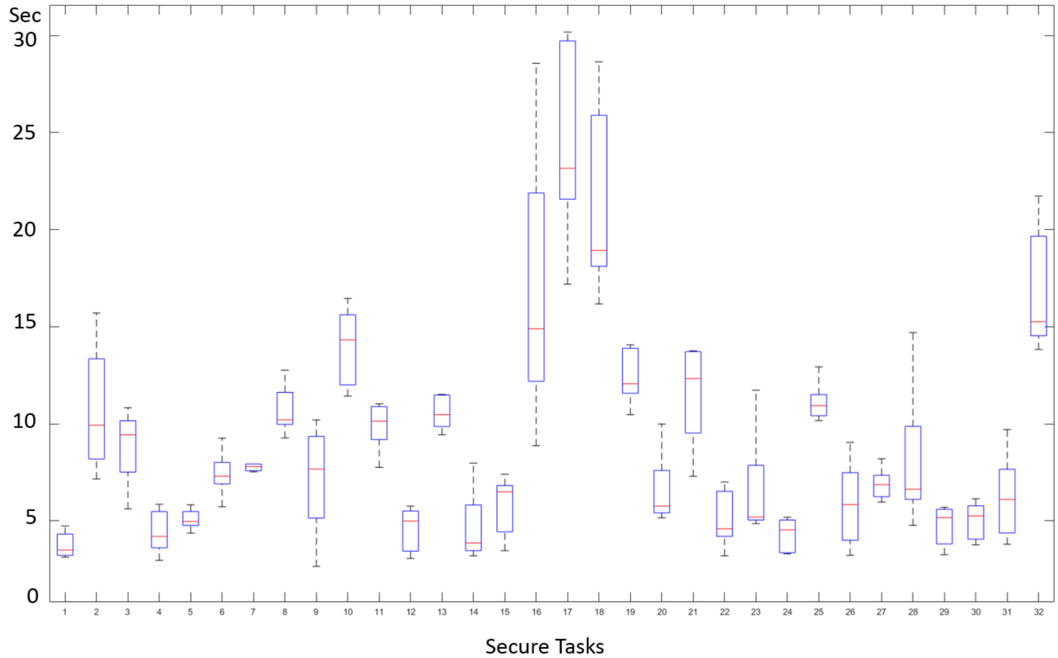


Figure 2.8: Four action time with all experiment tasks



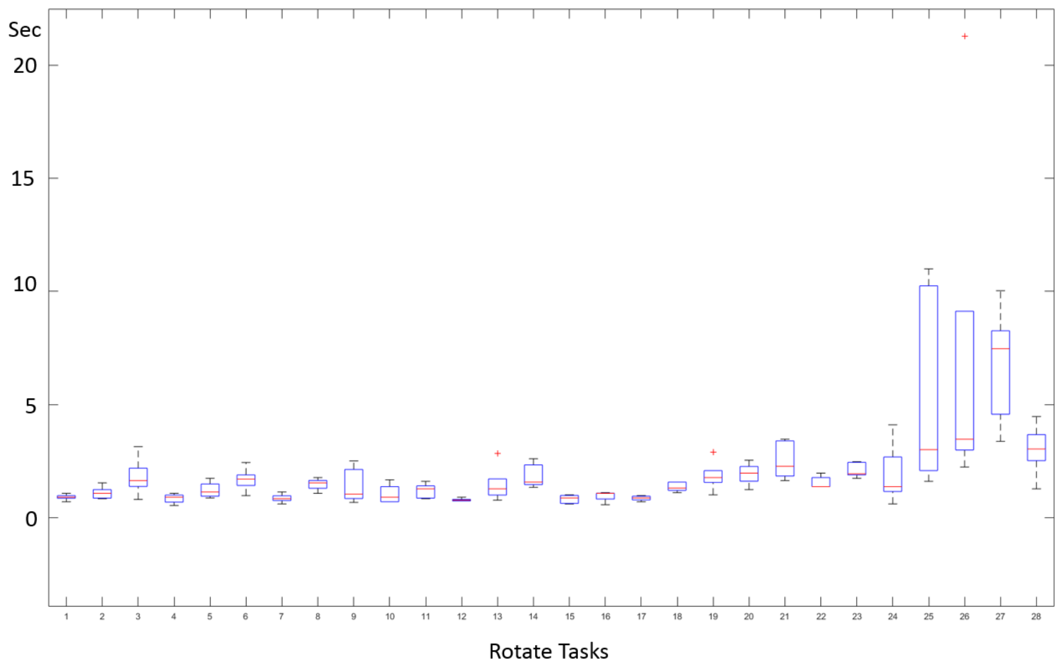


Figure 2.8: Four action time with all experiment tasks (Continued)

2.4 SURVEY FOR ASSEMBLY STABILITY

Assembly stability is other important factor that needs to be considered during assembly planning. With higher stability, the assembly task requires less usage of fixtures and the safety will also increase when components are harder to fall off. However, compared to the well-studied assembly time evaluation method, the methods for evaluating the stability during assembly process are limited. Also it is a challenge for applying automated stability evaluation during a lengthy search process with CAD assemblies. Hence, a survey of human perception of stability is used to provide a better understanding of how people evaluate different assembly so that an accurate stability estimation model for the assembly planning can be developed.

The survey examines two kinds of stability are evaluated: internal stability and external stability. Unlike the previous assembly time experiment which is used to obtain data for prediction model development, the survey results are compared with the computational results for validation. In this survey, stability is estimated based on 3D printed assembly and its 2D CAD image. The collected data will be discussed at the end of this section.

2.4.1 Internal and External Stability

Internal and external stabilities are the two scores that participants need to evaluate for every given product. Internal stability is used to describe the stability of the connections between components in assembly. By using internal stability, we can know if the part to part connections are stable with the assigned orientation and it is very helpful to evaluate the assembly plan when a lot of rotation actions are involved. Different from internal stability, external stability is used to evaluate if the whole assembly is easy to tip over when it is placed on the work section.

The sample description of these two kinds of stability is shown in Figure 2.9. For internal stability, the connection is unstable when P1 and part P2 are connected with a sloped surface and there no fixture to limit P2 from sliding along the surface. In comparison, the prismatic connection between P1and P2 is more stable since the sliding motion is less likely to happen. When evaluating the external stability, it is assumed that the components are fixed with each other and the whole assembly is seen as an individual part. In this case, the assembly of part P1 and part P2 is in two different orientations with the prismatic joint connection, the assembly in second orientation has better external stability since its projected center of mass (COM) is within its wide footprint face when it is placed on the assembly station while compared with it is in the first orientation which's projected COM is outside of the footprint. By using the second orientation in the assembly process, less fixtures will be needed to support the assembly for the subsequent tasks.

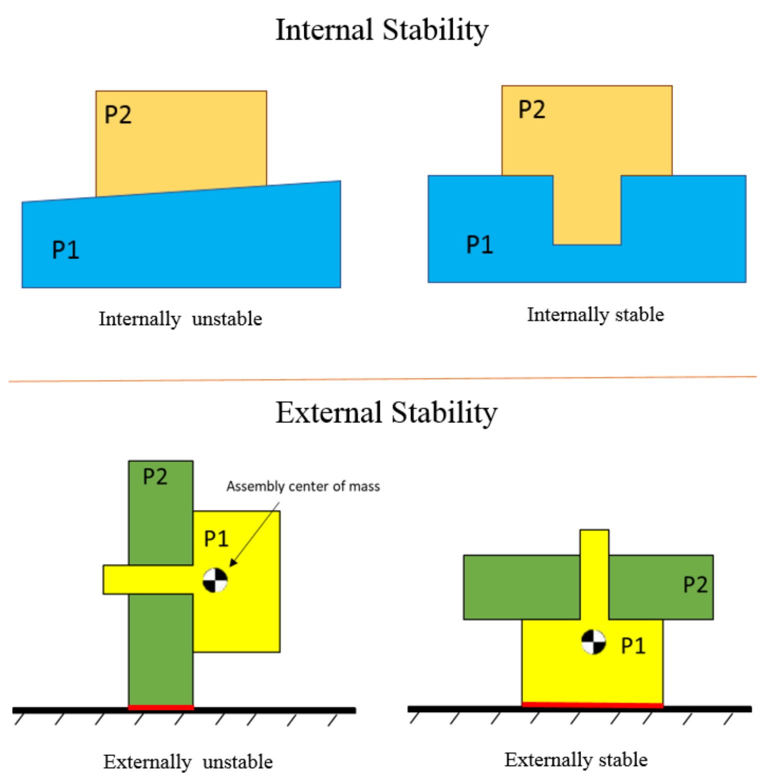


Figure 2.9: An assembly is showing different internal and external stability with two orientations

2.4.2 Product Description

In this survey, three 3D printed assemblies, clamp, pump and piston compressor are used and shown in Figure 2.10 with all the subassemblies that are used in this survey. The reason for choosing these three products is due to their structural differences which increases the diversity of cases so different ranges of results can be observed. Meanwhile, all the kinematic constraints within these assemblies are represented by the constraints shown later in Section 3.2. Hence, the proposed computational methods are applicable to these assemblies as well.

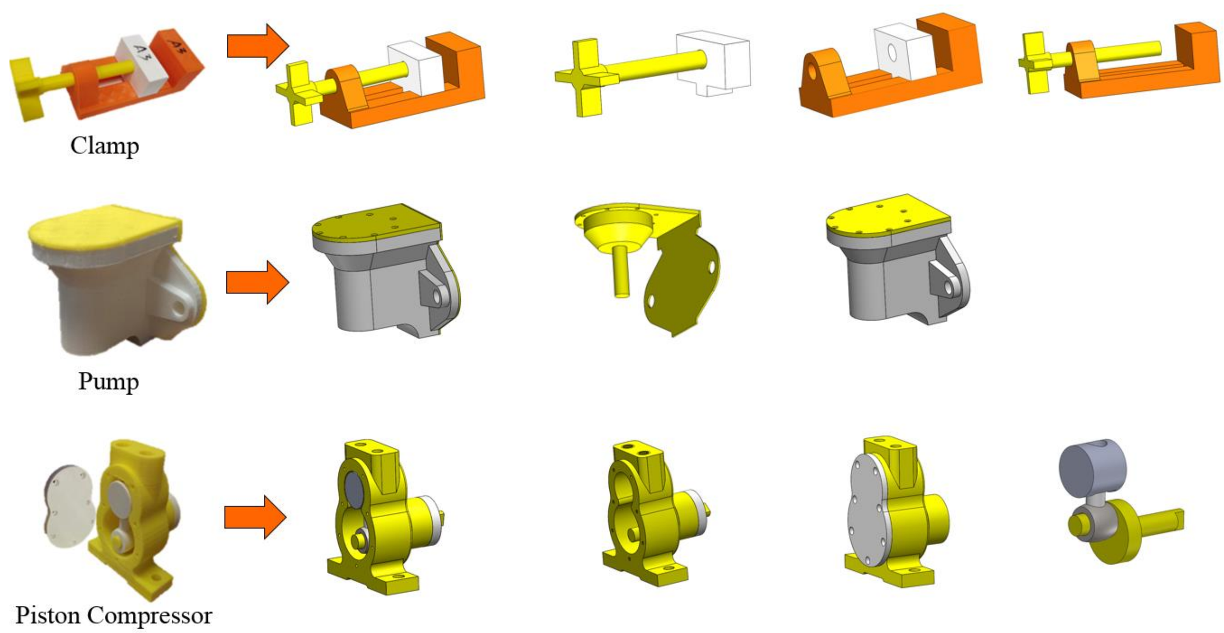


Figure 2.10: 3D printed assemblies that with their subassemblies are used in the survey

Of all of the products, the clamp is the simplest assembly and only contains three components. These components can only perform linear movement except the push rod which can also be rotated. The pump assembly has four components. The top and back lids are merely flat plates that have high a DOF when not fastened since they rest on flat surfaces on the exterior. Meanwhile the shaft inside the pump is rotatable and is able to be removed straight out the top when the lid is removed. Lastly, the piston compressor is the most complicated assembly which has a structure similar to a piston engine. The shaft and piston perform rotational and translation movement while installed in the pump body. The crank that connects to these two parts produce rotation movement that relates to the shaft and linear movement that relates to the piston. A lid that has a high DOF covers all these components and is located on the side face of the body. Since this study is focus on the stability evaluation before action is taken to secured parts together, fasteners are excluded in these assemblies.

The reason for using 3D printed assemblies instead of real products is the CAD files of real products cannot be accessed easily. In this study, it is important to have the CAD files of the products that are used in this experiment, so we can import these CAD files to the computational model and generate results are compared with the experiment data. Although the survey data will be more realistic when using real products, extra time will be need for recreating the CAD file. Hence, 3D printed assemblies are preferred in this study.

2.4.3 Design of the Survey

In the survey, each of the assemblies that we mentioned above will be decomposed into three to four subassemblies with different combinations of connected components (Figure 2.10). Meanwhile, three or four different orientations are assigned to the individual subassembly. The survey for rating the stability of the subassembly under the assigned orientation is designed in a seven points Likert scale [33]: extremely unstable(EU), unstable(U), weakly unstable(WU), neutral(NE), weakly stable(WS), stable(S) and extremely stable(ES). The full survey questions are included in Appendix A, and an example of the survey question is shown in Figure 2.11. First, the subassembly for the survey question is given with its exploded view. Based on the image, the participants can have a better understanding of the inner structure of the subassembly and can assemble the product from the 3D printed components. Next, the four orientations are represented by the main view and the side view of the subassembly and the participants need to place the product in the same orientation shown in the survey question before evaluating the stability. In total, 44 evaluation questions are designed in this study. To minimize the dependent effect (e.g. assigning similar score to two consecutive subassemblies that are derived from the same assembly), three surveys are defined with different randomized question orders.

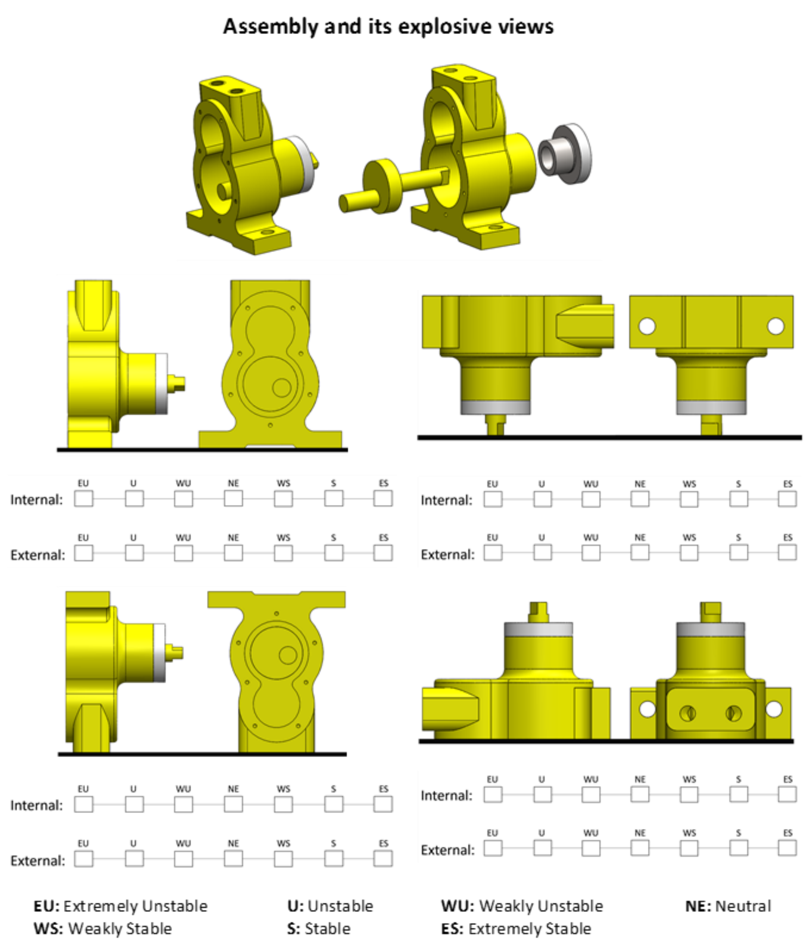


Figure 2.11: An example survey question that participants need to build the 3D printed assembly based on the product that is shown in (a) and evaluate its stability under four orientations (b)

A total of 21 subjects, undergraduate mechanical engineering students at Oregon State University, participated in this study. Three sets of surveys are randomly assigned to six groups and each of these group contains three or four people. Before answering the questions, each of the participants was asked to handle and manipulate the three 3D printed assemblies. After that, each of the participants assembled the subassembly as is shown in the provided questions for evaluating its stability with different orientations.

2.4.4 Survey Results

In this section, two analyses were conducted. First, we will study how the individuals evaluate all of the survey question and the variations between questions to question. Second, a t-test is used to see if there are any significant difference between the image-based and product-based evaluations.

2.4.4.1 Survey Data Variation

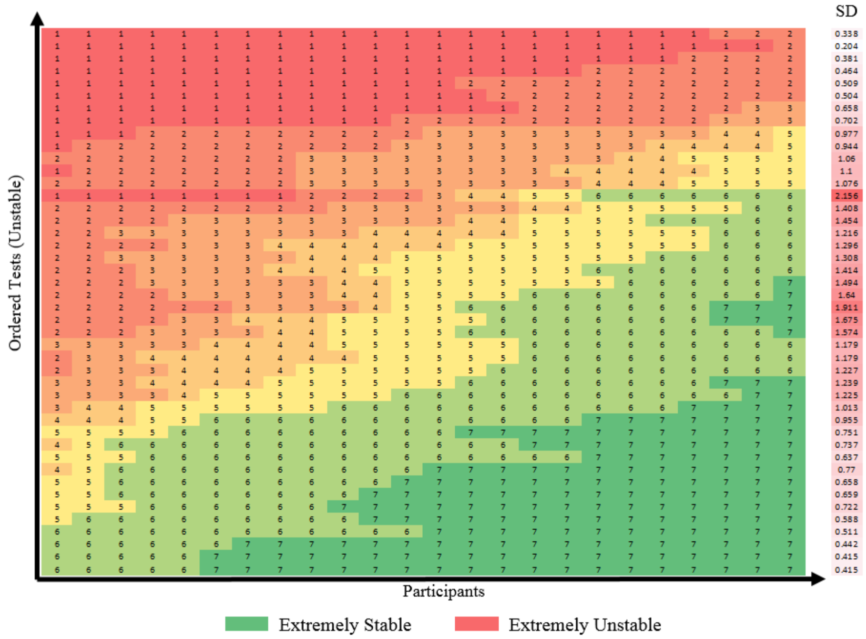
For each of the survey questions, the top two stable and unstable scores are removed for preventing outliers. After that, we have 24 responses for each of the questions (internal and external stability estimations) in the image based evaluation survey and 17 for responses in the 3D printed assembly based evaluation. Meanwhile, data that is collected in Likert scale is converted into values from one to seven.

All of the individual responses are represented by the ordered colored scale plot which is shown in Figure 2.12. Each row in the plot represents all of the responses for one survey question with their standard deviation. All of the rows are ordered by the mean stability score. In general, these four plots share a very similar pattern in that the top and bottom sections of the plot where the evaluated stabilities are more extreme, people have more consistent opinions. When the evaluated stabilities become more neutral, the mid-range of the plots indicate more diverse estimations. This phenomenon can be also described by the standard deviations which are shown beside the plots. Generally, the standard deviations (SD) are smaller on both sides of the plots and they start to increase when it is closer to the mid-range of the plots. Meanwhile, the SDs indicate the participants are more confident when they are evaluating external stability. The average SDs of the external stability estimation for both image-based and product-based survey are 0.97 and 1.22 and they are lower than the SDs of the internal stability estimation which are 1.44 and 1.65 accordingly.

These results indicate that the participants are able to portray consistent judgments when the assemblies are evaluated in more extreme situations. On the other hand, for assemblies that have neutral stability, the person to person variations increase. At the same time, the internal stability is harder for participant to evaluate since the SDs for those are averagely higher than the external stability evaluation.

