Evaluating Assemblies of Planar Parts Using the Liaison Graph and System Dynamics

Dimitrios Papanikolaou
Massachusetts Institute of Technology, USA
dimp@mit.edu

Abstract. *Current research on design and fabrication of planar part assemblies focuses on generative design methods, leaving analysis and evaluation of assemblability to be studied with empirical methods such as physical mockups. As a consequence, there is little understanding on whether a design is assemblable, or on how much time the assembling process might take. This paper proposes a new formal method to evaluate assemblability of interlocking planar parts that uses Network Analysis to evaluate assembly structure and System Dynamics to evaluate performance of assembling process.*

Keywords: *System Dynamics; Network Analysis; assembly; liaison graph; Digital Fabrication.*

Introduction

Current studies in Digital Fabrication focus on automating design and fabrication of assemblies of planar interlocking parts that are manufactured at custom shapes using 3-axis CNC routers (Sass 2006). These studies explore the limitations of design by manufacturability and assemblability. The workflow concept of these studies is based on the decomposition of an initial form into constructible parts, fabrication, and finally assembly of them to formulate the artifact. Unfortunately, most of assembly incompatibilities are discovered either during construction, or by building physical mock-ups with a significant loss in both time and cost, and a debatable reliance.

Every Digital Fabrication project embeds a certain degree of difficulty of assembly. This degree depends partly on the structure of the assembly design, and partly on the performance of the assembling system. For example, a project that involves a design of highly interconnected custom parts with complex interfaces that will be assembled by a poorly organized group of unskilled assemblers has a higher degree of difficulty than a design of parts with few connections and simple interfaces that will be assembled by a well organized group of skilled assemblers. Therefore, estimating the difficulty of production of designs is significant information because designers can predict conflicts and optimize the design.

Background

Previous work in Architecture

Previous research in understanding assemblability in architecture has focused on two main directions: CAD modeling (3D, 4D) and Physical Mockups.

3D CAD modeling of assemblies is based on an assembly file that includes individual part files. The design methodology is called constrain-based design and is based on constraining the part models inside the assembly model. However, studying assemblies in CAD is inadequate for two main reasons: first, a CAD model may have any structure of constraint delivery, but an assembly has always one. Second, CAD modeling represents the final state of the assembly, when all parts have been put together, but not the process of putting these parts together. 4D CAD modeling has been used for clash detection during assembly sequence. However, 4D modeling fails similarly to describe actual constraint delivery between parts. Moreover, CAD 4D is not able to define a proper assembly sequence. As a consequence, by studying a CAD model, the designer cannot tell if an assembly design might be assembled, nor he can make any estimation of the difficulty of the assembly sequence.

Physical mockups have been used during design development to test assemblability. However, there is a significant loss in time and cost. Moreover, in this fashion, testing is empirical, understanding the solution to the geometrical problem is obscure, and design development becomes intuitive. Clearly, designers need efficient tools to study and evaluate assemblies.

This paper deals with the following problem: How to define a formal methodology to evaluate the difficulty of assembly of a design? There are two issues to consider in this question: first, how to describe assembly structure; second, how to measure performance of the assembling system.

Assembly structure description has been studied in Product Development, and Manufacturing using Network Analysis methods such as the liaison graph. The liaison graph is a directed acyclic graph whose nodes represent parts and arcs represents liaisons. Direction of arcs indicates order of constraint delivery between two different parts. In a liaison graph no cycle is allowed since that would mean that a part constrains itself through a chain of constraint deliveries.

Performance of systems has been studied in Industrial Management using System Dynamics. System Dynamics (Forrester 1961) is a methodology coming from Control Theory, for studying the behavior in time of complex feedback systems. A System Dynamics model is a bipartite network consisting of states (stocks), actions that affect the states (flows) and decision variables that control the actions. System Dynamics has been extensively used to simulate supply chain performance.