External Stability

Image-Based Evaluation



Product-Base Evaluation

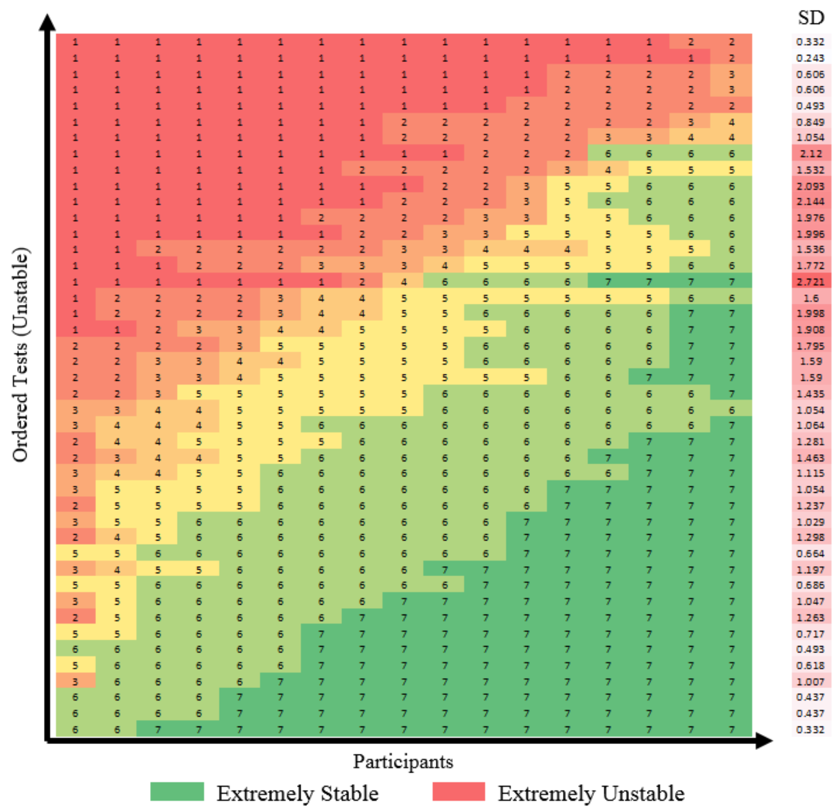


Figure 2.12: Color plot shows each individual evaluation for the forty-four tests in the survey

Internal Stability

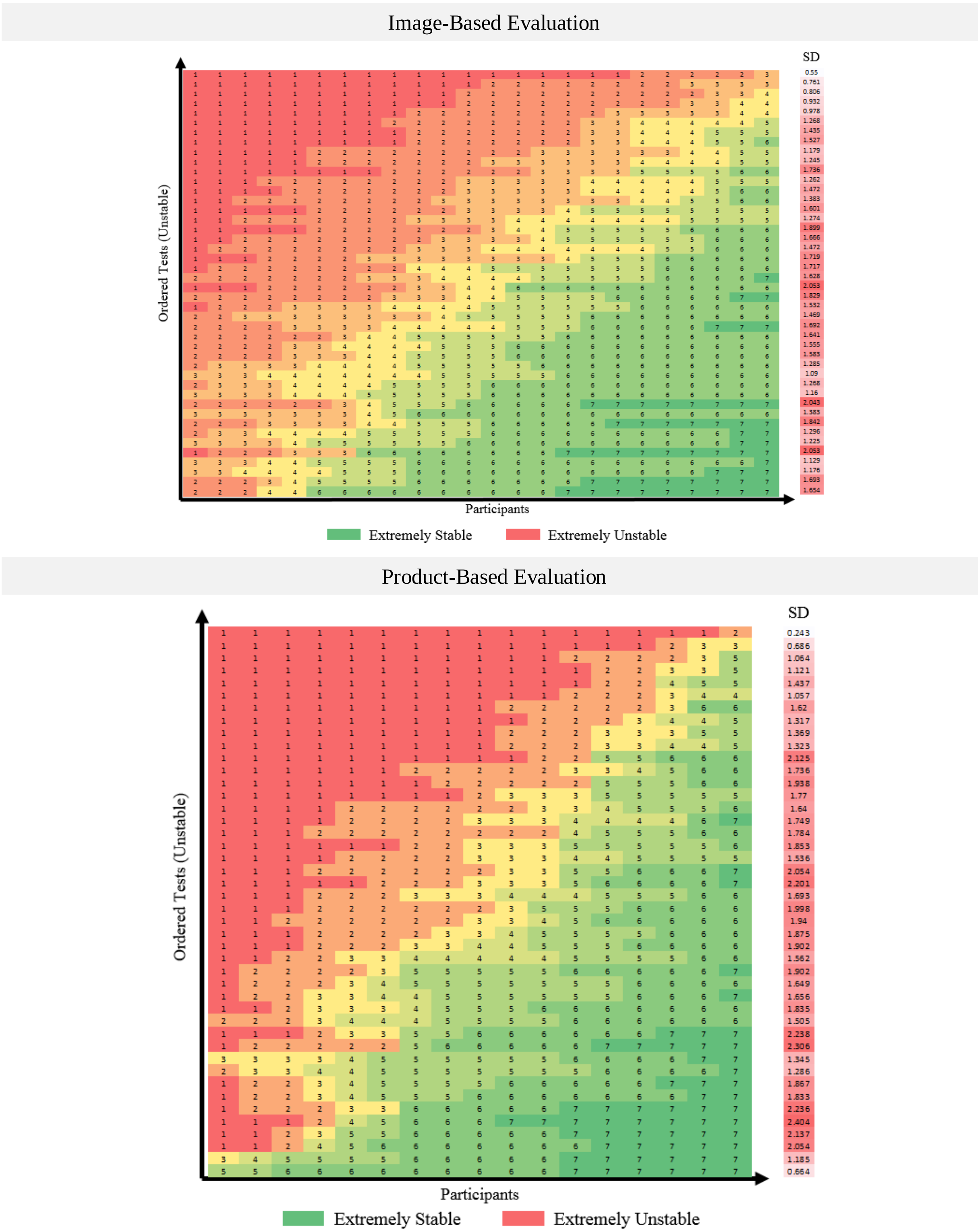


Figure 2.12: Color plot shows each individual evaluation for the forty-four tests in the survey (Continued)

2.4.4.2 Difference between Image-Based and Product-Based Evaluations

The reason for conducting Image-Based and Product-Based Evaluations is to study if there is any difference between using and not using physical assembly to evaluate the stability. The paired T-test method is conducted with the mean score of each of the estimations. The p-value for internal stability evaluation is 0 and it is 0.002 for external stability evaluation. These low p-values indicate the significant difference of the results where the 3D printed products are used. Hence, the product-based evaluation results will be used in the later section for more realistic comparison with the computational prediction.

2.5 CONCLUSION

Empirical data gathering is an important process to collect field data for assembly cost model development. Most of the assembly cost estimation tools such as DFA and MTM are all developed based on empirical data of human assembly studies and the collected data is used to implement the cost models that associates with model input parameters. Nevertheless, the traditional cost estimation methods are heavily depended on human interaction and not adaptive to an automated assembly planning environment. Hence, we developed a more automated method that can achieve automated planning early in the design process with only 3D CAD models. Experiments for obtaining usable data for this development are conducted.

In this section, we design two experiments for gathering assembly time data and stability evaluation. For assembly time experiment, we divide the assembly operation into move, install, secure and rotate action. So, the overall assembly time can be calculated by summing all of predictive times. The benefit of this is it can have a more accurate model to describe the various flexible production process. Parameters that can represent the features of the tessellated CAD models are selected as model input and measured during the experiment. Design of experiments (DOE) method is used to guide the experiment process and ensure the quality of the collect data. After collecting the assembly time data, most of the selected input features shows relatively high correlations with the corresponded action time.

Unlike assembly time experiments, the results from stability survey are used to validate the computational stability estimation methods. In this stability experiment, we focused on evaluating the internal stability and external stability by using images of the exploded view of the assembly and the 3D printed products. The internal stability is used to describe the stable level of the part connectivity and the external stability is for evaluating how stable when the whole assembly is placed on the working station. This experiment indicates that the judgments are inconsistent when people evaluating an assembly that has neutral stability.

CHAPTER 3 : ASSEMBLY TIME

3.1 INTRODUCTION

Assembly time estimation is an important factor in evaluating the performance of the assembly process. With the experimental assembly time data shown above in Chapter 2, machine learning methods can be applied to estimate assembly time based on tessellated CAD models.

Three mathematical time prediction models, *polynomial regression* (PR), *Gaussian Process* (GP) and *Artificial Neural Network* (ANN), are used for predicting assembly time in this research. The regression models are estimated by forward and backward selection to determine the significant terms in the model to predict new incoming data. Also, they are used as *basis functions* to represent the trends of the time estimates. Next, the GP is used to predict means and variances. With a selected kernel function, the GP model constructs a covariance matrix that contains correlations between all the input features. Model parameters are tuned to fit the model, which is done by minimizing the negative log marginal likelihood function through the use of standard optimization procedures. Additionally, the predictive confident interval of the GP can indicate the density of training data near input data. For example, a narrower CI shows that we have sufficient training data to support the prediction, and this information can be used in selecting the preferred assembly process design. Lastly, an artificial neural network with one hidden layer is used for predicting the assembly time and is compared with the regression and GP models. For all the models, 90% of the experiment data is randomly selected as training data for building the prediction models, and the remaining 10% is used as test data to gauge the accuracy in the model predictions. Also, a case study of a pump assembly is used to test the accuracy of the three models, and the results are compared with the design for assembly time prediction method[34]. Finally, an online Gaussian Process updating method is introduced so the model can be improved over time based on user feedback in future usage.

3.2 LITERATURE REVIEW

In this study, the time estimation method is developed based on the Design for Assembly (DFA) method, which is a product based method used for evaluating an assembly process. Three existing DFA methods, the Hitachi Assembly Evaluation Method (AEM), the Lucas DFA Method, and the Boothroyd–Dewhurst DFA Method, are the current state-of-the-art for evaluating a design for an assembly process. The Assembly Evaluation Method (AEM) was the first evaluation method for DFA and was developed by Hitachi [9]. For AEM, the assemble-ability evaluation score is used to evaluate the difficulty of assembly, using a simple downward insertion as the reference assembly. This score is panelized by complicated operation. Also, the cost ratio between the new design and the initial design is used to minimize the cost. In the Lucas DFA method, the measurement of the overall assembly difficulty is based on point scales which are assigned by the functional, feeding, and fitting analyses [35]. First, the functional analysis is used to determine the number of components that are essential to the product’s function. Second, indexes are assigned to each of the parts within the assembly to indicate the difficulty of the feeding and fitting based on their geometries, such as part size, weight orientation and so on. The higher handling and fitting difficulty a part has, the higher index value is assigned. Finally, the total handling index and the total fitting index is divided by the number of the essential components to compute the handling and fitting metric for evaluating the assembly design. The design of the assembly can be improved based on these analyses. Another DFA method, developed by Boothroyd, Dewhurst and Knight [10], is a product-based method that uses the physical features of the assembly parts to predict assembly time. In their study, assembly time is affected by the difficulties of handing and insertion. These difficulties are quantified based on the type of mate, geometry properties, weight of the component, etc. With this information, the assembly time is predicted by using basis functions. The DFA method also provides design guidelines for optimizing the product design, to reduce the assembly time and cost. Boothroyd and Dewhurst also developed DFMA software which is based on the method. The DFMA software can estimate the time and cost for assembling a product with the user's input information [36].

Recently, DFA time estimation is more automated, and with fewer or no human inputs, while the product design parameters can be extracted from the CAD software. In Mathieson' study, the number of manual inputs is reduced by using the connective complexity metrics method [12]. For this method, connections between parts are represented by bi-partite graphs and obtained from Solidworks. With this system, the complexity metrics are developed and used for constructing regression models to predict assembly time. Based on this work, Owensby and Summers [13] develop an automated time prediction tool for early design stages. In their study, the connections between parts are defined as mates within the Solidworks model, and this information is extracted automatically and used in complexity metric prediction modelling. This method increases the efficiency of estimating assembly time, but the authors admit that the mates within the same assemblies could be assigned differently, i.e., for the same connected parts, the inputs for evaluating their connection complexity could be different. In Ou and Xu's study [14], CREO Parametric is used to obtain the assembly and mechanism constraints to develop the assembly relations matrix, and assembly information matrix to generate that assembly sequence. Compared to other CAD packages, CREO Parametric captures more physical constraints within the model. In Rafibakhsh and Campbell's study [27], four common surface features, flat, cone, sphere and cylinder, are classified from triangulated 3D solids. With this information, the mating information can be obtained automatically from the connected parts without using commercial software and the design cost can be reduced. Meanwhile, the connection between parts are described more in-depth by using these primitives and such information are applied to the assembly time estimation for AAP.

Machine learning (ML) techniques have also been applied for estimating assembly cost. With ML, prediction models with higher accuracy can be achieved and the assembly cost can be predicted during early design iterations. Also, less detailed features are required for building accurate models [37]. In Chang's study [38], an Artificial Neural Network (ANN) method is used for estimating the assembly handing time, providing accurate results compared to the DFA method. ANN is also used in Miller's study [37] and the connectivity graph is used as an input for predicting the assembly time of vehicle sub-assemblies. Different from ANN which is independent from the basis function, the Gaussian Process (GP) method can integrate with an approximated predictive function and utilize a more sophisticated error assumption for more accurate results and quantification of uncertainty. Since the DFA time predictions are based on the regression models and much research [39–42] uses this form of model for evaluating the assembly sequence, GP is applied to these existing models for new predictive results with a confidence interval of the prediction. There are several advantages of using the GP method. First, GP can be used with insufficient data [43]. Also, the GP method is easier for implementation compared to the neural network method [44] in which the parameterization could be very difficult and require maximum a posteriori approximations [45]. Additionally, the predictive variance of the GP can indicate the uncertainty of the predictive result to support a robust sequence design. Knowing only the predictive time of the whole assembly sequence is not enough, particularly when the assembly time is highly related to the labor cost and assembly layout design, which may result in a sequence design with high uncertainty. Chen et al [46] introduced Decision-Based Design (DBD) to maximize the value of a designed artifice under uncertainty and risk. Hence, this method can be applied to achieve more robust assembly sequence design with quantification of the predictive uncertainty.

3.3 PREDICTION MODELS DEVELOPMENT

3.3.1 Stepwise regression

With the experimental data, stepwise regression is used to select the best combination of variables to predict the response. During the analysis, one predictor is removed or added at each time, where the predictor could be a *variable* or *interaction* term. The process of selecting significant factors stops when the predicted model cannot be improved in terms of increasing the correlation, or adj-R^2 value. All variables and interactions are tested by a F-test to fit the model, and a significant factor are kept if the P-value given by F-test is less than 0.05. The first step for stepwise regression is to choose a constant model as the initial equation. In this research, the initial equation for each of the four assembly actions is comprised of the first-order and second-order terms of all factors.

Based on the experimental data, four polynomial regressions are generated for predicting the four action times: move, install, secure and rotate. All four time regressions that are based on the quadratic model improved by the stepwise regression method are shown in Table 3.1 with the estimate parameters, p-values and the adjusted R^2 values.

Table 3.1: Predictive time models for move, install, secure and rotate.

Move (Adjusted R^2 = 0.652)			Install (Adjusted R^2 = 0.704)			Secure (Adjusted R^2 = 0.839)			Rotate (Adjusted R^2 = 0.695)		
	Estimate	p-Value		Estimate	p-Value		Estimate	p-Value		Estimate	p-Value
Intercept	2.016	0.517083	Intercept	-9.171	2.46E-07	Intercept	15.333	8.84E-33	Intercept	164.214	2.74E-14
x1	-0.584	0.001838	x1	-0.504	0.004042	x1	-0.212	0.681829	x1	3.937	0.118648
x2	0.210	0.000304	x2	3.507	7.12E-09	x2	10.109	2.39E-25	x2	-57.366	4.86E-12
x3	1.460	1.07E-07	x3	1.786	5.8E-05	x3	-2.810	2.23E-10	x3	-5.767	0.058292
x4	-1.413	1.09E-08	x4	-6.081	4.29E-08	x4	2.237	1.8E-07	x4	63.902	1.05E-11
x5	-1.289	0.101155	x5	0.345	0.025236	x5	-14.968	6.32E-14	x5	0.799	9.3E-07
x1:x5	0.153	0.00058	x6	-5.660	2.06E-08	x6	0.403	0.143885	x1:x3	2.040	8.61E-08
x3:x5	-0.387	2.42E-09	x1:x2	-0.345	2.19E-09	x1:x2	-1.380	1.9E-07	x2:x4	-11.660	3.37E-08
x4:x5	0.276	1.96E-06	x1:x5	0.206	8.92E-06	x1:x3	-1.330	3.25E-09	x1^2	-0.836	2.87E-05
x3^2	0.033	6.59E-08	x2:x3	-0.827	4.23E-05	x1:x4	0.976	5.25E-06	x2^2	5.054	3.57E-10
x4^2	-0.014	0.018477	x2:x4	1.387	4.68E-10	x1:x6	0.283	3.51E-05	x3^2	-1.159	4.06E-09
x5^2	0.205	0.044231	x2:x5	-0.280	4.69E-11	x2:x5	-2.015	6.91E-12	x4^2	7.023	1.47E-05
			x2:x6	1.049	0.00015	x3:x5	9.188	0.000137			
			x3:x6	-0.651	0.00488	x3:x6	0.345	0.004765			
			x4:x6	-1.211	1.3E-10	x4:x5	-3.707	0.003229			
			x5:x6	0.385	8.12E-07	x4:x6	-0.409	4.64E-05			
			x1^2	0.175	1.78E-08	x5:x6	-0.977	0.035027			
			x3^2	0.645	2.74E-05	x1^2	0.769	9.86E-19			
			x4^2	-0.952	8.76E-09	x2^2	2.383	2.25E-22			
			x5^2	0.039	7.38E-16						
			x6^2	0.573	1.07E-06						

The p-value of each factor is used to examine the significance level, which also shows the interaction effects. The significant variables and interactions are indicated if the p-value is less than 0.05. Thus, variables and interactions with p-values less than 0.05 are suggested to be kept in the regression function. Some of the variables, such as *moving distance* (x5) in move model and *weight* (x1) in rotate model which should have an effect on the assembly time, show insignificant p-values. The reason for that is because the scale of the experiment is small, so their effects are not obvious. Nevertheless, these variables still need to be considered in the model for further research. The adjusted R-squared value of all four models are 0.652, 0.704, 0.839 and 0.695, which indicates that the fitness of the chosen model of this process is suitable.

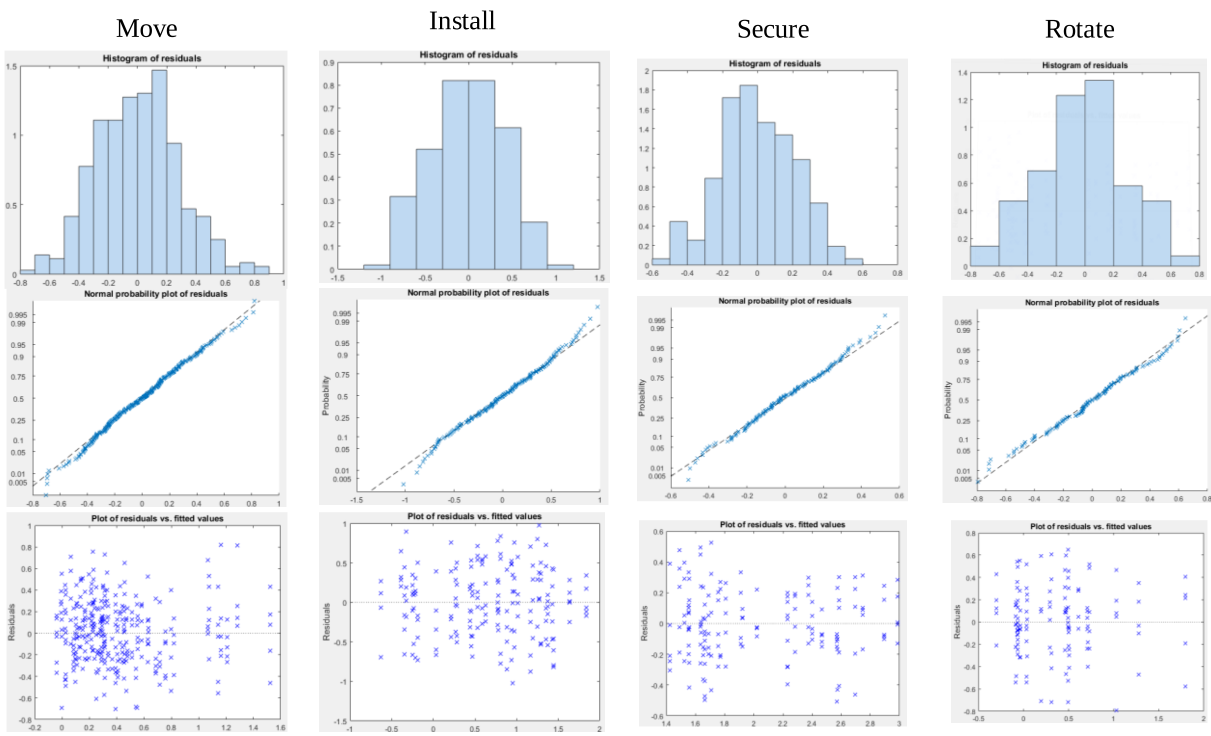


Figure 3.1: Histogram of residuals, Normal probability plot of residuals and residuals vs. fitted values plots for four time prediction models

A normal probability plot of residuals is shown in Figure 3.1. The plot shows the error terms of the regression models are approximately normally distributed along a least-square line. Thus, it is reasonable to assume that no serious assumptions are violated under the analysis. The histograms of residuals show there is no obvious outlier in the modified model. These models were built using the training data, and three testing data sets were used to test the prediction models.

3.3.2 Gaussian Process

Gaussian Process (GP) is a machine learning technique which is used in this study. There are some advantages to the GP method. First, GP can predict the time with a confidence interval (CI), so people can understand the confidence level of the prediction. For a GP model, the width of the predictive CI can indicate the density of the training data. The narrow CI indicates high confidence of the prediction and wide CI shows the density of the training data in the predictive space is not sufficient. This is very helpful information to present the uncertainty in the model to the designer for decision making. Second, GP can integrate with a base function to make more precise predictions. Since the linear regression model is estimated previously, it can be used in the GP model as a basis function. During the parameter tuning process, the mean function is calibrated for more accurate prediction.

The predictive distribution that the GP generates is defined as

$$\mathrm{g(x)} \sim \mathrm{GP}\ (\mathbf{h}(\mathrm{x})^{\mathrm{T}}\mathbf{b}, \mathrm{K}(\mathrm{X}, \mathrm{X}') + \mathbf{h}(\mathrm{x})^{\mathrm{T}}\mathrm{B}\mathbf{h}(\mathrm{x}')) \tag{4}$$

By given the testing data X_*, training data X and basis function $\boldsymbol{h}$, the predictive distribution returns a Gaussian distribution with the mean and variance shown below:

$$\overline{g}(x_*) = H_*^T \beta + \ K_*^T K_y^{-1}(y - \ H^T \overline{\beta}) \tag{5}$$

$$cov(g_*) = cov(f_*) + R^T (HK_y^{-1}H^T)^{-1}R \tag{6}$$

For the prior distribution, the mean is assumed to be the basis function, and the covariance matrix is generated by the kernel function K. The covariance matrix is a $(n + n_*)$ by $(n + n_*)$

matrix where n is the number of training inputs and n_* is the number of testing inputs. This kernel function needs to be determined for calculating the covariance between all pairs of inputs. For this study, the kernel function is shown below:

$$K\left(X^i, X^j,\right) = \exp\left(2\sigma_f\right) \exp\left(0.5\left(\frac{X^i - X^j}{l}\right)^2 + I\sigma_n\right) \quad (7)$$

The kernel function contains three variables: the signal standard deviation σ_f, the length-scale vector l and the noise standard deviation σ_n. In general, σ_f decides the width of the prediction confidence interval, and l determines the closeness of the input training data in its own dimension. For instance, with a relatively large l, the data is considered as close to each other, so the prediction is more dependent upon its neighborhood data. Also, σ_n defines the noise level of the prediction model. To build an appropriate prediction model, these three variables need to be optimized. The noise level is typically assumed to be zero in surrogate modeling approaches, since the sampling from a computer model is assumed to be noise-free. It will be non-zero when fitting a model to experimental data.

For defining unknown parameters σ_f,l, σ_n and β in the model, minimizing the negative of the log likelihood is used in this study. The negative log likelihood objective function is selected from the GPML toolbox and is shown below:

$$F_{objective} = \frac{1}{2}\vec{\alpha} \cdot (\vec{\mathrm{y}} - \beta\mathrm{h}) + \sum \log L_{ii} + \frac{1}{2}\log(2\pi\sigma_n) \quad (8)$$

Where:

$$\mathbf{L} = \text{Cholesky}\left(\frac{\mathrm{K}}{\sigma_n} + I\right) \quad (9)$$

$$\vec{\alpha} = \mathbf{L}^T \backslash (\mathbf{L}(y - m\beta\mathrm{h})/\sigma_n \quad (10)$$

By minimizing this objective function, the optimal parameters can be estimated so the prediction model can fit the training data properly. A Nelder–Mead numerical method is used for optimization. However, the GP model is known to include some local minima problems [47]. To overcome this, a Latin Hypercube (LHC) sampling method is conducted for generating many different starting points for the optimization. In this study, two hundred starting points are drawn from LHC and optimization takes the samples and input data to minimize the negative log marginal likelihood function. The training data sets which were used for building regression models are also used as training data for the GP method. The trained models generate predictive results based on the same testing data. The optimized parameters and the objective function values for each of the models are shown in Table 3.2 with optimization run time, MSE, function object value and number of iterations.

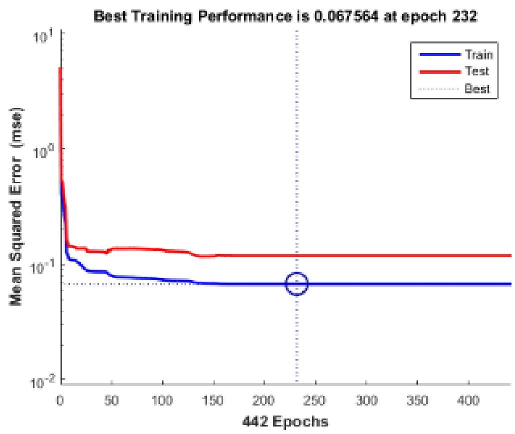
Table 3.2: Optimized GP parameters for four time prediction models with optimization information

Move		**Install**		**Secure**		**Rotate**	
σ_f	0.807	σ_f	0.292	σ_f	0.472	σ_f	0.466
l_{weight}	0.265	l_{weight}	0.392	$l_{OBB\ max\ surface}$	0.532	$l_{OBB\ max\ surface}$	0.626
$l_{OBB\ vol}$	0.606	$l_{OBB\ vol}$	0.171	$l_{OBB\ min\ surface}$	0.352	$l_{OBB\ min\ surface}$	0.847
$l_{OBB\ max\ surface}$	0.426	$l_{OBB\ max\ surface}$	0.712	$l_{thread\ number}$	0.893	$l_{thread\ number}$	0.747
$l_{OBB\ min\ surface}$	0.987	$l_{OBB\ min\ surface}$	0.712	$l_{inser\ distaince}$	1.033	$l_{rotate\ angle}$	0.787
$l_{moving\ distance}$	0.907	$l_{moving\ distance}$	0.612	l_{nut}	0.111	σ_n	0.245
σ_n	0.054	$l_{align\ features}$	0.773	$l_{tool\ effect}$	0.572	β	0.054
β	1.107	σ_n	0.213	σ_n	0.153		
		β	1.053	β	1.033		
Run time (s)	58.95		45.09		36.85		48.5
MSE (log)	0.123		0.329		0.068		0.067
f_{obj}	195.75		192.75		135.75		195.75
Iterations	576		616		620		568

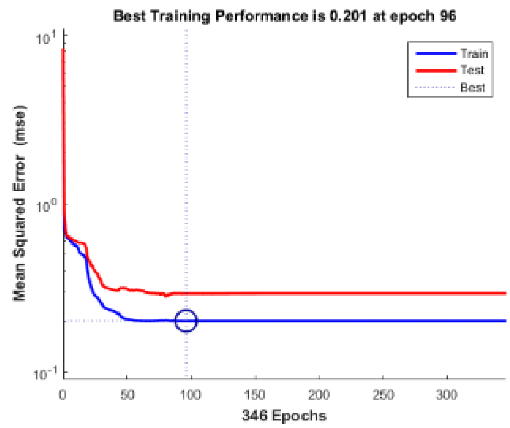
3.3.3 Artificial Neural Network

The Artificial neural network (ANN) is a biologically inspired machine learning method that is applied for function approximation, classification data processing, etc. An ANN is constructed by artificial neurons which represent the sigmoid functions. They are connected with coefficients (weight) and placed in layers. Usually an ANN contains three kinds of layers: input layer, output layer, and hidden layer(s). The ANN receives input data through the input layer, then the hidden layer processes the input data and passes the results to the output layer [48]. By optimizing the weights between the neurons in the ANN, accurate results can be obtained.

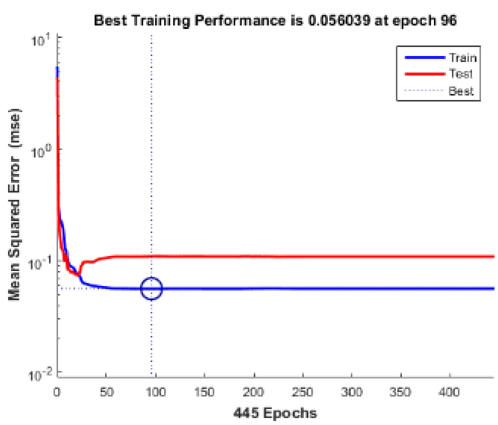
In this study, four ANNs are built for predicting the times for moving, installing, securing, and rotating actions. Based on the previous studies [49,50], the assembly time functions is represented by the ANN models that contain a single hidden layer with $2n + 1$ nodes, since it is assumed that the time functions should be continuous and n is the number of the input features. The four ANNs are implemented by using the MATLAB artificial neural network toolbox [51]. Seventy percent of the data is used as training data and ten percent of data is used as the validation set during the training process to prevent overfitting. The rest of the data are used as testing set. Considering the noise of the experimental data, Bayesian Regularization is used for training the networks to minimize the Mean Squared Error (MSE), since this method can produce better generalizations for noisy datasets [51]. Also, this training method has been proven to be more robust than standard back–propagation and can prevent overfitting data [52]. A gradient of 10^{-6} is used for the training. The training history for the four models is shown in Figure 3.2. Compared to the GP models, the training times of the ANNs models are shorter, which are 12s, 10s, 10s and 19s for the move, install, secure and rotated models.



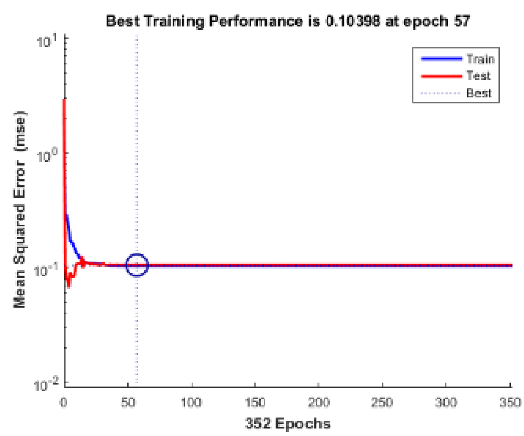
(a) move



(b) install



(c) secure



(d) rotate

Figure 3.2: Training performance for the (a) move, (b) install, (c) secure and (d) rotate ANN models

3.4 RESULTS

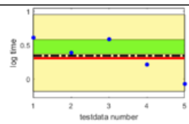
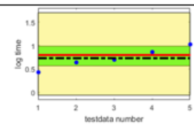
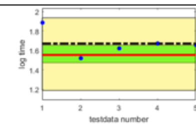
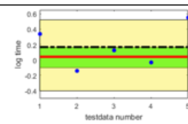
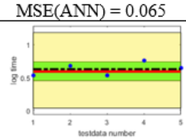
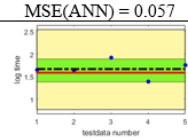
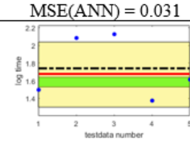
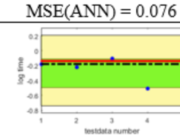
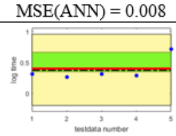
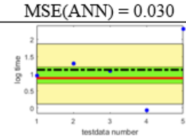
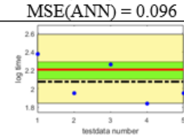
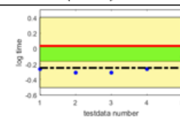
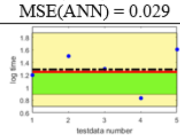
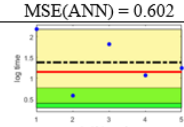
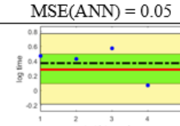
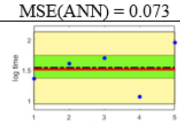
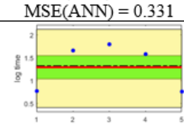
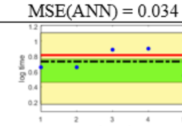
In this section, the predictive assembly action times which are generated by regression, GP and ANN models and will be compared with the testing data. Also, a case study of a tessellated oil pump model is conducted for testing these two models and the DFA time estimation method is applied to this model for comparison. The purpose of applying the DFA method is to ensure the predictive results that regression and GP models generated are not significantly different from this well establish time estimation method, while all of the predictions are generated automatically from a tessellated model containing limited information.

3.4.1 Time Models

Ten percent of the experimental data set from each of the action tasks are set as testing data for the verification of the predictive results from the regression and GP model. Table 3.3 shows the 95% confidence intervals (CI) of a regression prediction (green) and a GP prediction (yellow). The ANNs predictive means are represented by the red lines. The predictive means for the regression and GP are not presented in the plot for simplification. The log-transformed action time from five different experimental study participants and the mean response are represented by blue dots and dash line in each of the data sets. The mean square errors (MSE) between these three models are not significantly different in most of the cases, and the MSEs are small. However, in some of the tests, such as test 4 in move and install predictions, the calibration of the regression model with the GP method results in more accurate predictions with smaller MSE values. Meanwhile, ANNs shows similar results compared to the GP models.

Of note is that the 95% CIs from the GP are consistently wider than from the regression prediction, while they all successfully capture the mean response. The reason for this is the GP includes the noise explicitly in the model, resulting in higher variances. Recalling the GP kernel function (Eq.7), the signal standard deviation σ_f and the noise standard deviation σ_n are optimized so the trained GP model is able to represent the experimental data noise and the person to person variance. Since all the uncertainties can be represented by the variances, the whole assembly planning is optimized by the multi-objective robust reliability-based design optimization method.

Table 3.3: Predictive results of the regression GP and ANN models: 95% CIs of GP (yellow regions), 95% CIs of regression (green regions), ANNs predictive means (red lines), training data (blue dots) and training means (dash lines)

test	Move	Install	Secure	Rotate
1	 MSE(GP) = 0.065 MSE(REG) = 0.081 MSE(ANN) = 0.065	 MSE(GP)=0.05 MSE(REG)= 0.044 MSE(ANN) = 0.057	 MSE(GP)=0.026 MSE(REG)= 0.026 MSE(ANN) = 0.031	 MSE(GP)=0.074 MSE(REG)= 0.081 MSE(ANN) = 0.076
2	 MSE(GP) = 0.008 MSE(REG) = 0.008 MSE(ANN) = 0.008	 MSE(GP) = 0.03 MSE(REG) = 0.031 MSE(ANN) = 0.030	 MSE(GP) = 0.1 MSE(REG) = 0.119 MSE(ANN) = 0.096	 MSE(GP) = 0.05 MSE(REG) = 0.057 MSE(ANN) = 0.044
3	 MSE(GP) = 0.028 MSE(REG) = 0.044 MSE(ANN) = 0.029	 MSE(GP) = 0.592 MSE(REG) = 0.603 MSE(ANN) = 0.602	 MSE(GP) = 0.0617 MSE(REG) = 0.057 MSE(ANN) = 0.062	 MSE(GP) = 0.046 MSE(REG) = 0.041 MSE(ANN) = 0.05
4	 MSE(GP) = 0.072 MSE(REG) = 0.11 MSE(ANN) = 0.073	 MSE(GP) = 0.324 MSE(REG) = 1.037 MSE(ANN) = 0.331		 MSE(GP) = 0.036 MSE(REG) = 0.036 MSE(ANN) = 0.034
5	 MSE(GP) = 0.092 MSE(REG) = 0.092 MSE(ANN) = 0.093	 MSE(GP) = 0.205 MSE(REG) = 0.203 MSE(ANN) = 0.164		 MSE(GP) = 0.028 MSE(REG) = 0.034 MSE(ANN) = 0.024

The objective not only includes the assembly time, but the uncertainties such as data noise, worker variances are also considered. Because uncertainties can be in the different metrics, constraints such as assembly time or labor cost can be set up and handled through a probabilistic formulation. By applying the Decision-based Design method [46], the designer can easily generated assembly plantings with different trade-offs among the multiple objectives and different risk levels (i.e. levels of uncertainty).

3.4.2 Case Study of Piston Compressor

In this this section, a tessellated oil pump model is used to for testing the regression, GP and ANN models. Also, the DFA time estimation method is applied to this model for comparison. The purpose of applying the DFA method is to ensure the predictive results that these three models generates are not significantly different from this well establish time estimation method, while all of the predictions are generated automatically from a tessellated model that has limited information.

Tesk	Action
1	4 to 1
2	5 to 1
3	10 to 1,4,5
4	7 to 10
5	9 to 8
6	6 to 9
7	6,9,8 to 7,1
8	3 to 1
9	2 to 1
10	Secure 9 bolts

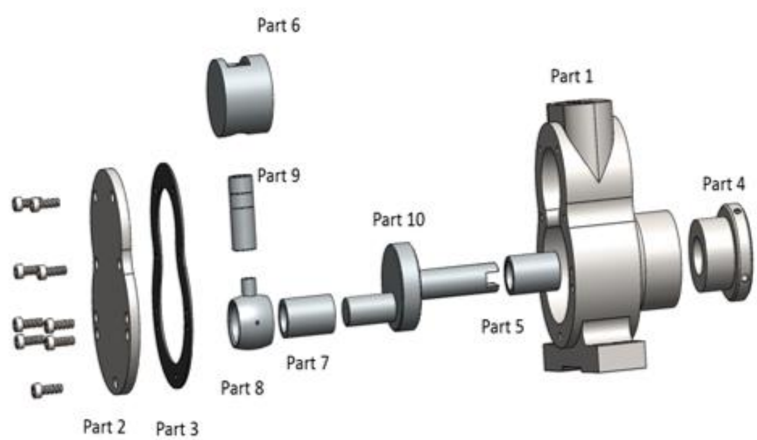


Figure 3.3: Exploded view of pump model

These input variables are applied to the regression and GP models for predicting the results. The move and install action are all included from tasks 1 to 9, and task 10 is securing all nine bolts.

The exploded view of the pump assembly is shown in Figure 3.3 with the assigned assembly sequence. For each task, the input variables for the move, install, secure and rotate models are extracted from the subassembly. For example, in task 3, part 10 is a moving part that needs to be inserted to the subassembly which includes part 1, 4 and 5. This process involves move and install actions, and the total assembly time for this task is the sum of the moving and installation times from the two prediction models.

Table 3.4: Predictive results compared with DFA.

Task	Task process	DFA (s)	Regression(s)	95% CI	Err	GP (s)	Err	95% CI	ANN(S)	Err
1	4 to 1	3.3	2.76	±2.37	-16%	2.68	-19%	±7.67	3.40	3%
2	5 to 1	4.43	3.27	±2.42	26%	3.33	-25%	±7.53	2.11	-52%
3	10 to 1,4,5	3	3.33	±2.33	11%	3.15	5%	±8.21	4.31	44%
4	7 to 10	2.93	3.25	±2.41	11%	3.29	12%	±7.59	2.11	-28%
5	9 to 8	3.3	2.96	±2.5	-10%	3.62	10%	±8.07	2.65	-20%
6	6 to 9	3	2.68	±2.39	-11%	2.51	-16%	±8.06	2.48	-17%
7	6,9,8 to 7,1	7.1	2.89	±2.36	-59%	2.57	-64%	±6.86	2.51	-65%
8	3 to 1	3.86	7.38	±8.84	91%	8.45	119%	±21.16	9.14	137%
9	2 to 1	3.75	3.15	±3.90	-16%	2.96	-21%	±11.41	2.17	-42%
Total assembly time		**34.67**	**31.67**		-9%	**32.57**	-6%		30.86	-11%
10	Secure 9 bolts	49.7	68.49	±9.81	38%	68.70	38%	±19.62	67.75	36%
Total time		**93.37**	**100.16**		7%	**101.27**	8%		98.61	6%

The predictive time and the 95% CI for each of the pump assembly task are shown in Table 3.3 and they are compared with DFA method. For task 1 to 9, the predictive times from the regression and GP models are closer to the DFA result compared to the ANN, while most of the errors are within twenty percent. For ANN, most of the prediction errors are greater than 20%. Considering the estimated ±50% error typical of DFA methods [34], the errors in ANN are still acceptable. The predictions for task 7 and 8 have very high error percentages for all of the prediction models. There are two reasons cause these differences. First, the training data is not sufficient: the CIs in GP models for task 7 and 8 are higher than the others, which indicate the density of the training data in that predictive space is not sufficient. Second, the DFA method assigns a higher insert time for installing the subassembly of 6, 9, 8 in task 7 since the parts in this subassembly have high degrees of freedom and are difficult to be aligned. Such information is hard to capture in the tessellated model and it is not considered in the prediction models currently.

On the other hand, the predictive times of the secure action also have a high error percentage, which is 38 percent for both prediction models. Due to half of the parts in this pump assembly being fasteners, this accumulated error is considerable. To reduce this error, more training data are required to improve this model accuracy. At last, the total prediction times of the regression and GP model have 7 and 8 percent error, respectively, compared to the DFA method.

3.5 ONLINE GAUSSIAN PROCESS MODEL UPDATING

The developed Gaussian Process model is able to produce accurate time predictions for small scale assemblies based on the current training data from manual assemble experiment. Nevertheless, when it is applied to predict assembly time for bigger product, the accuracy may decrease due to the insufficient training data from larger scale experiment. Hence, new data is required for model improvement in future applications. However, the model updating duration for traditional Gaussian Processes will increase exponentially (O(n3)) while more training data is used. This is not desirable since our objective is to develop a real-time prediction model for various applications and this requires fast computational time. For this reason, an *Online Gaussian Process* (OLGP) method is implemented in this study. OLGP can have constant updating, even with increasing amounts of training data. The main difference between GP and OLGP is the training data for OLGP is clustered into different subsets, and each of these subset is used to generate a GP model. The advantage of using OLGP is the data points in each of the clusters are from the assemblies that have similar scales, so it allows the prediction model to be more precise by adapting the scale of the input data.

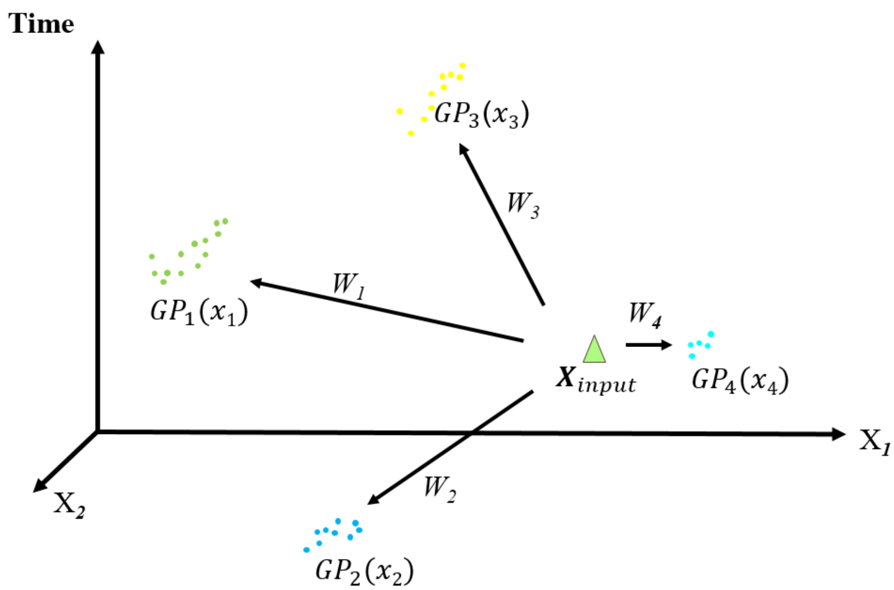


Figure 3.4: A general structure of OLGP with input variables

To apply OLGP, first, the training data is clustered into different groups. The maximum number of data points is assigned for each group and a GP model is generated for each group. An example of OLGP with four clusters is shown in Figure 3.4. While predicting the expected time, a distance measure W_i which indicates the distance between inputs X_{input} and the training data group k is calculated by the equation below:

$$W_i = \exp\left(0.5\left(\frac{x^i - x^j}{l}\right)^2\right) \tag{11}$$

where l is the length-scale vector and the same as in Eq.9 . Once the distance measures are generated, the predictive mean $\bar{y}$ and standard deviation $\bar{\sigma}$ can be obtained by the equations below:

$$\bar{y} = \sum_{k=1}^{N} w_k y_k \,/\, \sum_{k=1}^{N} w_k \tag{12}$$

$$\bar{\sigma} = \sum_{k=1}^{N} w_k \sigma_k \,/\, \sum_{k=1}^{N} w_k \tag{13}$$

where y_k and σ_k are the predictive mean and SD which are calculated by Eq.5 and Eq.6 with the training data from group k. The advantage of using this approach is that predictions that come from different GP models can be weighted. Thus, the GP models in which the training data have similar scales to the input data will have more weight when making a prediction. Meanwhile, instead of updating a single GP model which is developed based on the entire training data, the update process only occurs for the sub-model in which the training data is close to the new feedback data point.

3.6 CONCLUSION

In this chapter, we aim to develop an automated time prediction tool for evaluating assembly times for sequence. An assembly process is defined as a combination of four operations: Moving, Installation, Securing, and Rotation. Input features for the prediction models are designed and can be automatically detected from the 3D model based on its geometric features and without any user inputs. Hence, the assembly sequence could be evaluated based on a more automated technique. To obtain training data for model training, experiments for assembling a chain saw, an internal combustion engine and an airplane seat were conducted. With the experimental data, the polynomial regression, GP and ANN methods were used to construct time prediction models for each of the actions. The results show these three models model have similar performances, but the regression model has higher MSE for some of the predictions. Meanwhile, the 95 percent CI of the GP model successfully captures the person to person variance during the experiment. To demonstrate, a tessellated pump model is used for a case study. The predictive assembly time from these three models were compared to the well-established DFA method to ensure the predictive results from automated methods are reasonable and accurate. The GP model had the lowest error of 6 percent compared to the DFA total assembly time. On the other hand, these three models had high errors, between 36 to 38 percent, while predicting the total secure time.