While Network Analysis provides a concise and formal way to study systems' structure and System Dynamics provide an effective way to simulate systems' performance it is not clear how a liaison graph could provide information on a System Dynamics model of an assembly process.

Proposal

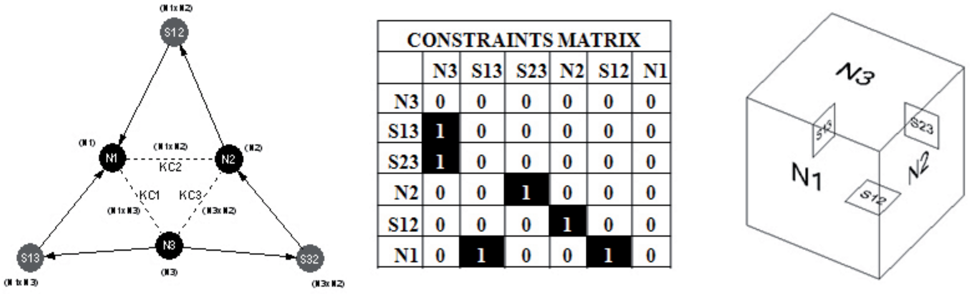
This paper proposes a new method to evaluate the degree of difficulty of assembly of interlocking planar parts using Network Analysis to describe assembly structure and define an assembly sequence and then a System Dynamics model to execute the assembly sequence.

Methodology

Structural Analysis of assembly design

An assembly is a system of parts connected through liaisons, the goal of which is to deliver one or more key characteristics (KC). A KC is a requirement that the assembly must meet such as a minimum distance between two parts (Whitney 2004). This paper deals with assemblies of planar, perpendicularly interlocking parts.

Assembly structure is described through the liaison graph and the corresponding adjacency matrix. The adjacency matrix of a liaison graph with n nodes is an n*n matrix whose columns and rows represent the nodes of the network. A mark in column i and row j represents a link from node i to node j. This means that in order to find the precedents of node j we first trace row j and record all marks that we find; then we identify the nodes that correspond to the columns of these marks. Similarly, to find the decedents of node j we have to trace column j and record the rows that correspond to marks that we find.



CONSTRAINTS MATRIX						
	N3	S13	S23	N2	S12	N1
N3	0	0	0	0	0	0
S13	1	0	0	0	0	0
S23	1	0	0	0	0	0
N2	0	0	1	0	0	0
S12	0	0	0	1	0	0
N1	0	1	0	0	1	0

Figure 1
The liaison graph (left) and adjacency matrix (middle) of an assembly of 6 parts (right).

The number of connections that a node has with other neighbor nodes is called the degree of the node. If the network is directed, then each node has an in-degree and an out-degree.

Evaluating an assembly sequence

An assembly sequence is a valid way to trace the liaison graph from precedent nodes to decedent nodes starting from a root node. Validity depends on connectivity rules that are explained later in this paper. A root node is a node that has no precedents. The difficulty of each step relates to the in-degree of the node which indicates the number of simultaneous liaisons that must be achieved during that step. For example, a part will be more easily connected to another part if it has one liaison rather than if it has multiple liaisons. Therefore, the in-degree distribution along an assembly sequence indicates the difficulty of the assembling process.

In the adjacency matrix an assembly sequence can be represented as an ordering of the rows and columns. Such ordering can be derived by rearranging the rows and columns of the adjacency matrix so the resulting matrix has all its marks below the diagonal (Figure 1, middle). The sequence of the sums of each column gives the in-degree distribution of the assembly sequence.

Assemblability rules for Planar Part Assemblies

In what follows I present a set of rules for evaluating planar part assemblability and defining a valid assembly sequence.