CHAPTER 4 : ASSEMBLY STABILITY

4.1 INTRODUCTION

Digital manufacturing has become more important due to its ability to predict downstream times and costs from early design phases. With rapid development of software and hardware, many assembly processes can be simulated to estimate the cost and time before production. Such automated approaches could come in the form of assembly sequence planning, facility planning, and assembly tool and fixture planning [3]. Such tools are more useful when they better predict time and cost estimates.

For product assembly sequence planning, assembly time is perhaps the most commonly used factor for evaluating the quality of the assembly plan. To estimate the assembly time, many studies [6–8,10,13,53] have developed different time prediction models to estimate assembly time based on human motion and design of the assembly. However, optimizing assembly time in the assembly planning process is not sufficient to generate a reliable assembly plan. Focusing only on assembly time optimization may generate unstable solutions which require a significant amount of fixturing during the assembly process or even result in infeasible assembly task. To describe this problem, a simple example is given in Figure 4.1 which shows two sequences that can be used to assemble parts A, B and C. When optimizing the assembly plan by only minimizing the assembly time, the sequence 1 will be preferred because of its faster installation time. This installation is faster because in sequence 1, part B is not blocked by part C and the final location of B is more accessible from the outside. Nevertheless, in terms of stability, sequence 1 is not a good choice since part B can roll out of place if there is any movement during the A-B and C installation. Since situations like these arise more often in complex products, stability should be considered during the search of optimal assembly plans.

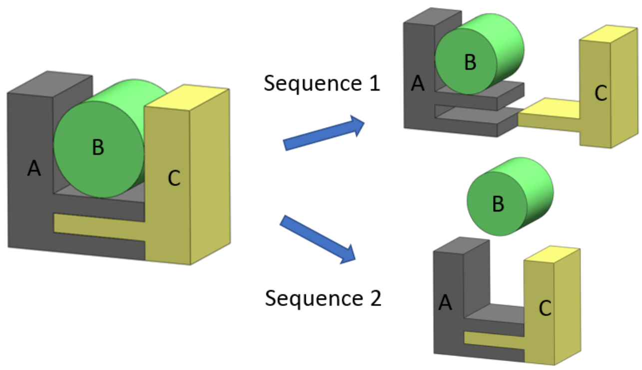


Figure 4.1: Two assembly sequences with different stability

In this study, we propose an automated evaluation method that can evaluate the assembly stability for tessellated CAD models. Tessellation is the most popular CAD format and has been used in a variety of different applications because of its simplicity

The theory-based stability involves detecting the degrees-of-freedom (abbreviated here as DOF) between mating parts and it has been used in many studies for stability evaluation. The usefulness of a calculated DOF is that it can quickly indicate the unconstrained translation and rotation motion. In this study, the DOF is generated automatically by analyzing the primitive shapes detected between mating surfaces. Moreover, tip-slide difficulty is introduced to estimate the assembly parts' stability during transportation.

Although using theory-based method is straightforward, the absence of spatial kinematic between subassemblies may result in a less stable assembly plan. Some research [54–58] have studied the system stability by applying linear programming methods to evaluate the stability status of a given assembly. In this chapter, we introduce an approach to simulate the kinematic behavior of the assembly during assembly process using physics simulation. By carefully defining the kinematic constraints, the spatial movement of every part within the assembly is simulated. This spatial movement analysis is used to capture the displacement of the assembly parts and then for evaluating systematic stability.

4.2 LITERATURE REVIEW

Assembly stability is one of the most important factors in evaluation of assembly sequences [59] and the subassemblies are constantly evaluated by this characteristic [60,61]. Various approaches have been studied for estimating the subassembly stability during assembly planning generation. One of the most popular methods is to use a predetermined stability index (SI). In Lee and Shin's study [17], SI is calculated based on the DOF and part's relative stability. To determine part's relative stability, first, Lee and Shin defined seven types of connections, attach, sticky, force-fit, push & twist, screw, connectors and weld, to describe how two parts are connected. Meanwhile, for each of these connections, the stability score is assigned based on the mating type of the part which can be categorized into "insert", "semi-insert" and "place-on". Another similar approach is conducted by Wang, et al. [18] and Dong, et al. [19]. The SI in these studies is assigned based on the restricted DOF and connection strength. On the other hand, the generation of SI is not fully automated since the assembly interference still needs to be extracted manually [18]. Also the SI only describe stability by predetermined connection and ignore the actual connected geometry which also has may impact on the stability. Meanwhile, the stability that before some secure actions such as screw and weld is not mentioned.

Separate from the SI approach, there are studies that evaluate the stability by static equilibrium. One of the earliest methods for evaluating assembly stability is developed by Blum, et al. [62] in 1970. In this study, an automated evaluation method is developed to assess the stability of the stacked blocks by solving force and moment equilibrium equations. Meanwhile, to simplify the formulation of the stability problem, they assume that the reaction forces between blocks can only occur at the vertices of the blocks and intersection of line segments. More recently, linear programming is introduced by Mattikalli, et al. [54] to formulate the frictionless assembly stability problem in the form of an optimization problem. By a given gravity direction, they solved the assembly stability problem by optimizing the potential energy and reaction force. The same authors also extended their study in [55] by including friction in the stability model introduced in [54]. Mattikalli et al. relocated the reaction forces between connected faces of mating parts to the surrounding vertices of the shared area. Then, they determined and solved the friction by using linear programming. The linear programming approach is also used to solve the stability problem while motion path planning is involved. Rakshit and Srinivas [56] applied Stewart–Trinkle [57] model and Baraff's model [58] to simulate the friction and frictionless disassembly process.. Bernheisel [63,64] developed an algorithm to detect the stability of a stack of polygonal parts during transportation. This method is also applied to path planning optimization problem. Different from the methods above which are used to evaluate the stability by solving force-closure problem, form closure formulation is also used to determine the immobility of assembly. For example, Cheong et al. [65] determined bounds on the number of contacts required to immobilize an assembly which is a chain of two-dimensional hinged polygons. Meanwhile, more robust immobilization method is also developed for the contact points that have small perturbations. Our method is distinct from these since we attempt to evaluate the stability during every evaluation within the tree searching process for assembly planning which involves large amount of arbitrarily defined tessellated solids, so the complexity of the assembly is high and it is difficult to formulate force and form closure problem. It is also completely automated while previous methods appear to be solved partially by human intervention.

Recently, with the development of virtual reality technology, the assembly process can be conducted in a virtual environment and tested by physics simulation engines. In Jayaram et al.'s study [66], a virtual assembly design environment is constructed. The interactions between parts are used in this study so the part can slide and rotate during assembly process. Meanwhile, physical based modeling is used for simulating the dynamic behaviors. More recently, more sophisticated physics engines such as Bullet Physics [67], PhysX [68], and Open Dynamics [69] have been developed and used in different research projects [20,70–72] for stability testing and assembly planning. Regardless of the advances, none of these methods are found to develop an automated evaluate method for stability estimation on a complex assembly using physics simulations. In this work, we aim to deliver a comprehensive stability evaluation model which overcomes the limitations of the existing approaches by combining theory-based and simulation-based approaches.

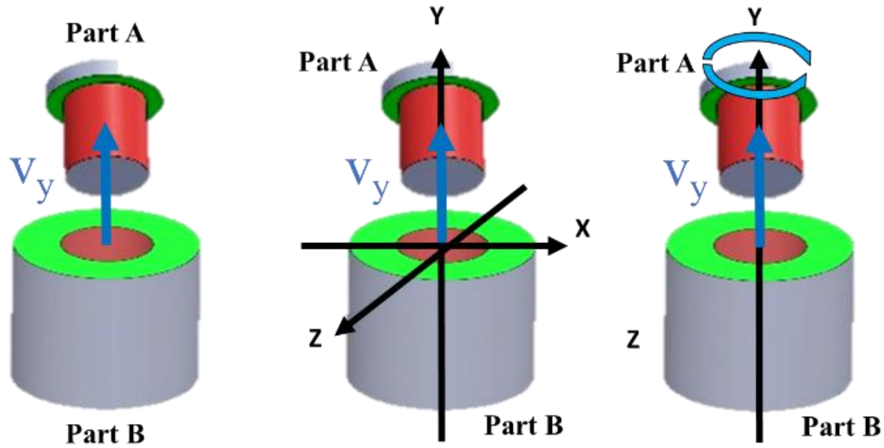
4.3 THEORY-BASED STABILITY

In this study, degrees-of-freedom (DOF), tip and slide difficulty are considered as the three factors that are related to an assembly's stability. The DOF can indicate the restriction of the translation and rotation movement of a part. In this study, we propose a new method that can obtain the DOF as well as tip and slide difficulty from the tessellated models automatically by using its free directions and connected primitives. Finally, an evaluation function that includes DOF, tip and slide difficulty is designed and used to estimate the stability during assembly planning generation.

4.3.1 Degree-of-Free Determination by Connected Primitive Surfaces and Part-to-Part Free Directions

Connected primitive surface and Part-to-Part direction are the two key elements for determining the DOF of two connected tessellated parts. In this study, the proposed method is able to identify the linear and rotational actions between two components based on their connected primitives and free directions. Meanwhile, these actions are quantified by DOF and used for evaluating the stability of the connection. The acquisition of primitive information is implemented in our previous study [73] that a fuzzy logic-based mesh segmentation method can accurately classifies a tessellated mechanical CAD models into five kinds of primitives: flat, cylinder, cone, sphere and curve. With the classified primitives, the connected primitives between two components can be easily obtained and can be used for representing the kinematic constrain.

Free directions are used here for determine the DOF coordination and representing the linear movement. They can be obtained based on the shared primitives between two part. For instance, if a physical connection is detected between a positive cylinder and a negative cylinder primitives, the free direction will be the two opposite directions along the centerline of the cylinder. By using this methodology, free directions can be easily generated with different combinations of contacted primitives.



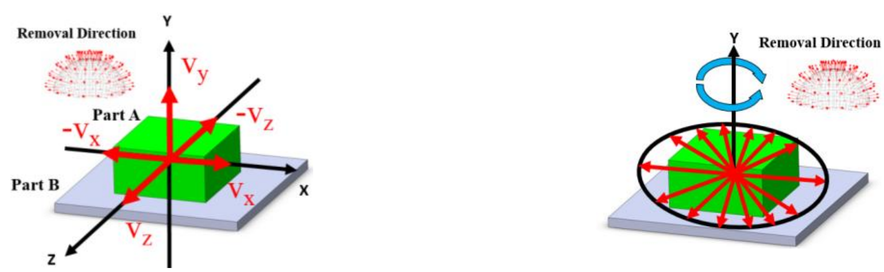
(a) (b) (c)

Figure 4.2: Process of determining the degree-of-freedom (DOF): a free direction (a) of Part A is used to conduct the X-Y-Z coordination for assigning translation DOF (b), and the rotational DOF is defined by the cylinder primitives (c)

An example in Figure 4.2 is used to describe a general process of calculating the DOF by using free directions and connected primitive. In this example, there is one free direction for removing part A from part B which are connected by the flat and cylinder primitives. For calculating DOF, first, the coordination for referencing the DOF need to be generated. In this case, the free direction, V_y, is used to represent the linear movement and Y axis of the coordination can be located along this direction. Due to there is not free direction perpendicular to the Y axis, the X and Z axis can be randomly assigned (Figure 4.2b). Otherwise, one of these two axes need to be aligned with the free direction that perpendicular to the Y axis. After deciding the coordination, we can assign 0.5 linear DOF to the Y axis since part A can only move along with the positive Y direction. Finally, the rotation DOF can be defined by the connected cylinder primitives. Since part A can rotate about the cylinder which's axis is on Y axis and it does not interfere with the flat primitive, 1 rotate DOF can be assigned to Y axis (Figure 4.2c). Finally, the DOF for part A in this assembly is equal the sum of the linear and rotate DOF which is 1.5.

In this study, we assume that the connections between two parts or subassembly can be categorized into five different kinematic constraints: planar, prismatic, cylindrical, revolute and spherical constraints. By using the proposed method, these constraints can be identified within tessellated assembly and their stabilities can be represented by DOFs and used for evaluating the internal connections of the assembly. Meanwhile, the obtained constraints are also used for implementing physics simulation for another stately evaluation approach in the later section. The generations of DOF for the five constraints is demonstrated in this section.

4.3.1.1 Planar Joint



(a) (b)

Figure 4.3 : Using removal directions and primitive information to generate DOF for planar joint. (a) Generating translation DOF (b) Generating rotation DOF

A planar joint is defined as a moving part A that moves independently of a reference part B in the manner shown in Figure 4.3a; these directions result from the removal directions explained above in Section 4.3.1. First, Y-axis is assigned along with the free direction V_y which is normal to the contacted surface of Part B. Since part A is blocked by part B in the -Y direction, the translation DOF on the Y-axis is equal to 0.5. The X-axis, assigned along V_x, has a valid opposite removal direction which is $-V_x$ since the linear movement of part A is not restricted on this axis and it results in 1 translation DOF. Finally, the Z-axis is along the cross product of V_y and V_x, and its translation DOF is also not limited. Overall, the linear translation DOF of part A is 2.5 in this

X-Y-Z coordination. After that, rotation DOF can be calculated with the primitive information. In this case, Part A is supported by the flat primitive and the rotation DOFs of X and Z axes are limited. However, all of the free directions that are perpendicular to the Y-axis are not blocked by any other primitives in Figure 4.3b. This indicates that a rotational degree of freedom about the Y-axes is available in the pair of parts and the total DOF should thus increase from 2.5 to 3.5.

4.3.1.2 Prismatic, Cylindrical Joints

Compared to planar joint that the connected parts contain multiple free directions, the process for identifying the prismatic and cylindrical joints is straight forward since they can contain two free directions in maximum. An example of prismatic joint is given in Figure 4.4a that part A contains two free directions, $\pm V_Z$, and one translation DOF can be defined on the Z-axis which is aligned with these two directions. Meanwhile, part A is blocked by the four flat primitives of part B. So, the normal of one of these faces (e.g. the top face) which is perpendicular to Z-axis can be used to generate the Y-axis Y, and X-axis can be defined afterward. Therefore, part A has only one translation DOF on Z-axis and zero rotation DOF since the rotation is limited by the four contacted primitives.

(a) (b)

Figure 4.4:Using removal directions and primitive information to generate DOF for (a) prismatic joint and (b) cylindrical joint

Like prismatic joints, the Z-axis can be quick defined for the cylindrical joint that is shown in Figure 4.4 b, also the two opposite removal directions result in one translation DOF only on the Z-axis. Additionally, these two parts are only connected by one cylindrical primitive whose centerline is aligned with the Z-axis. It is easy to know that Part A can rotate both direction about Z-axis and has one rotation DOF along Z-axis. Finally, the DOF for this prismatic joint is two.

4.3.1.3 Revolute and Spherical Joints

Unlike the three kinds of joints that we discussed earlier, the revolute and spherical joints (Figure 4.5 shows an example of each type) have no removal directions since they are used for performing rotational movements only. Therefore, DOF can be only defined by the primitive information. For the revolute joint, the axis of the cylinder primitive is set as Z-axis. Meanwhile, the two flat primitives have no interruption of the rotation action and this joint has one rotation DOF.

(a) (b)

Figure 4.5 : Revolute and Spherical joints

With the implemented classification method, sphere surface can be detected. Thus, when two components which are like the example in Figure 4.5b are only connected by the spherical primitive feature, three rotation DOF can be assigned immediately.

Overall, all the kinematic constraints that we have been observed in study can be successfully described by these five kinematic constraints. By using the proposed method, DOF can be extracted from these connections automatically and applied to the assembly planning searching process.

4.3.2 Tip Difficulty and Slide Difficulty

Tip difficulty (TD) and Slide difficulty (SLD) is used for estimating how easy or difficult a component can tip over and slide. TD is defined as the force that can cause part tipping. To calculate the TD, it is assumed that the sliding friction between two parts is infinite and consequently there is no sliding between assembly parts. The tipping action can be estimated by using the information that is automatically extracted from the CAD model. The gravity direction is set as the opposite of the average of the part removal directions in this study.

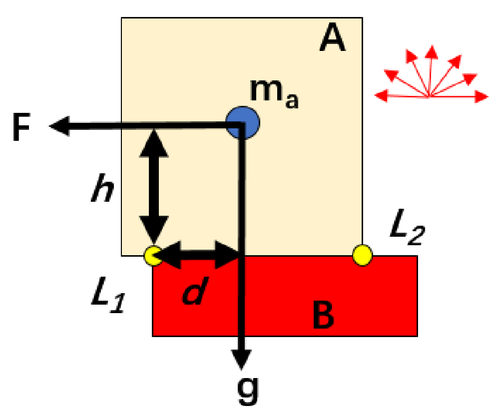


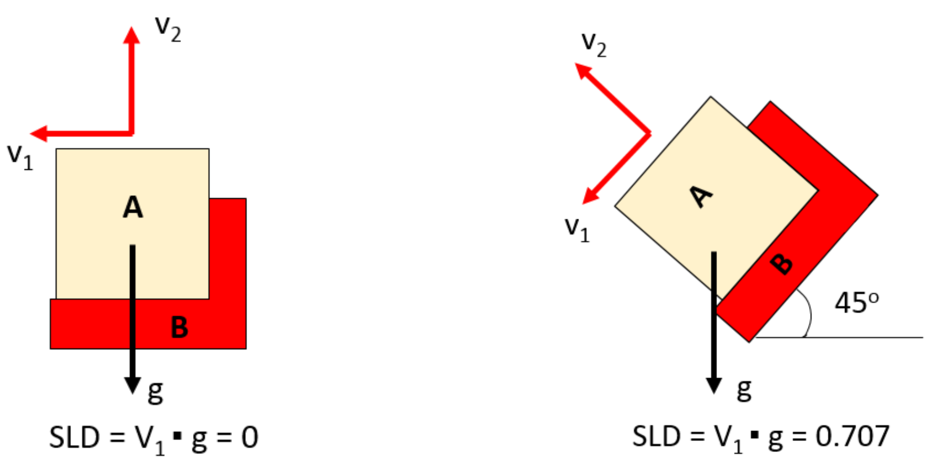
Figure 4.6: Free body diagram for calculating minimum tipping torque

For a 2D example in Figure 4.6, the removal directions for removing part A from B is shown as a half circle, and the gravity is assumed and set on the opposite direction of the sum of these vectors. Since part A is easier to rotate around L1 compared to L2, the edge L1 is automatically selected as the rotation point. The TD can be calculated by

$$\mathrm{TD} = m_a * g * d/h \tag{14}$$

where m_a is the mass of part A. Since the tessellated models do not contain any mass information, unit density is assumed for every part.

Slide difficulty (SLD) is defined as the maximum dot product between the gravity unit vector and all the removal directions, which is between -1 and 1. An example of the generation of SLD and how it is used for evaluating the sliding motion is shown in Figure 4.7. Suppose V_1 and V_2 are the two free directions for part A with the unit gravity vector g. In this case, part A will more likely to slide along the direction of V_1 instead of V_2. In this case, SLD is defined as the dot product of V_1 and gravity vector g. When the assembly is rotated (e.g. in Figure 4.7b), part A become more unstable and easier to slide. Meanwhile, SLD is increased. By using SLD, the slide difficulty can be quickly evaluated in this study.



(a) (b)

Figure 4.7: SLD increases when a component become more likely to slide

4.3.3 Combined Stability Score

With the generated DOF, TD and SLD values for a pair of parts, the combined stability score for an assembly of arbitrary parts can be calculated by the following equation:

$$F_{stability} = log(\, log(N) + \overline{DOF} - log(TD_{min}) + SLD_{max}) \tag{15}$$

where $\overline{DOF}, TD_{min}$, and SD_{max} are the average *DOF*, minimum *TD* and maximum *SLD* in an assembly accordingly. *N* is the number of components that are not fasteners in the assembly. The reason for including this variable is that we consider the stability of an assembly as related to the number of insecure parts. Meanwhile, log transformation is used in this equation to normalize the variables into similar scale. For this equation, the smaller value indicates an assembly is more stable. By using this equation, the stability of the subassembly can be evaluated in a more efficient and automated manner during assembly process planning.

4.4 SYSTEMATIC STABILITY EVALUATION

During real production processes, various motions are performed on an assembly and their effect on the stability is significant. Evaluating the assembly stability during this dynamic process is difficult due to the inconsistency of the assemble motions. To solve this problem, many approaches [20,70–72] have been applied physics engine to simulate the assembly process component for stability evaluation. In this study, we propose a novel method to evaluate the assembly stability by studying the kinematic behaver of every single part with in the assembly. This is achieved by constructing the kinematic constraint between every two components in the assembly. Based on the constraint detection method in the previous section, the process of configuring the simulation can be fully automated and the assembly process can be simulated correctly. Meanwhile, a formulation is applied to quantify the simulated behavior into stability score. Since this simulation-based approach is significantly different from the theory-based method, the results simulated results are valuable for comparing and validating the theory-based method.

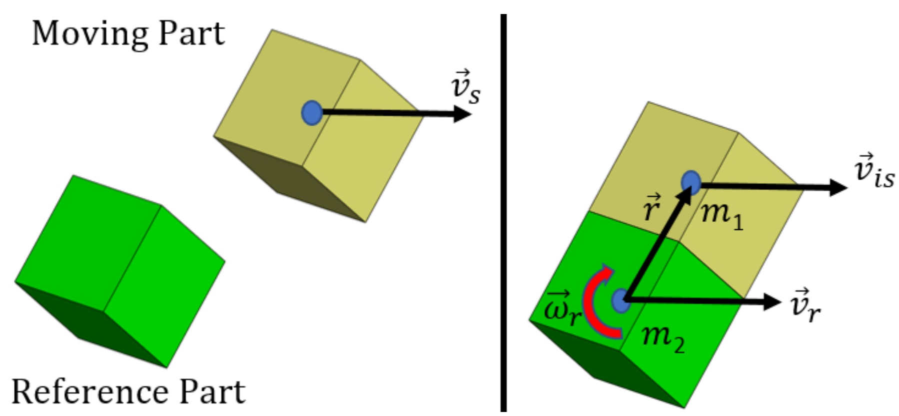
4.4.1 Importing Assembly Model to Physics Engine

To conduct more accurate simulations to approximate the spatial kinematic behavior, kinematic constraints are constructed between parts while the assembly mode is imported into the physics engine. In this study, Bullet Physics engine is used and it can simulate various kinematic constraints between parts. Three major constraint methods that are provided by the Bullet Physics are used in this study: hinge, slider and 6-Degree of Freedom (6DOF) constraints. By using these constraint method, all of the five kinematic constraints what we mentioned in section 4.3.1 can be generated. For example, when a part can only transfer or rotate about one axis (e.g. prismatic or revolute joint), slider or hinge constraint can be applied to construct this connection between two parts accordingly. When the motion is involved in multiple axes, such as planer, cylindrical and spherical joints, 6-DOF constraint can be used to set up the rotation and liner translation in different axes simultaneously.

The process of constructing these constraints is straightforward and fully automated since every necessary information including the DOF, removal direction and the local coordination have been obtained in the efforts of the previous metric. For example, the hinge constraint can be assigned directly if a part only has one rotational DOF. For slider, it will be assigned when a part has one or half translational DOF. For one translational DOF, a part can move in both opposite direction in the first state and the half translational DOF indicates the part is blocked and can only move in one direction. For a 6-DOF constraint, one rotational DOF will be assigned if a part is only blocked by a flat surface. The translational DOF is determined by the removal directions.

4.4.2 Physics Simulation

After constraints are assigned, the assembly process can be simulated in the physics environment. In this study, we assume that all the assembly tasks happen on the assembly table and are accomplished by hands. Also, subassemblies are placed in a fairly unstable state while they are being placed on the table since the acceleration changes significantly. Hence, we want to simulate the placing motion of the subassembly.



(a) (b)

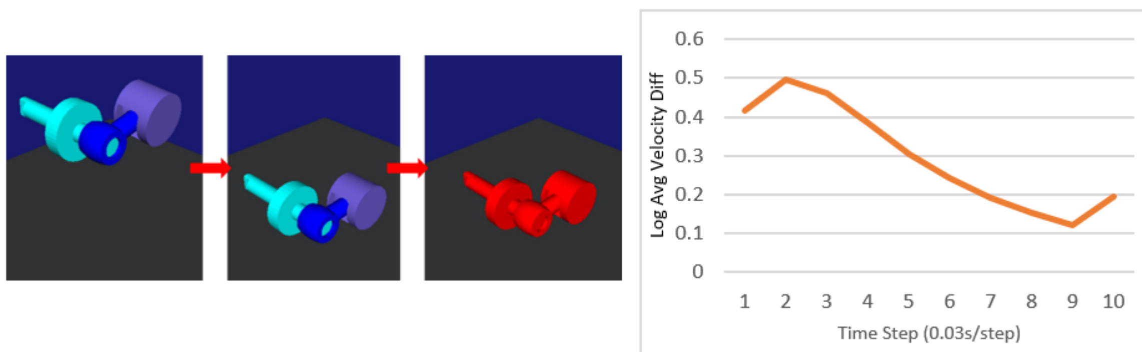
Figure 4.8: The simulation-based stability of the assembly is depended on (a) the actual simulated velocity $\vec{\boldsymbol{v}}_{\boldsymbol{s}}$ of the study part and (b) the internally stable velocity $\vec{\boldsymbol{v}}_{\boldsymbol{is}}$ of the study part that is referenced to the reference part when they are assumed to be attached.

During simulation, simulated velocity $\vec{v}_s$ and internally stable velocity $\vec{v}_{is}$ are used to evaluate the assembly stability and shown in Figure 4.8. $\vec{v}_s$ is the moving part linear velocity which can be captured in each step of the simulation. $\vec{v}_{is}$ is used to predictive the moving part linear velocity while it is assumed to be fixed to the reference part and it is considered as the most internally stable.

There is only one reference part in the assembly and it is randomly selected. Meanwhile, $\vec{v}_{is}$ for each of the study parts in the simulation is calculated by the equation below:

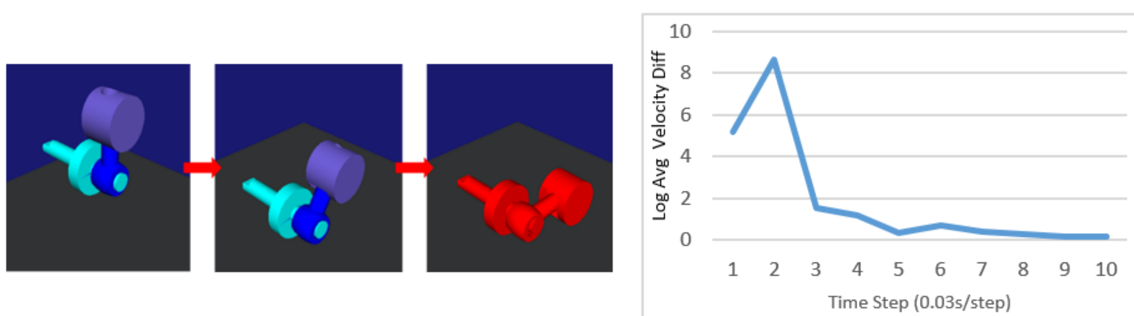
$$\vec{\mathrm{v}}_{\mathrm{is}} = \vec{\mathrm{v}}_{\mathrm{r}} + (\vec{\omega}_{\mathrm{r}} \times \vec{\mathrm{r}}) \tag{16}$$

where $\vec{v}_r$ and $\vec{\omega}_r$ are the linear and angular velocity of the reference part. $\vec{r}$ is the relative center of mass distance from the reference part to the fixed part. After $\vec{v}_{is}$ is calculated, the magnitude difference of velocity between $\vec{v}_s$ and $\vec{v}_{is}$ can be captured in each step of the simulation. Once the assembly is contacted with the table, all of the velocity differences between every moving part and the reference part will be recorded, and the log-transformation is applied to the average differences of the first ten time steps (0.03s per step) data and it is used as the stability score for evaluation.



(a) (b)

Case 1



(c) (d)

Case 2

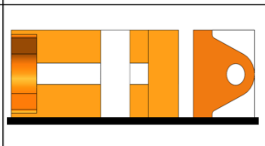
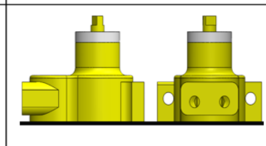
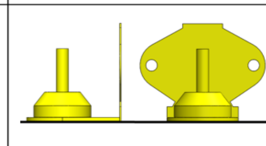
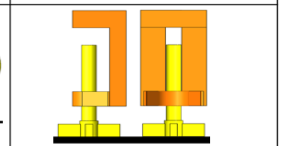
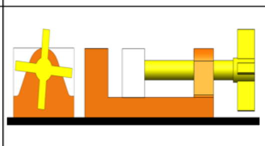
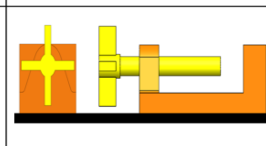
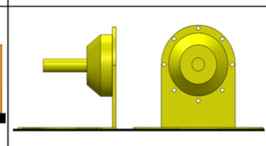
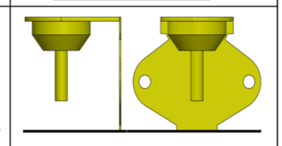
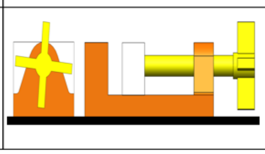
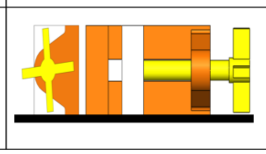
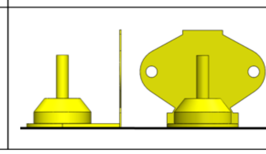
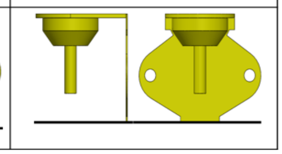
Figure 4.9: Simulation of placing a stacker in two orientations with the Time-Displacement plots

An example simulation of an assembly which contains shaft, crank and piston in two different orientations is shown Figure 4.9. Meanwhile, two plots are used to indicate the log average velocity difference between every two parts in the assembly during simulation. For both simulations, data is being captured once one of the components contacts the table. In the first case, the average velocity difference between parts are not significant since the before and after configurations are similar and this orientation is stable for the assembly being placed on the table. In the second case, the assembly is not stable under the assigned orientation. Once the shaft contacts with the table, the crank starts to rotate about the shaft. Meanwhile, the piston also starts to move along the crank. This indicates a low stability which the maximum average velocity difference is almost equal to 10. Meanwhile, the average displacement is case 2 is higher than which is in case 1.

4.5 RESULTS COMPARISON

The tessellated models of all the assemblies that are used in the survey are also tested within the theory and simulation methods for the same orientations in the survey questions. The top two assemblies that are the most stable and unstable from the three evaluations are shown in Table 4.1 for partial comparison. The theory-based evaluation and the survey share very similar results. For unstable assemblies, they both capture the same assembly with only one different orientation. For stable assemblies, the subassembly with only a base and the push rod of the clamp is captured by the survey. Slightly different results are generated by the physics simulation. In this cast, subassembly of the piston compressor with the shaft and lid is considered as stable, while the vertically placed clamp with no block is the most unstable.

Table 4.1: The most stable and unstable assemblies that are evaluated by the simulation, theory and survey methods.

	Most Stable	Most Stable	Most Unstable	Most Unstable
Simulation				
Survey				
Theory				

Meanwhile, all of the 44 stability scores from the three evaluations are normalized between zero and one for comparison and are then sorted by the order of survey results (see Figure 4.10). Thirty-three simulation-based evaluations and 36 theory-based evaluations are within the 68 percent interval of the survey results. Most of the out-of-interval computational results are on both sides of the plot where the interval is narrower. This is due to people have more consistency while evaluating highly stable and unstably assemblies. Three trend lines (shown with three dotted lines) indicate a relatively close agreement between these three significantly different methods. Pearson correlations between these three sets of data are considered as relatively strong which are all above 0.4. A paired t-test is also used to determine whether the evaluations are significantly different between every two methods when the same assembly model is evaluated. The p-values are shown in Figure 4.10. All the p-values are larger than 0.05 which indicates the means differences are not significant.

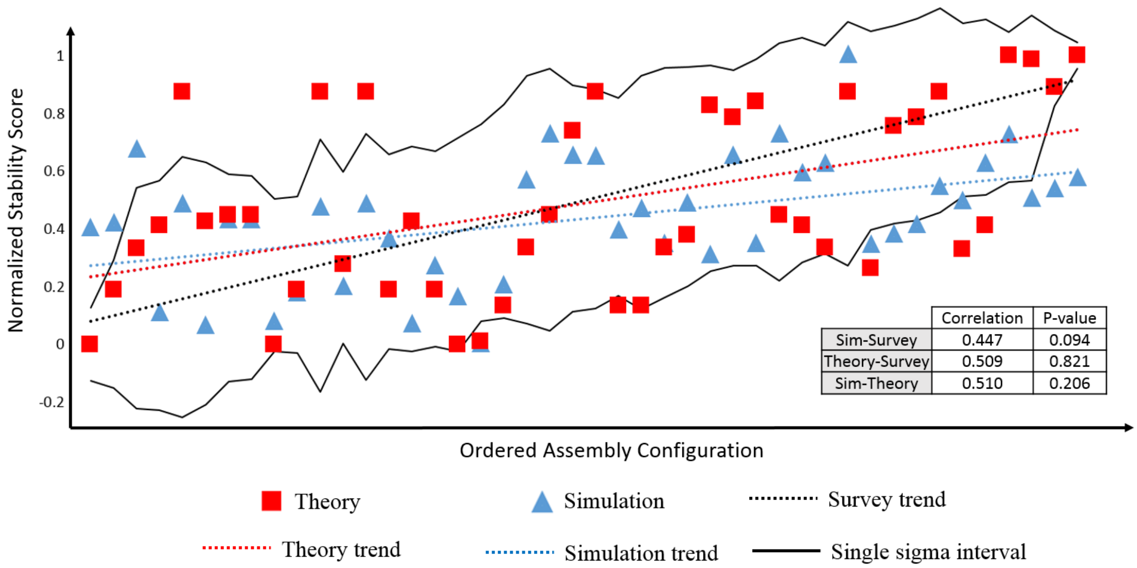


Figure 4.10: All test scores from the theory and simulation-based are compared with the Survey

During this comparison, we also notice some differences between these evaluations methods. For example, the maximum stability score difference between survey and theory-based method is found from the evolution of the subassembly of the piston compressor (Figure 4.11). The theory-based stability score for this evaluation is 0.87 and it is 0.20 in the survey. In other word, it is different between extremely unstable and stable while the two scores are converted back to the Likert scale accordingly.

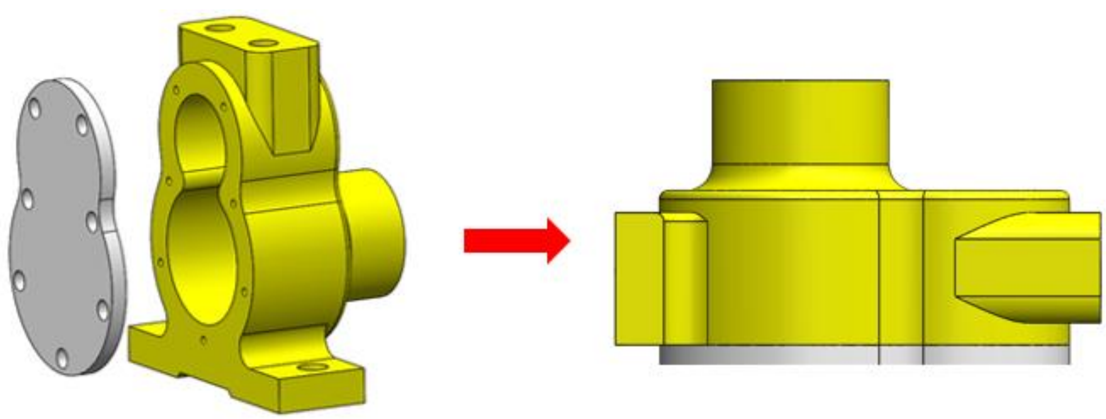


Figure 4.11: Large difference stability result is found when evaluating the orientated piston compressor subassembly in survey and theory-based evaluation

There are two reasons for causing this difference. First, the body and lid in the subassembly have high degree-of-freedom and they dominate the theory-based evaluation. Recalling the combined stability score function (Eq.15), the average degree-of-freedom is equal to 3.5 for this evaluation and it is a very high number compared to other cases in this study. Hence, it is evaluated as very unstable by the theory-based method. Second, the effect of production environment is not considered in this survey so some of the stabilities may be overestimated by the participants. For example, this subassembly is considered as stable because it does not require fixture to keep the two components aligned when it is settled on the assembly station. However, when a batch of these subassemblies need to be transferred between different stations, the stability of this subassembly reduces significantly since the planer connection between this two parts is very sensitive to vibration and acceleration. Hence, to remain alignment between these two parts during transportation, the stability of this subassembly need to be increase. Such simulation is hard to be conducted. Nevertheless, it is important to capture these differences so we can have a better understanding how human evaluation is compared with the proposed computational models.

Overall, compared to the simulation-based method, the theory-based method is the preferred method for evaluating assembly candidates with two major advantages: First, the computational time for using theory-based method is noticeably lower than the simulation-based method. For theory-based method, the average runtime for evaluating an assembly is 0.015 second and it is 1.574 second for simulation-based method. Both methods are run on a computer with Intel i7-6650U 2.20GHz CPU and 8 GB of RAM. This is important for use in search problems such as automated assembly planning since numerous evaluations need to be conducted within the larger planning search process. Second, the setup process for physics simulation is tedious. Many parameters in the physics engine need to be tuned to ensure the high accuracy of the simulation.

Meanwhile, recording and examining the simulation results are difficult because four to five hundred data points could be generated for each simulation. Hence, using theory-based method can reduce the complexity and consequently increase the efficiency.

4.6 CONCLUSION

In this chapter, we developed an automated evaluation method that can evaluate the assembly stability for tessellated CAD models. This evaluation method is crucial for automated assembly sequence planning because it can quickly and accurately estimate the stability of different candidate subassembly designs and their orientations. This allows an automated planner to define plans not simply based on time efficiency, but also on stability. With stability evaluation, an automated planner can design an assemble instruction in which the subassemblies can have higher stability. Meanwhile, the potential cost for utilizing fixtures to support unstable subassemblies during the production can be reduced. To valid the proposed mothed, a survey for evaluating 3D printed assemblies is also conducted to study if there is any significant difference between human perception of stability and the computational method.

The stability evaluation method is divided into theory-based stability and simulation-based stability approaches. The theory-based stability is estimated based on the number of components, part-to-part degree of freedom, tip difficulty and slide difficulty. Automated methods are developed to calculate these factors based on tessellated model geometry. In the simulation-based stability, physics simulations are used to observe the predicted kinematic behavior of the whole assembly system. The motion of placing an assembly is simulated to study the kinematic behavior during assembly tasks. Kinematic constraints are used between every pair of parts to increase the accuracy of the simulation. An average part velocity difference is used as the stability score to describe the stability level during physics simulation.

In the result section, a comparison of the human survey, the theory-based computational method, and the simulation-based methods is presented. Most of the computational predictions that are generated from the proposed methods are within the 68 percent interval of the survey results. Meanwhile, relatively strong correlations are found between these three results which indicates the proposed automated methods are capable to predict assembly stability that is close to human evaluation. The theory-based method is preferred method for evaluating the stability during automated assembly planning. Unlike physics simulation which require a lot of adjustments for different parameters like constraint friction, travel limitation, solver iterations and so on, the theory-based method only take information that is extracted from the tessellated model with zero human interaction. Also, the theory-based method is more suitable and efficient for evaluating large assembly due to the faster runtime.

CHAPTER 5 : COMBINING TIME AND STABILITY ESTIMATION WITH ASSEMBLY PLANNING OPTIMIZATION

5.1 INTRODUCTION

Many factors such as cycle time, subassembly stability, worker performance variation can affect the industrial production process. Ideally, all of these factors need to be considered during the assembly planning process to ensure an optimal plan can be achieved. The implementations of assembly time evaluation and predictive uncertainty from Chapter 3 and assembly stability evaluations from Chapter 4 provide the foundation for introducing the multi-objective optimization to the automated assembly planning process. With this multi-objective optimization, designers can decide the orientation of the final generated assembly, so it can satisfy different production criteria and increase efficiency.

In this chapter, a weighted-sum objective function that includes assembly time, stability and robustness is used for evaluating the cost during the optimization process. The advantage of using the weighted-sum method is that engineers can quickly decide how the finial generated plan is oriented based on different production environments by adjusting the weights between these factors. For example, in high volume production, reduction of assembly time is more of a concern compared to the assembly stability since the labor cost is highly related to the overall production time. Hence, an assembly plan that with good assembly time with relative low stability may be preferred since the stability can be increased by using fixtures, which serve of role for making assembly processes more efficient despite their upfront cost. In the other hand, for low volume or customized production, the cost of designing and using new fixtures may not preferable. Hence, an assembly planning that with high stability and minimum usage of fixtures is more desired. Meanwhile, the multi-objective functions with different assigned weights are apply to the assembly planning optimization and the final results are compared.

5.2 LITERATURE REVIEW

Multi-objective optimization methods has been used in research [74–77] for generating more adaptive assembly plan for various production environments. In Cakir, et al.'s study[74], two objective functions about assembly line workload smoothness and production cost are designed with the constraints of number for center, mean task time, assembly sequence and number of assigned tasks for each center. Meanwhile, to obtain Pareto-optimal solutions, the authors also propose a new solution algorithm which is based on simulated annealing called m_SAA and it show better performance compared to the simulated annealing method. In another study, a weighted-sum objective function is used for applying genetic algorithm in assembly planning problem [75]. This objective function includes five kinds of cost: cycle time, workload smoothness, frequency of tool changes, number of tools and complexity of subassembly. Also, multiple combinations of weights are assigned to these costs for forming different fitness functions and a comparison is conducted to study the impact of the weights on the assembly sequence. A genetic algorithm is also applied in Tiacci's study [76] to design mixed-model un-paced assembly lines with parallel workstations. The summation of the unweighted annual labor cost and equipment cost is used as objective in this research for long term cost minimization. Meanwhile, this direct summation of objectives is also use with genetic algorithm for mixed model assembly line balancing problem in Akpinar and Bayhan's study [77]. The number of workstations, workload smoothness between and within workstations are the three factors that in the objective function.

5.3 DESIGN OF WEIGHTED-SUM OBJECTIVE FUNCTION

When using a weighted-sum objective function, the process of assigning weights to each of the objectives is not straightforward. Especially when the objective scores are from different evaluation systems, it is difficult for designers to assign suitable weights to adapt the various scales. In this study, we develop a weight adjustment method to capture the scales for each of the objectives. This is done for the following objective function:

$$\boldsymbol{f} = \boldsymbol{W_{adj\_time}} * (\boldsymbol{T_{move}} + \boldsymbol{T_{install}}) + \boldsymbol{W_{adj\_stab}} * \boldsymbol{S} + \boldsymbol{W_{SD}} * (\boldsymbol{SD_{move}} + \boldsymbol{SD_{install}}) \quad (17)$$

where:

$$\boldsymbol{W_{adj\_time}} = \boldsymbol{1} - \boldsymbol{W_{stabibliy}} \quad (18)$$

$$\boldsymbol{W_{adj\_stab}} = \boldsymbol{L} * \boldsymbol{W_{stabibliy}} \quad (19)$$

$$\boldsymbol{L} = \overline{\boldsymbol{T}} \ / \ \overline{\boldsymbol{S}} \quad (20)$$

In the objective function (Eq.17) the binary weight $\boldsymbol{W_{SD}}$ with value of one or zero is used for deciding if time prediction with high certainty is preferred during optimization. Meanwhile, $\boldsymbol{W_{adj\_time}}$ and $\boldsymbol{W_{adj\_stab}}$ are defined by the user input weight $\boldsymbol{W_{stabibliy}}$ and its value is between zero and one. When $\boldsymbol{W_{stabibliy}}$ is equal to zero, it only considers assembly time in the optimization process. When it is equal to one, the assembly plan is optimized based only on stability. To adapt the different scales of assembly time and stability, a random tree search method is applied to estimate the assembly time and stability in different levels of the tree so the average assembly time

$\overline{\boldsymbol{T}}$ and average subassembly stability $\overline{\boldsymbol{S}}$ can be used for approximating the scale ratio $\boldsymbol{L}$ for weight adjustment.