1. A planar part A is represented by the normal vector a of its plane. Each node in the liaison graph is assigned the value of the normal vector of the part it represents.
2. A liaison connecting part A with part B is represented by the liaison vector ab. Each arc in the liaison graph is assigned the value of the liaison vector it represents. In the liaison graph liaisons are represented by solid lines.
3. Two nodes can be connected by a liaison if and only if the cross product of their normal values is 0 or 1. If it is 0 then the parts are perpendicular; if it is 1, then the parts are coplanar.
4. There are 3 liaison types to connect 2 perpendicular parts A and B: a, b, ab. Type a means that B connects to A along the direction of the normal vector of A. Type b means that B connects to A along the direction of the normal vector of B. Type ab means that B connects to A along the direction of the cross product of A and B (Figure 2).
5. An Adjacency Key Characteristic (AKC) between two adjacent parts A and B is the cross product vector of A and B and it indicates the direction of the edge between A and B. In the liaison graph an AKC is represented by a dashed line.

6. The in-degree of a node defines the difficulty of its assembly step.
7. Two or more nodes can be clustered into one subassembly and represented as one node.
8. A part can be located by another part by one or more liaisons. If the liaisons are more than one then their vectors must be parallel.
9. Two parts can be connected by a third part which is perpendicular to them. The third part has a normal value equal to the cross product of the two parts.
10. A part can be installed if all of its predecessor parts have been installed first.
11. If a part has zero in-degree but non-zero out-degree, then this part is a root part.
12. If a part has zero out-degree but non-zero in-degree then this part is an end part.

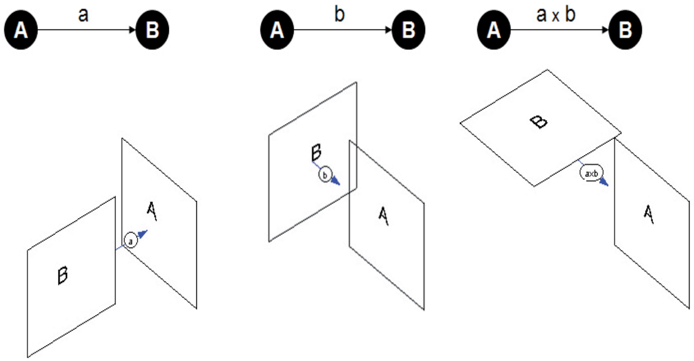


Figure 2
The 3 liaison types for planar part assemblies.

Defining an assembly sequence

To find a valid assembly sequence we start testing all possible ways to trace the liaison graph; on every step we select one of the 3 possible liaison types and verify that all incoming liaisons are of the same type (a, b, or ab).

Dynamic analysis of an assembling system

The basic module of a System Dynamics model is a goal-seeking feedback system of two stocks, the start stock and the end stock, that are connected by a flow. The flow is controlled by a decision function which tries to equalize the level of the end stock to a desired goal level. This system is called a feedback system because the decision function uses information from a past result (the level in the end stock) to control a future action (the rate in the flow).

This paper proposes a System Dynamics implementation to measure performance of an assembling process in executing an assembly sequence. An assembling process is defined as a process that creates liaisons to a set of parts according to an ordered in-degree distribution. For example, an in-degree distribution of [0,1,1,2] means that the first part needs no liaisons (root), the second, and the third parts need 1 liaison each, and the fourth part needs 2 liaisons to be assembled. Two stocks, the start stock and the end stock, describe the level of achieved liaisons in the system. In the beginning of the simulation the level of the start stock is zero because no part is assembled yet. In the end of the simulation, the level of the end stock is equal to the total number of links in the liaison graph, because all parts have been assembled. The flow that changes the two levels is controlled by the assembling rate. If the average assembling time per part is fixed, then the assembling rate will fluctuate according to the in-degree distribution sequence which denotes the difficulty of the assembly sequence. More refined

System Dynamics models that include learning factors, error factors, etc. can be built starting from this basic structure.