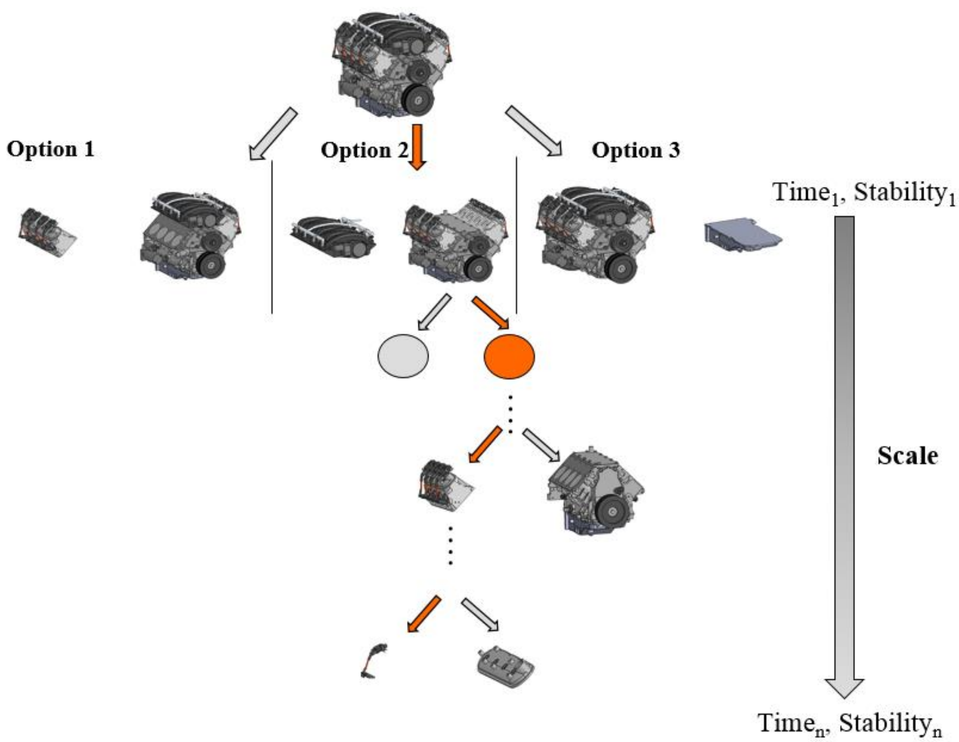


Figure 5.1: Applying random tree search to obtain assembly time and stability with different scales for objective function weight adjustment

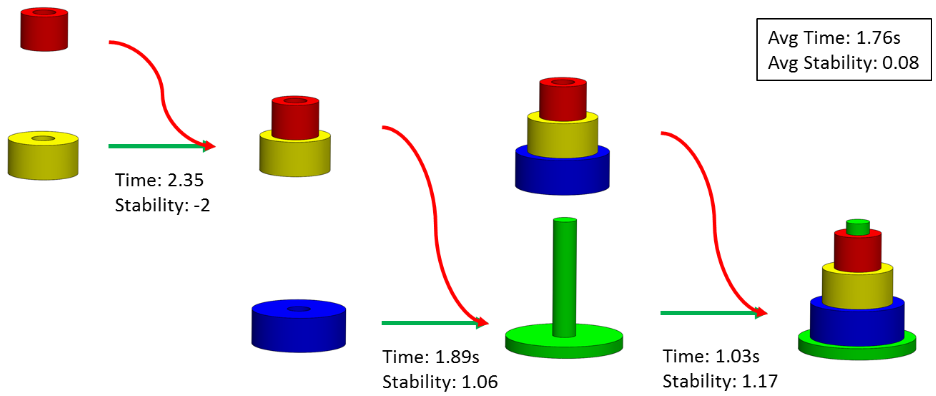
An example of how the random tree search for weight adjustment functions is shown in Figure 5.1. In the beginning of the tree where multiple assembly tasks are generated, one of these options is randomly selected for populating the consecutive assembly tasks. Once the options are produced in the next level, another random selection is conducted. This process will stop once it reaches the bottom of the tree when no more options are generated. During this process, assembly time decreases along the tree branch since the difficulty of the assembly tasks decreases with the reduced assembly complexity, volume and weight. Meanwhile, the scale ratio for weight adjustment can be obtained based on the average assembly time and stability from the visited assembly tasks in this random tree search.

5.4 RESULTS

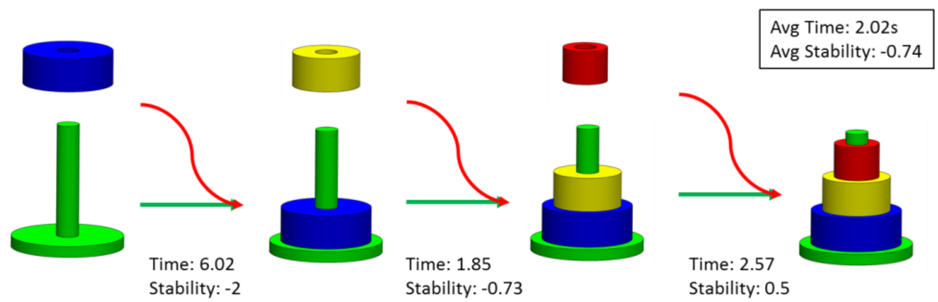
In this section, the tree search method is applied to three assemblies: stacker toy, oil pump and a crank-slider mechanism to generate the assembly sequences with different assigned weights in the objective function. For each of these assemblies, two assembly sequences are generated with stability weights set to zero and one accordingly. When the stability weight is set to zero, the generated assembly sequence will be assembly time-oriented; when it is equal to one, subassemblies that have high stability will be preferred in the assembly sequence. By comparing these two sequences, the effect of different weight settings in the objective function can be visualized. On the other hand, the comparison about uncertainty weight setting is not included here for now since this comparison is not straightforward. Hence, we only focus on the tradeoff between assembly time and stability in the assembly sequence in this section.

5.4.1 Assembly Sequences of Stacker Toy

Two assembly sequences for the stacker toy is shown in Figure 5.2. For time-oriented assembly sequence (Figure 5.2a), the stacker blocks are preferred to be assembled together first in this process. After that, the subassembly of blocks is inserted into the base. The average assembly time for this sequence is 1.76s. When more weight is assigned to the stability, each of the stacker blocks is installed to the base in order (Figure 5.2b). Although the average assembly time is higher than the previous case, the stability of this solution increases significantly since each of these stackers is connected with the base by the cylinder joint and the tip-slide action is limited. Thus, the average stability for each of the assembly tasks is -0.74 which indicate this sequence is more stable than the previous one.



(a): Time-oriented assemmly Sequence

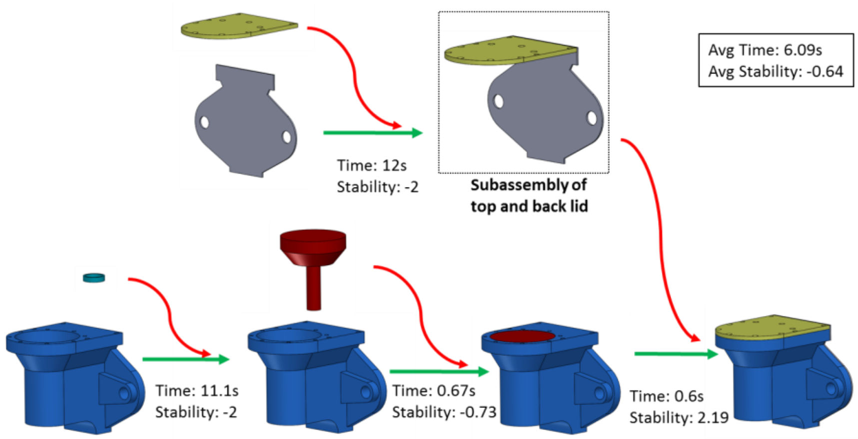


(b): Stability-oriented assemmly Sequence

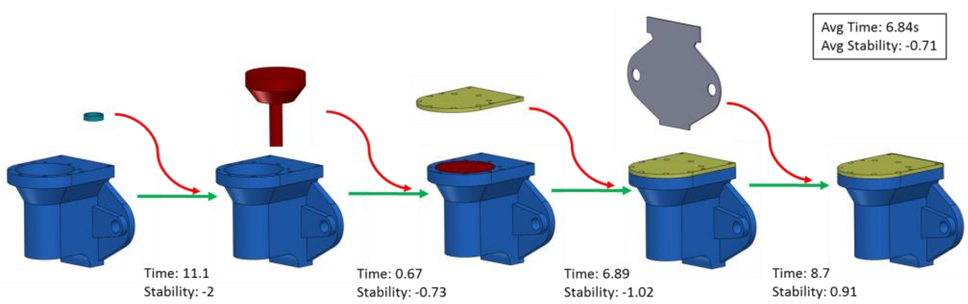
Figure 5.2: Time-oriented (a) and stability-oriented (b) assemmly sequences for stacker toy

5.4.2 Assembly Sequences of Oil Pump

Two assembly sequences (Figure 5.3a and Figure 5.3b) for the oil pump model are also generated based on different stability weights in the objective function. The major difference between these two sequences is for time-oriented sequence, the subassembly of the top lid and back lid is generated, and this subassembly is installed to the pump body; for time-oriented sequence, the top lid and back lid are installed to the pump body individually in two separated tasks. The reason for causing this difference is when installing the subassembly of top lid and back lid, this subassembly is easier to be handled since it has bigger oriented bounding box. Thus, it is preferred in creating faster assembly times. However, the internal stability of this subassembly is not good since these two lids are connected by the planer joint and the tip-slide motions are not constrained, and the estimated stability score for this task is 2.19 which is the highest compared to the others in this sequence. Hence, these two lids are installed separately in the stability-oriented sequence.



(a): Time-oriented assemmly Sequence

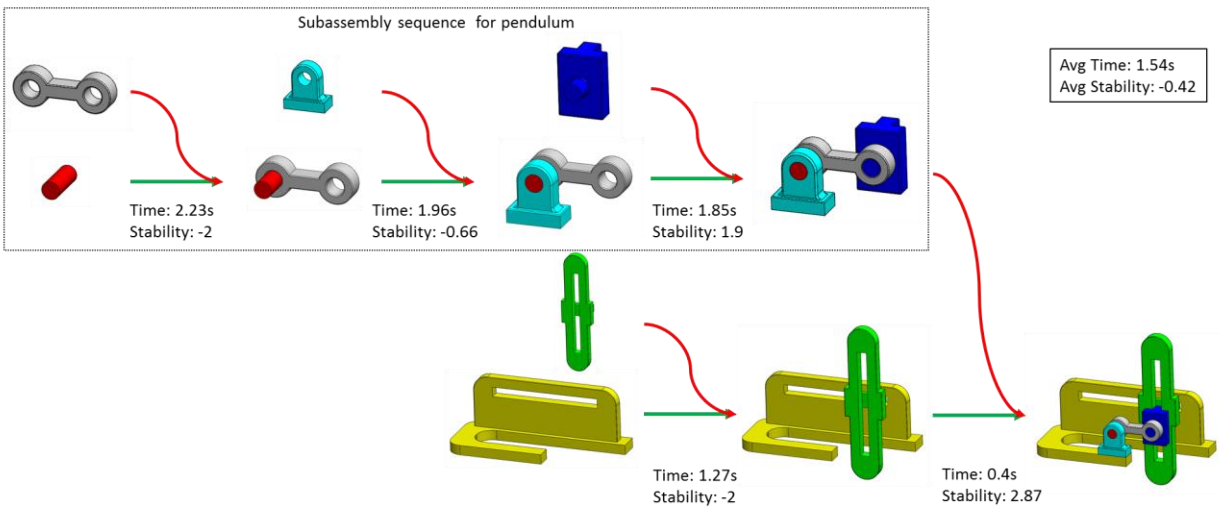


(b): Stability-oriented assemmly Sequence

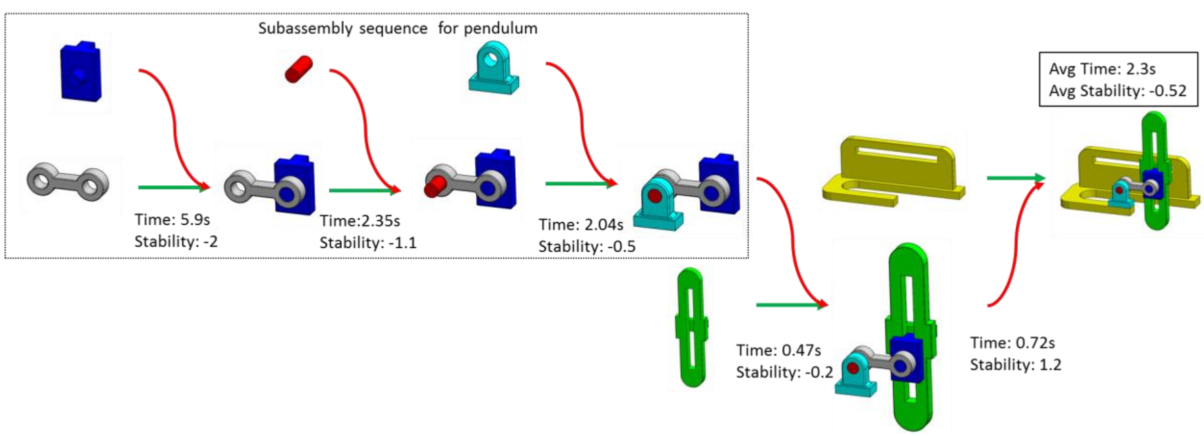
Figure 5.3: Time-oriented (a) and stability-oriented (b) assemmly sequences for oil pump

5.4.3 Assembly Sequences of Crank-Slider Mechanism

The final test case is a crank-slider linkage mechanism shown in Figure 5.4. Compared to the two previous assemblies, this mechanism model has the most complicated structure since it includes cylinder, planar and prismatic constraints. Meanwhile, the assembly sequences in both cases are not alike even though the pendulum subassembly is generated. For time-oriented sequence, the average assembly time for each assembly task is 1.54s and it is 2.3s for stability-oriented sequence. The major difference between these two sequences is the process for assembling the pendulum sub-model. In the time-oriented sequence, the rod has a high degree of freedom until the very end of this subsequence when the blue slider is installed to it and limits its rotation movement. For stability-oriented sequence, the rotational movement of the rod is restrained by the red pin and blue slider in the beginning, and it results in a more stable sequence.



(a): Time-oriented assemmly Sequence



(b): Stability-oriented assemmly Sequence

Figure 5.4: Time-oriented (a) and stability-oriented (b) assemmly sequences for crank-slider linkage mechanism

CHAPTER 6 : CONCLUSIONS

Designing an assembly plan is one of the most important steps for manufacturing. Especially when up to 85 percent of the production cost can be committed in the early design stage, design tools that can produce accurate assembly cost estimates are critical. With precise cost estimation, optimization can be introduced to achieve assembly cost reduction and high productivity. Many evaluation tools have been developed in the past decades for evaluating different assembly costs such as assembly time, stability, worker motion complexity and so on. However, to apply these tools in production planning, designers need to provide lots of detailed information about the product design, worker performance, factory layout, etc. With increased complexity of modern products and higher demand of design automation with CAD models, such large amount of human involvement in assembly planning is not desirable. Hence, in this PhD work, we introduced an automated cost estimation tool to estimate the assembly time and stability based on CAD models. The benefit of this tool is that it can be easily embedded within an automated assembly planning (APP) process. In this case, users only need to provide the CAD assembly model and this design tool is able predict different assembly costs and guide the formation of assembly plan based on designer's preferences.

Tessellated CAD models are used in this research due to its independence from any commercial software and commonly used in industry. However, the tessellation format contains limited information, which challenge the development of accurate models for assembly time and stability estimation. Thus, assembly experiments were conducted for data gathering and the collected data was used for model development. This study includes two experiments for assembly time and stability survey. In the assembly time experiment, we introduced four assembly actions: move, install, secure and rotate, to describe the required motions that for accomplishing assembly tasks in the production environment. At the same time, prediction model inputs that can be automatically obtained from tessellated models were defined and measured in the experiment. In addition, workbench and modular assembly stations were used to simulate the production environment in different scales. For the assembly stability survey, external and internal stability were introduced for describing the stability of an assembly during different assembly process. Also, survey questions for evaluating these two kinds of stability by using 3D printed assemblies and their images were designed for understanding the human perception of stability.

With the collected assembly time data, we applied one of the machine learning methods, Gaussian Process (GP), to generate the time prediction models to estimate the four proposed actions times. While comparing with the regression and artificial neural network models, the developed GP model shows similar accuracy with better performance of capturing the data variation. Additionally, this GP model is compared to the widely used design for assembly (DFA) time estimation method and it produced very similar results in the case study problem, which indicated this fully automated method is capable to be applied to the assembly planning process for generating highly accurate assembly time prediction.

For assembly stability, we proposed two different approaches: theory-based stability evaluation and systematic stability evaluation. To conduct theory-based stability evaluation, we implemented a method to calculate the degree of freedom, tip difficulty and slide difficulty scores of the components in a tessellated assembly model. Meanwhile, these scores are combined in an evaluation function which is applied in the assembly planning search process for stability estimation. In addition, a physics engine is used to simulate the assembly process. During simulation, kinematic behaviors for each of the components can be captured and used to evaluate the systematic stability from a dynamics standpoint. The computational results from these two proposed methods are compared with the stability survey data for validation.

Finally, a weighted-sum objective function approach that includes estimates for assembly time and stability, while considering model prediction uncertainty was designed and applied in the assembly planning search process. Meanwhile, three assemblies: stacker toy, oil pump and a crank-slider mechanism are used for generating time-oriented and stability-oriented assembly plans to demonstrate the effect of different objective weight settings. By adjusting the objective weights in the objection function, various assembly plans can be generated to satisfy different production requirements of assembly time, stability.

6.1 CONTRIBUTIONS

- A fully automated approach to predict assembly time and stability from only tessellated 3D models is developed.

- A machine learning method for assembly time estimation is developed that is accurate and can model uncertainty.

- Automated kinematic constraint generation from tessellated 3D models is developed for use in physics simulation.
- An efficient method to evaluate assembly stability using only CAD models has been created.
- The first study in understanding how humans perceive stability is performed in the course of this research.
- A multi-objective optimization scheme is introduced for creating more adaptive assembly plans.

6.2 FUTURE WORK

In this PhD work, an automated method to evaluate the assembly time and stability for assembly planning problem based on tessellated 3D CAD model was developed. Future extensions of this work could be done to further improve this research. First, conducting more assembly experiments with assemblies in larger scale would make the models more accurate for use in large system manufacturing. The current training data for developing the time prediction models were collected from small scale assembly tasks. With assembly experiment of larger assemblies, the application of this time estimation can be extended. Second, the effect of human factors on assembly planning can also be introduced. Including human factors in automated assembly planning is a challenge since it involves various information factory layout, work station design, human motions, etc. However, embedding such information into the evaluation process can increase the evaluation accuracy.

BIBLIOGRAPHY

[1] Jiao, J., and Tseng, M. M., 1999, “A Pragmatic Approach to Product Costing Based on Standard Time Estimation,” International Journal of Operations & Production Management, 19(7), pp. 738–755.

[2] National Research Council (U.S.). Committee on Engineering Design Theory and Methodology., 1991, *Improving Engineering Design : Designing for Competitive Advantage*, National Academy Press.

[3] Wang, L., Keshavarzmanesh, S., Feng, H. Y., and Buchal, R. O., 2009, “Assembly Process Planning and Its Future in Collaborative Manufacturing: A Review,” International Journal of Advanced Manufacturing Technology, 41(1-2), pp. 132–144.

[4] Maynard, H. B., Stegemerten, G. J., and Schwab, J. L., 1948, *Methods-Time Measurement*, McGraw-Hill Book Co.

[5] B.zandin, K., 2003, *MOST Work Measurement System*, Marcel Dekker.

[6] Singer, G., Golan, M., and Cohen, Y., 2014, “From Product Documentation to a ‘method Prototype’ and Standard Times: A New Technique for Complex Manual Assembly,” International Journal of Production Research, 52(2), pp. 507–520.

[7] Jiao, J., and Tseng, M. M., “Customizability Analysis in Design for Mass Customization.”

[8] Tuan, S. T., Puchong, J., Alam, S., Lumpur, K., Karim, a N. M., Kays, H. M. E., Amin, a K. M. N., and Hasan, M. H., 2014, “Improvement of Workflow and Productivity through Application of Maynard Operation Sequence Technique (MOST),” *International Conference on Industrial Engineering and Operations Management*, pp. 2162–2171.

[9] Miyakawa, S., and Ohashi, T., 1986, “The Hitachi Assembly Evaluation Method (AEM),” *International Conference on Product Design for Assembly*, Newport, Rhode Island.

[10] Boothroyd, G., Dewhurst, P., and Knight, W., 1994, *Product Design for Manufacture and Assembly*, Marcel Dekker, New York.

[11] “DFMA Software” [Online]. Available: http://www.dfma.com/software/dfma.asp. [Accessed: 30-Nov-2017].

[12] Mathieson, J. L., Wallace, B. A., and Summers, J. D., 2010, “Assembly Time Modeling through Connective Complexity Metrics,” Proceedings - 2010 International Conference on Manufacturing Automation, ICMA 2010, pp. 16–23.

[13] Owensby, J. E., and Summers, J. D., 2014, “Assembly Time Estimation: Assembly Mate Based Structural Complexity Metric Predictive Modeling,” Journal of Computing and Information Science in Engineering, 14(1), p. 011004.

[14] Ou, L.-M., and Xu, X., 2013, “Relationship Matrix Based Automatic Assembly Sequence Generation from a CAD Model,” Computer-Aided Design, 45(7), pp. 1053–1067.

[15] Alkan, B., Vera, D., Ahmad, M., Ahmad, B., and Harrison, R., 2016, “A Model for Complexity Assessment in Manual Assembly Operations Through Predetermined Motion Time Systems,” Procedia CIRP, 44, pp. 429–434.

[16] Cho, H., Lee, S., and Park, J., 2014, “Time Estimation Method for Manual Assembly Using MODAPTS Technique in the Product Design Stage,” International Journal of Production Research, 52(12), pp. 3595–3613.

[17] Lee, S., and Shin, Y. G., “Assembly Planning Based on Subassembly Extraction,” *Proceedings., IEEE International Conference on Robotics and Automation*, IEEE Comput. Soc. Press, pp. 1606–1611.

[18] Wang, Y., Liu, J. H., and Li, L. S., 2009, “Assembly Sequences Merging Based on Assembly Unit Partitioning,” The International Journal of Advanced Manufacturing Technology, 45(7-8), pp. 808–820.

[19] Dong, T., Tong, R., Zhang, L., and Dong, J., 2007, “A Knowledge-Based Approach to Assembly Sequence Planning,” The International Journal of Advanced Manufacturing Technology, 32(11-12), pp. 1232–1244.

[20] Aleotti, J., and Caselli, S., 2011, “Physics-Based Virtual Reality for Task Learning and Intelligent Disassembly Planning,” Virtual Reality, 15(1), pp. 41–54.

[21] “Man Sitting Topview Icon Stock Illustration.” [Online]. Available: https://www.dreamstime.com/stock-illustration-man-sitting-topview-icon-flat-design-illustration-image74204156. [Accessed: 02-Dec-2017].

[22] “Workers Stock Images” [Online]. Available: https://www.shutterstock.com/search/workers?searchterm=workers&language=en&page=2. [Accessed: 02-Dec-2017].

[23] “Tools and Engine” [Online]. Available: https://www.cartechbooks.com/techtips/how-to-build-racing-engines-engine-build-tips/.

[24] Gottschalk, S., 2000, "Collision Queries Using Oriented Bounding Box," University of North Carolina at Chapel Hill.

[25] "TVGL: Tessellation and Voxelization Geometry Library by Design Engineering Lab" [Online]. Available: http://designengrlab.github.io/TVGL/. [Accessed: 16-Feb-2017].

[26] "MIConvexHull" [Online]. Available: https://miconvexhull.codeplex.com/. [Accessed: 16-Feb-2016].

[27] Rafibakhsh, N., and Campbell, M. I., 2015, "Hierarchical Primitive Surface Classification From Triangulated Solids for Defining Part-to-Part Degrees of Freedom," *Volume 2B: 41st Design Automation Conference*, ASME, p. V02BT03A016.

[28] Rafibakhsh, N., Huang, W., and Campbell, M. I., 2017, "Automatic Detection of Fasteners From Tessellated Mechanical Assembly Models," Journal of Computing and Information Science in Engineering, 18(1), p. 011005.

[29] Liao, X., Li, Q., Yang, X., Zhang, W., and Li, W., 2007, "Multiobjective Optimization for Crash Safety Design of Vehicles Using Stepwise Regression Model," Structural and Multidisciplinary Optimization, 35(6), pp. 561–569.

[30] Puertas Arbizu, I., and Luis Pérez, C. J., 2003, "Surface Roughness Prediction by Factorial Design of Experiments in Turning Processes," Journal of Materials Processing Technology, 143-144, pp. 390–396.

[31] Wu, L., Yick, K., Ng, S., and Yip, J., 2012, "Application of the Box–Behnken Design to the Optimization of Process Parameters in Foam Cup Molding," Expert Systems with Applications.

[32] Liu, Y., Huang, W., Rafibakhsh, N., Campbell, M. I., and Hoyle, C., 2016, "Design of Experiments to Support Automated Assembly Planning," *Volume 11: Systems, Design, and Complexity*, ASME, p. V011T15A031.

[33] Likert, R., 1932, "A Technique for the Measurement of Attitudes," Archives of Psychology.

[34] Boothroyd, G. (Geoffrey), Dewhurst, P., and Knight, W. A. (Winston A., 2011, *Product Design for Manufacture and Assembly*, CRC Press.

[35] Lucas Engineering Systems Ltd, and University Of Hull, 1993, "Design For Manufacture and Assembly Practitioners Manual."

[36] Wang, L., Keshavarzmanesh, S., Feng, H.-Y., and Buchal, R. O., 2009, "Assembly Process Planning and Its Future in Collaborative Manufacturing: A Review," The International Journal of Advanced Manufacturing Technology, 41(1-2), pp. 132–144.

[37] Miller, M. G., Mathieson, J. L., and Summers, J. D., 2012, "Representation: Structural Complexity of Assemblies To Create Neural Network Based Assembly Time Estimation Models," Proceedings of the ASME 2012 International Design Engineering Technical Conferences &

Computers and Information in Engineering Conference, pp. 1–11.

[38] Chang, G. A., 2002, "A Neural Network Model for the Handling Time of Design for Assembly," Journal of the Chinese Institute of Industrial Engineers, 19(1), pp. 35–48.

[39] Koganti, R., Zaluzec, M., Chen, M., and Defersha, F. M., 2006, "Design for Integrated Assembly and Disassembly of Automotive Products."

[40] Wang, H., Rong, Y., and Xiang, D., 2014, "Mechanical Assembly Planning Using Ant Colony Optimization," Computer-Aided Design, 47, pp. 59–71.

[41] Thomas, U., and Wahl, F., 2001, "A System for Automatic Planning, Evaluation and Execution of Assembly Sequences for Industrial Robots," *IEEE International Conference on Intelligent Robots and Systems, Maui, Hawai, USA*, pp. 1458–1464.

[42] Argyrou, A., Giannoulis, C., Papakostas, N., and Chryssolouris, G., 2016, "A Uniform Data Model for Representing Symbiotic Assembly Stations," Procedia CIRP, 44, pp. 85–90.

[43] Wang, J. M., Fleet, D. J., and Hertzmann, A., 2008, "Gaussian Process Dynamical Models for Human Motion," IEEE Transactions on Pattern Analysis and Machine Intelligence, 30(2), pp. 283–298.

[44] Mark Gibbs, D. J. C. M., "Efficient Implementation of Gaussian Processes."

[45] MacKay, D., 1992, "Bayesian Interpolation," Neural computation.

[46] Chen, W., Hoyle, C., and Wassenaar, H. J., 2013, *Decision-Based Design : Integrating Consumer Preferences into Engineering Design*, Springer.

[47] Rasmussen, C. E., and Williams, C. K. I., 2006, *Gaussian Processes for Machine Learning*, University Press Group Limited.

[48] Ripley, B. D., 1996, *Pattern Recognition and Neural Networks*, Cambridge University Press.

[49] Stathakis, D., "How Many Hidden Layers and Nodes?"

[50] Hecht-Nielsen, R., 1987, "Kolmogorov's Mapping Neural Network Existence Theorem," *Proceedings of the IEEE First International Conference on Neural Networks , San Diego, CA, USA*, pp. 11–13.

[51] MathWorks, 2010, "Neural Network Toolbox - MATLAB."

[52] Burden, F., and Winkler, D., 2008, "Bayesian Regularization of Neural Networks," pp. 23–42.

[53] Namouz, E., 2013, "Automated Complexity Based Assembly Time Estimation Method."

[54] Mattikalli, R., Baraff, D., Khosla, P., and Repetto, B., 1995, "Gravitational Stability of Frictionless Assemblies," IEEE Transactions on Robotics and Automation, 11(3), pp. 374–388.

[55] Mattikalli, R., Baraff, D., and Khosla, P., "Finding All Gravitationally Stable Orientations of Assemblies," *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, IEEE Comput. Soc. Press, pp. 251–257.

[56] Rakshit, S., and Akella, S., 2015, "The Influence of Motion Paths and Assembly Sequences on the Stability of Assemblies," IEEE Transactions on Automation Science and Engineering, 12(2), pp. 615–627.

[57] STEWART, D. E., and TRINKLE, J. C., 1996, "AN IMPLICIT TIME-STEPPING SCHEME FOR RIGID BODY DYNAMICS WITH INELASTIC COLLISIONS AND COULOMB FRICTION," International Journal for Numerical Methods in Engineering, 39(15), pp. 2673–2691.

[58] Baraff, D., and David, 1994, "Fast Contact Force Computation for Nonpenetrating Rigid Bodies," *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH '94*, ACM Press, New York, New York, USA, pp. 23–34.

[59] Wang, Y., and Liu, J., 2013, "Subassembly Identification for Assembly Sequence Planning," The International Journal of Advanced Manufacturing Technology, 68(1-4), pp. 781–793.

[60] Laperrière, L., and ElMaraghy, H. A., 1996, "GAPP: A Generative Assembly Process Planner," Journal of Manufacturing Systems, 15(4), pp. 282–293.

[61] Laperrière, L., and ElMaraghy, H. A., 1994, "Assembly Sequences Planning for Simultaneous Engineering Applications," The International Journal of Advanced Manufacturing Technology, 9(4), pp. 231–244.

[62] Blum, M., Griffith, A., and Neumann, B., 1970, "A Stability Test for Configurations of Blocks."

[63] Bernheisel, J. D., and Lynch, K. M., 2004, "Stable Transport of Assemblies: Pushing Stacked Parts," IEEE Transactions on Automation Science and Engineering, 1(2), pp. 163–168.

[64] Bernheisel, J. D., and Lynch, K. M., "Stable Pushing of Assemblies," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, IEEE, pp. 3280–3287.

[65] Jae-Sook Cheong, Goldberg, K., Overmars, M. H., and van der Stappen, A. F., "Fixturing Hinged Polygons," *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, IEEE, pp. 876–881.

[66] Sankar Jayaram, Uma Jayaram, Yong Wang, Tirumali, H., Lyons, K., and Hart, P., 1999, "VADE: A Virtual Assembly Design Environment," IEEE Computer Graphics and Applications, 19(6), pp. 44–50.

[67] “Bullet Physics Library” [Online]. Available: http://bulletphysics.org/wordpress/. [Accessed: 09-Feb-2017].

[68] “PhysX | GeForce.”

[69] “Open Dynamics Engine.”

[70] Gonzalez-Badillo, G., Medellin-Castillo, H. I., and Lim, T., 2013, “Development of a Haptic Virtual Reality System for Assembly Planning and Evaluation,” Procedia Technology, 7, pp. 265–272.

[71] Read, A., Ritchie, J., and Lim, T., 2016, “A UNITY Sketch Based Modelling Environment for Virtual Assembly and Machining to Evaluate DFMA Metrics,” *Volume 1B: 36th Computers and Information in Engineering Conference*, ASME, p. V01BT02A049.

[72] Guan, Q., Liu, J. H., and Zhong, Y. F., 2002, “A Concurrent Hierarchical Evolution Approach to Assembly Process Planning,” International Journal of Production Research, 40(14), pp. 3357–3374.

[73] Rafibakhsh, N., and Campbell, M. I., 2015, “Hierarchical Primitive Surface Classification From Triangulated Solids for Defining Part-to-Part Degrees of Freedom,” *Volume 2B: 41st Design Automation Conference*, ASME, p. V02BT03A016.

[74] 2011, “Multi-Objective Optimization of a Stochastic Assembly Line Balancing: A Hybrid Simulated Annealing Algorithm,” Computers & Industrial Engineering, 60(3), pp. 376–384.

[75] Chen, R.-S., Lu, K.-Y., and Yu, S.-C., 2002, “A Hybrid Genetic Algorithm Approach on Multi-Objective of Assembly Planning Problem,” Engineering Applications of Artificial Intelligence, 15(5), pp. 447–457.

[76] Tiacci, L., 2015, “Coupling a Genetic Algorithm Approach and a Discrete Event Simulator to Design Mixed-Model Un-Paced Assembly Lines with Parallel Workstations and Stochastic Task Times,” International Journal of Production Economics, 159, pp. 319–333.

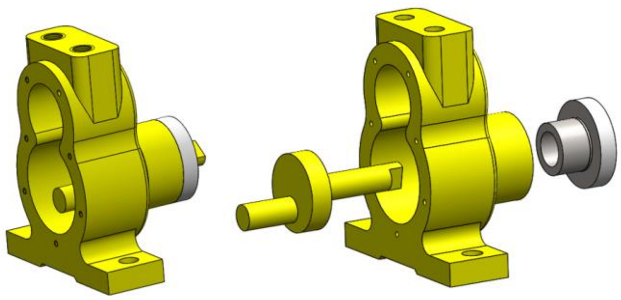
[77] 2011, “A Hybrid Genetic Algorithm for Mixed Model Assembly Line Balancing Problem with Parallel Workstations and Zoning Constraints,” Engineering Applications of Artificial Intelligence, 24(3), pp. 449–457.

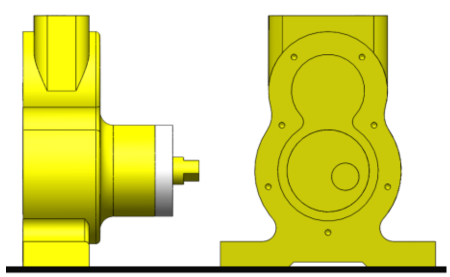
APPENDICES

APPENDIX A: HUMAN SUBJECT SURVEY OF ASSEMBLY STABILITY

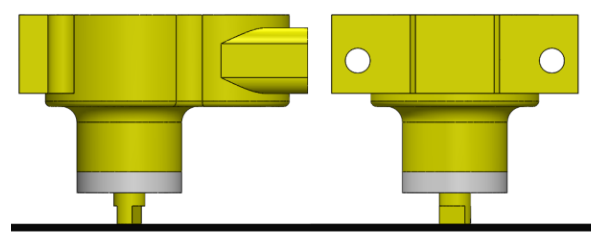
EU: Extremely Unstable **U:** Unstable **WU:** Weakly Unstable **NE:** Neutral
WS: Weakly Stable **S:** Stable **ES:** Extremely Stable

Assembly 1-1 and its explosive views

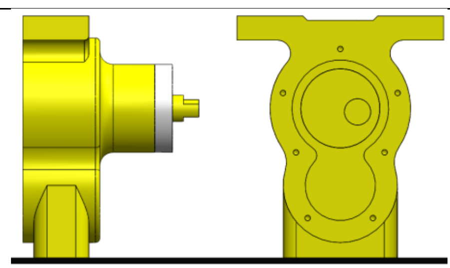




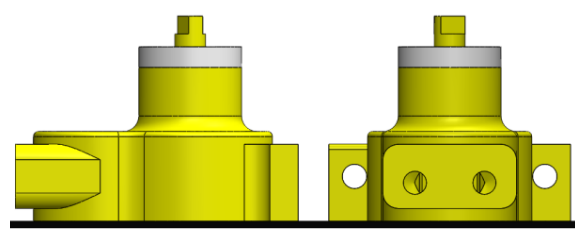
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

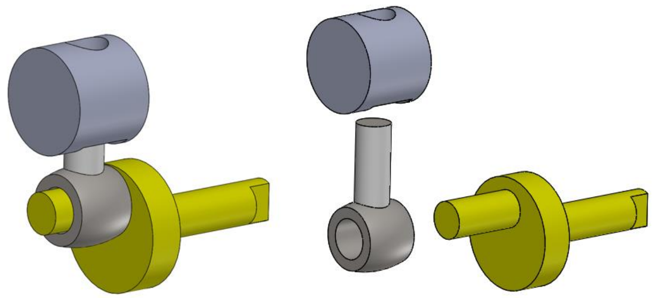


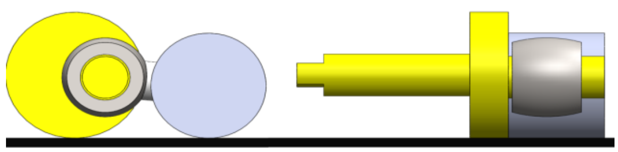
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

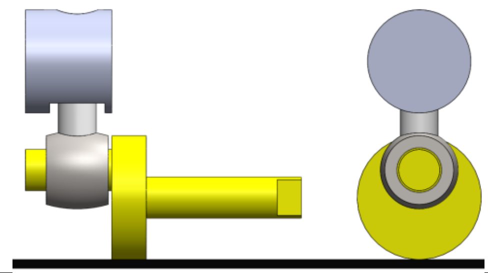
Assembly 1-2 and its explosive views





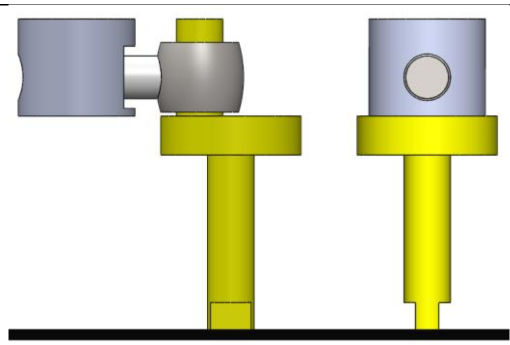
Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐



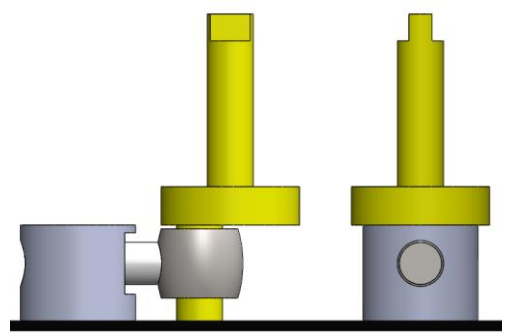
Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐



Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

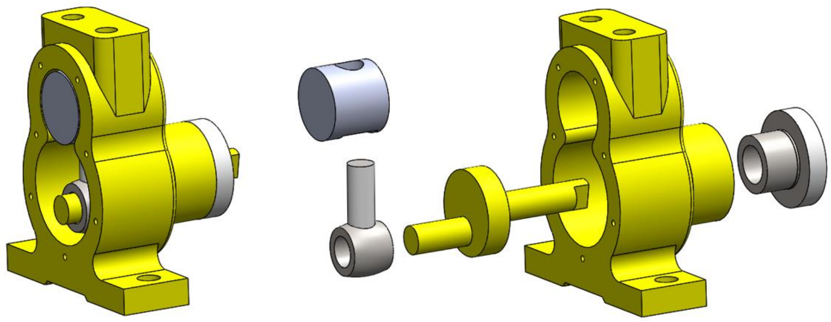
External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

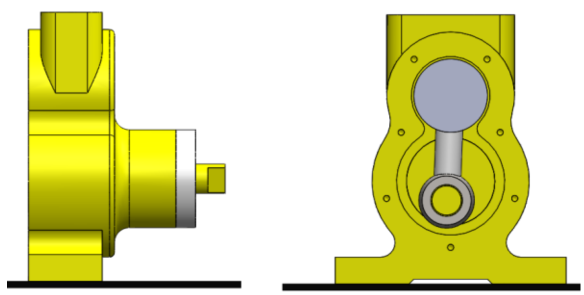


Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

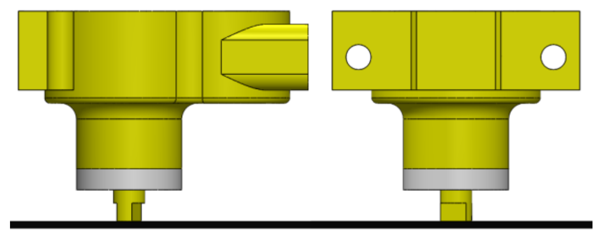
External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

Assembly 1-3 and its explosive views

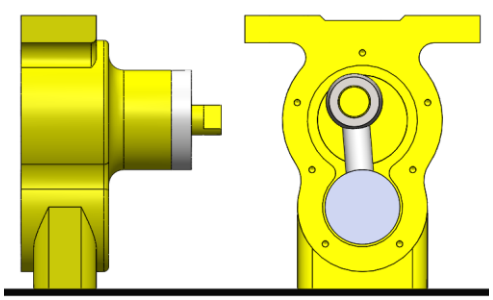




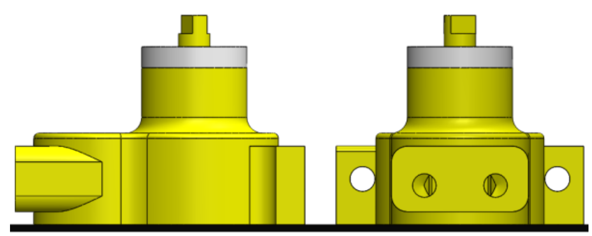
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

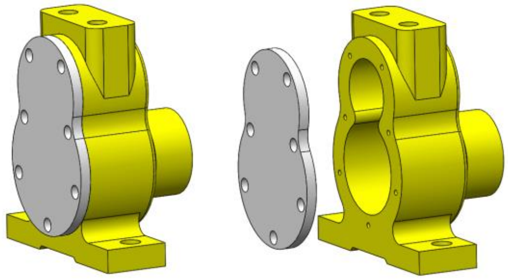


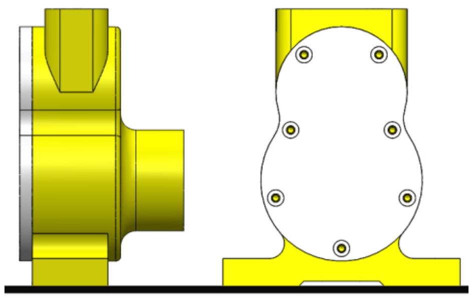
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



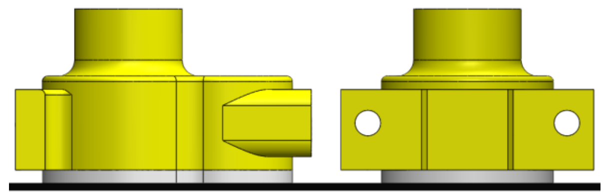
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

Assembly 1-4 and its explosive views

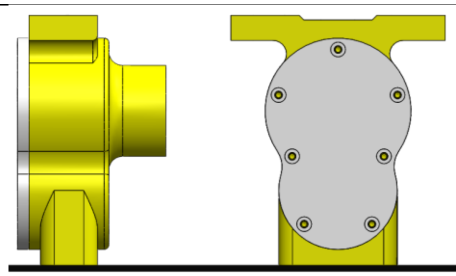




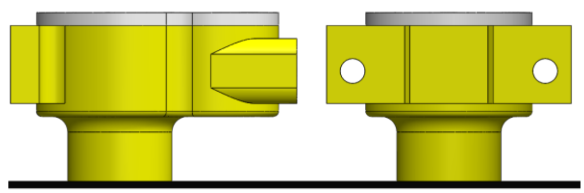
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

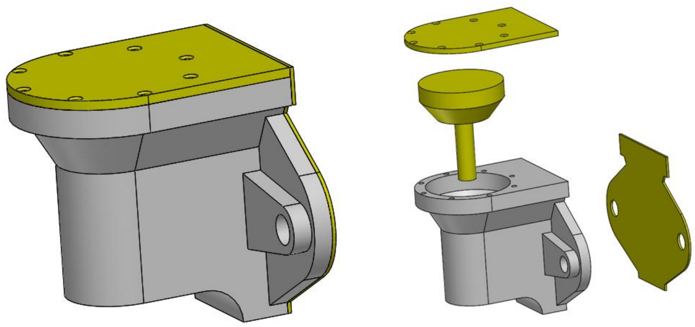


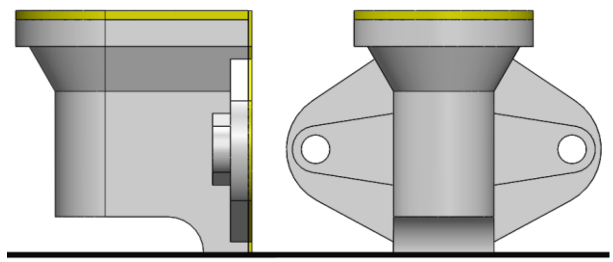
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



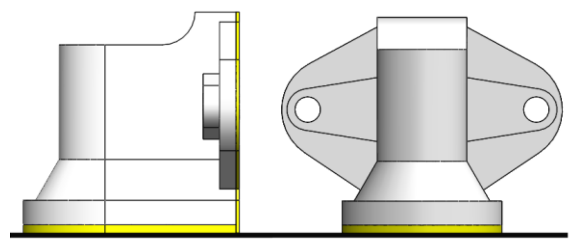
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

Assembly 2-1 and its explosive views

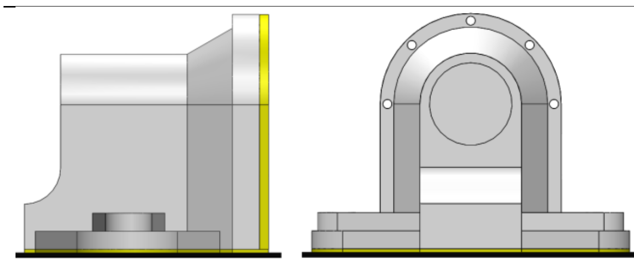




	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

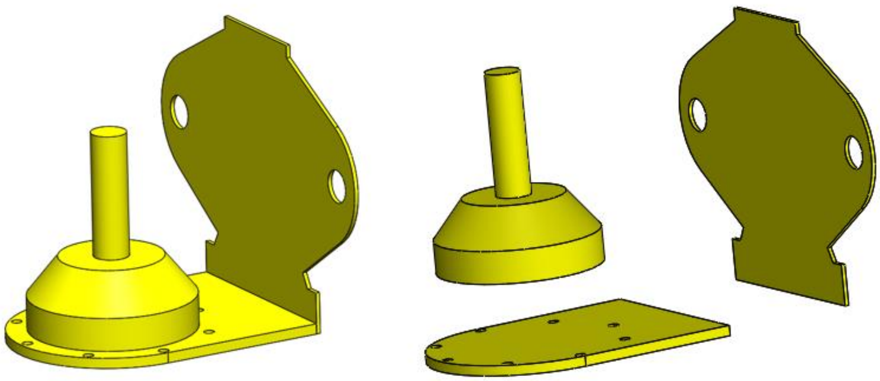


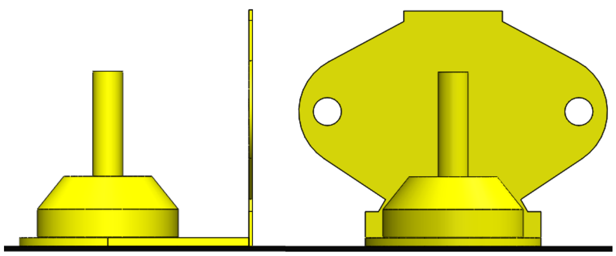
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



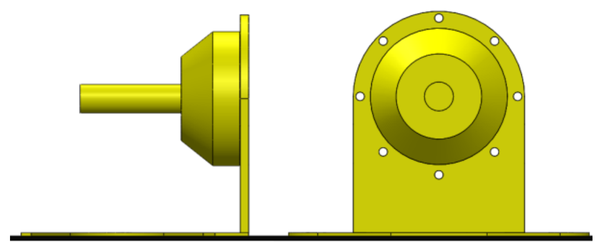
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

Assembly 2-2 and its explosive views

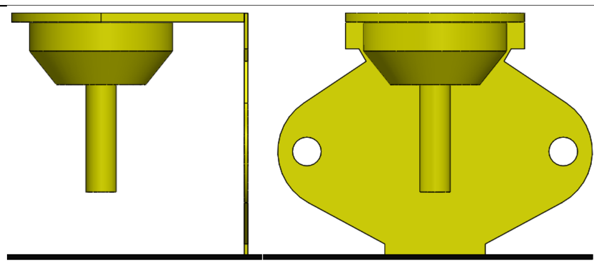




	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

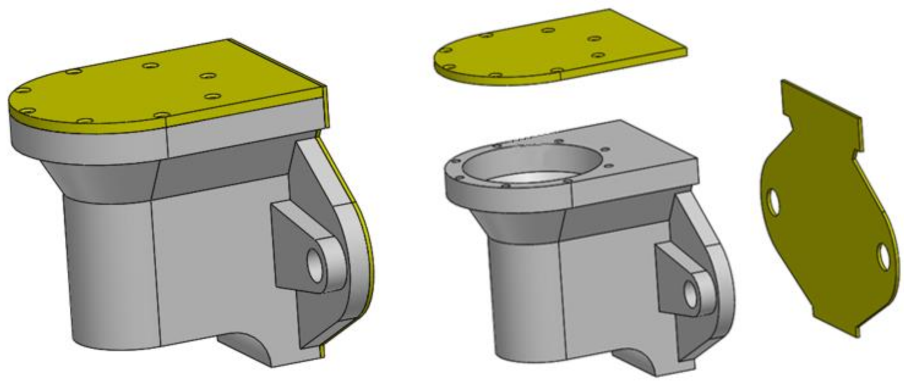


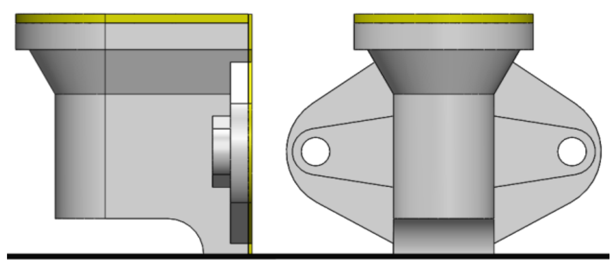
	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐



	EU	U	WU	NE	WS	S	ES
Internal:	☐	☐	☐	☐	☐	☐	☐
External:	☐	☐	☐	☐	☐	☐	☐

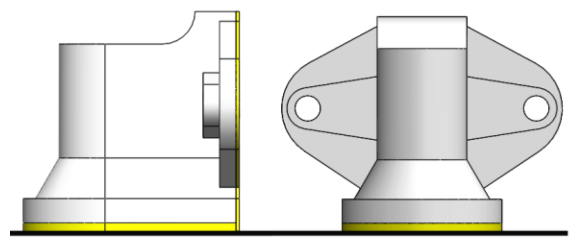
Assembly 2-3 and its explosive views





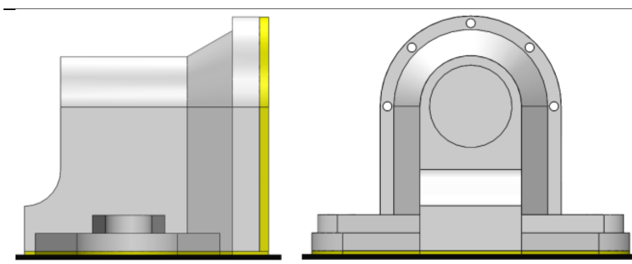
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



Internal: EU U WU NE WS S ES

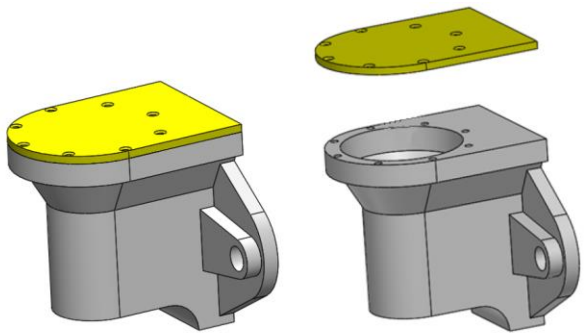
External: EU U WU NE WS S ES

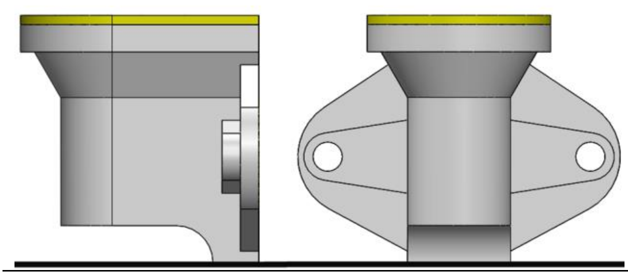


Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES

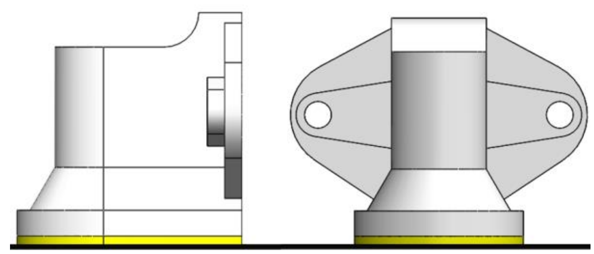
Assembly 2-4 and its explosive views





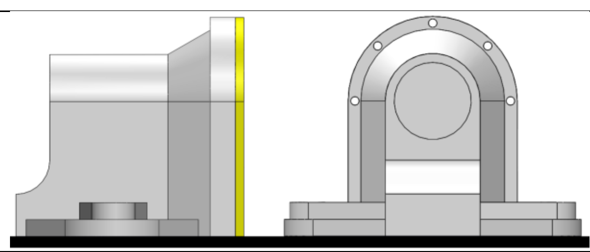
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



Internal: EU U WU NE WS S ES

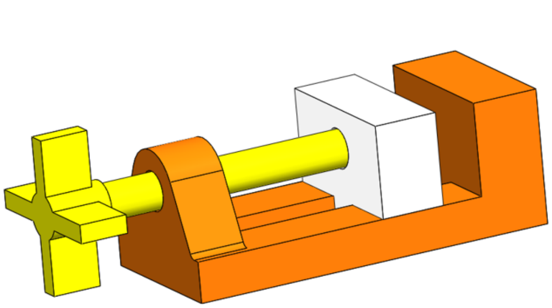
External: EU U WU NE WS S ES

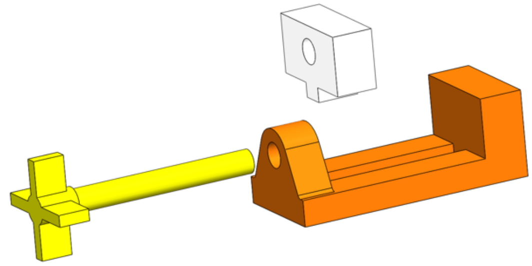


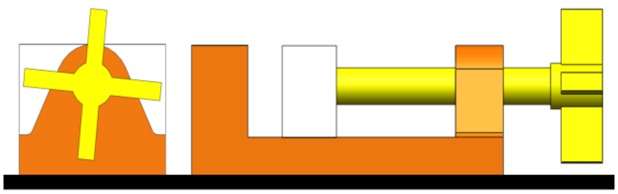
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES

Assembly 3-1 and its explosive views

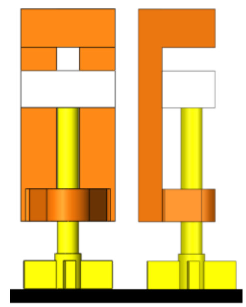






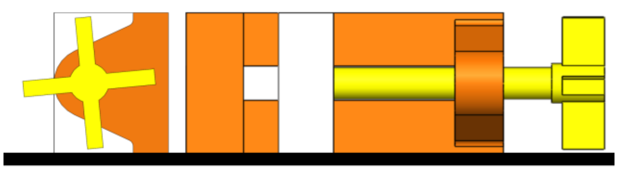
Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐



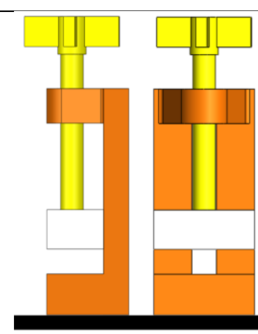
Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐



Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

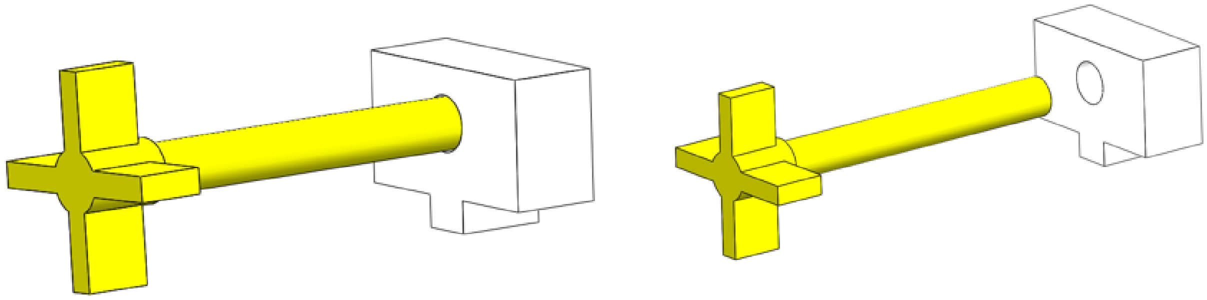
External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

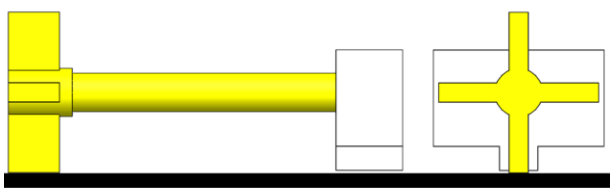


Internal: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

External: EU ☐ U ☐ WU ☐ NE ☐ WS ☐ S ☐ ES ☐

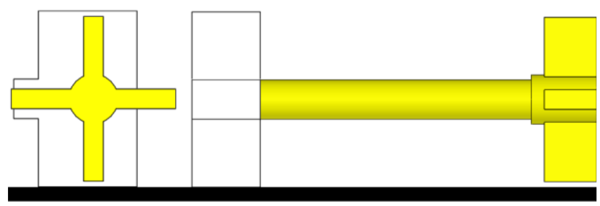
Assembly 3-2 and its explosive views





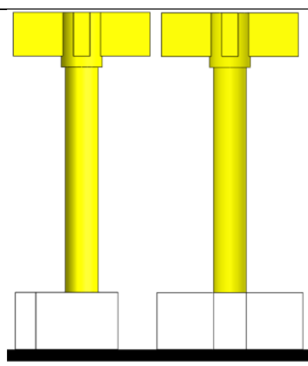
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



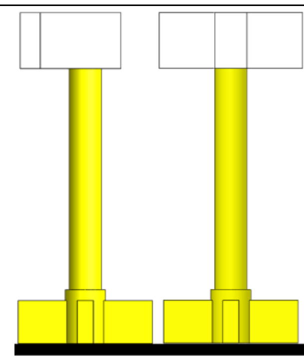
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



Internal: EU U WU NE WS S ES

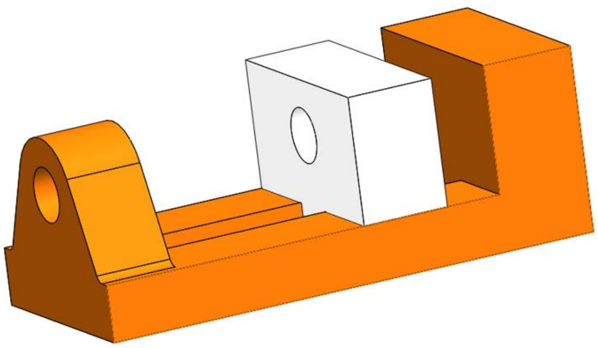
External: EU U WU NE WS S ES

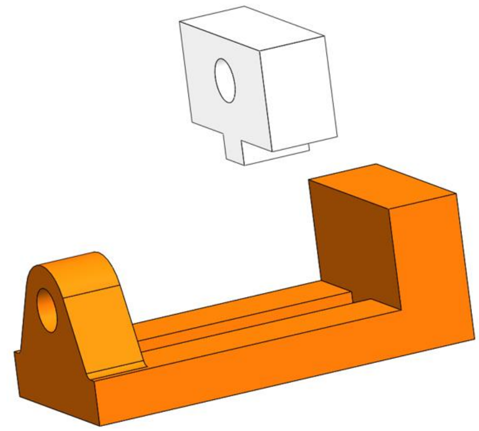


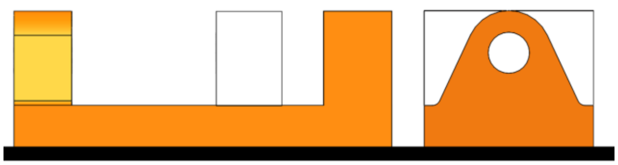
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES

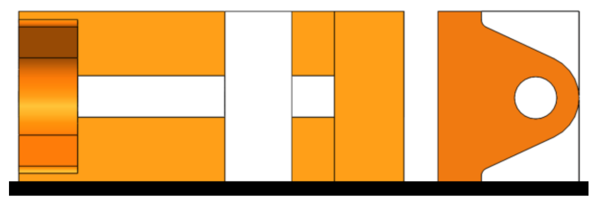
Assembly 3-3 and its explosive views



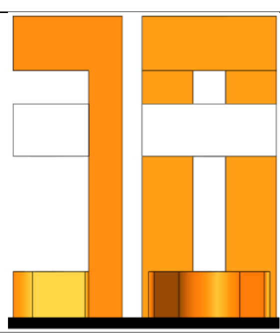




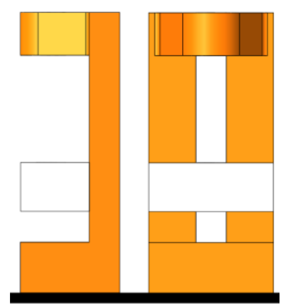
	EU	U	WU	NE	WS	S	ES
Internal:							
External:							



	EU	U	WU	NE	WS	S	ES
Internal:							
External:							

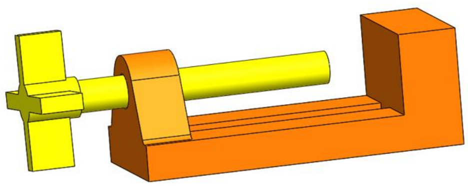


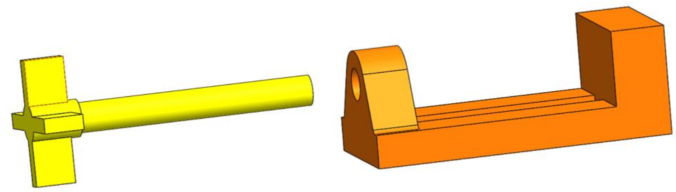
	EU	U	WU	NE	WS	S	ES
Internal:							
External:							

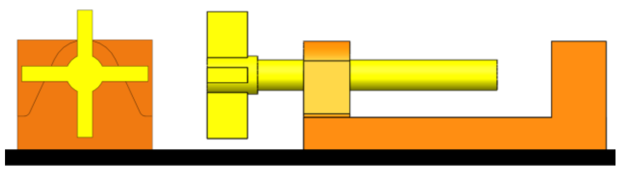


	EU	U	WU	NE	WS	S	ES
Internal:							
External:							

Assembly 3-4 and its explosive views

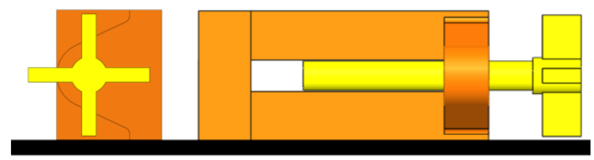






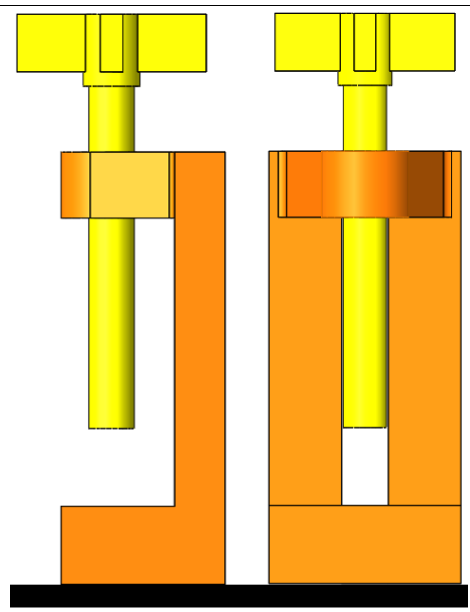
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



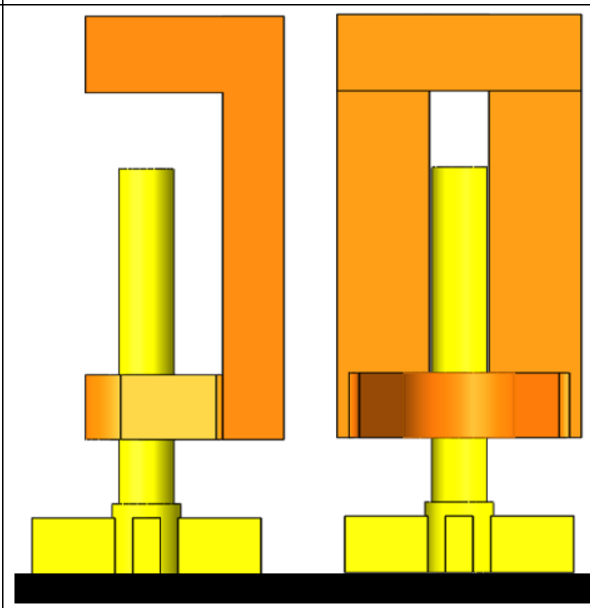
Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES



Internal: EU U WU NE WS S ES

External: EU U WU NE WS S ES