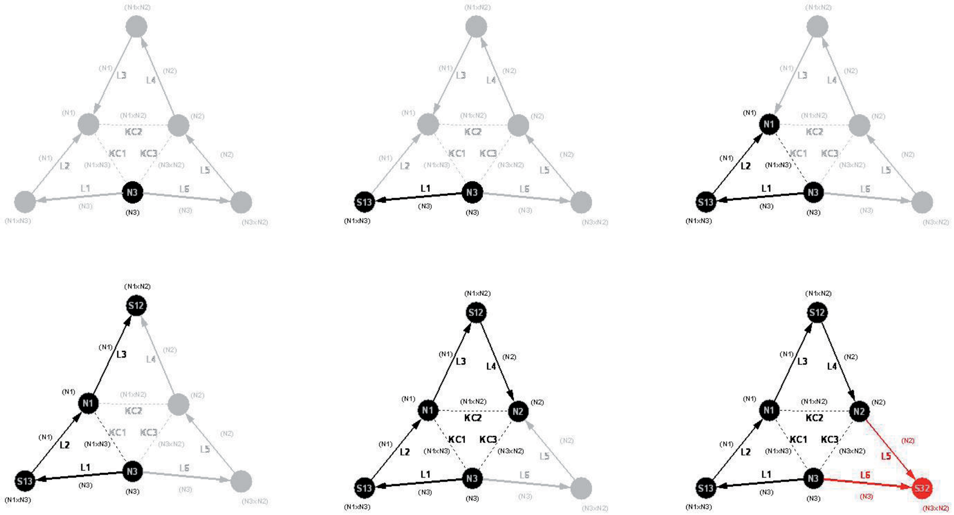


Figure 3
An invalid assembly sequence of the 6 parts of figure 1. The 6th step violates rule 8.

Building a System Dynamics model of an assembly process

A System Dynamics model of the assembling process can be described by a process-state network model as follows: an assembling process P assembles two parts by changing their liaison states A, and B (Figure 4). As and Bs are the start states, before the assembly process connects them. Ae and Be are the end states, after the assembly processes Pa and Pb connect them. Ag and Bg are the goals of the two assembling processes. t1 and t2 are the times of Pa and Pb respectively. In the first time frame Pa modifies A from As to match it to Ae by the decision function Da. When Ae matches Ag the decision function Da passes control to decision function Db that controls assembling process Pb. Pb uses Ae to modify Be to match it with Bg.

Experiments

Experiment 1: Structural analysis of a chair's assembly

The following experiment refers to the design, fabrication and assembly of a chair made from interlocking planar parts. The chair was designed in 3D CAD modeling software (Rhinoceros V4.0) and the parts were fabricated from 1" plywood sheets in a 3-axis CNC router. The assembly consisted of 29 interlocking pieces of plywood: 16 where horizontal and 13 where vertical. Modeling of the assembly focused on representing two states of the artifact: the assembled form where all parts are put together and the flattened parts in cut-sheets for fabrication. The assembled form seemed to be a valid configuration of the artifact with no clashes between the solid volumes of the parts. Unfortunately, assembly process stopped at a certain point; installation of parts was impossible due to conflicts in the installation vectors. The designers had no tools to describe, understand, and evaluate the assembly process.

A representation of the assembly with the liaison graph clearly shows that the assembly sequence is in fact impossible due to installation vector

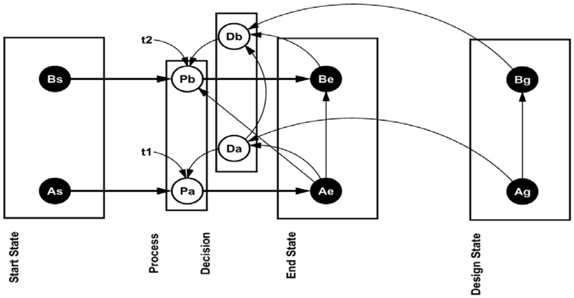


Figure 4
A process-state model of an assembly of 2 parts.

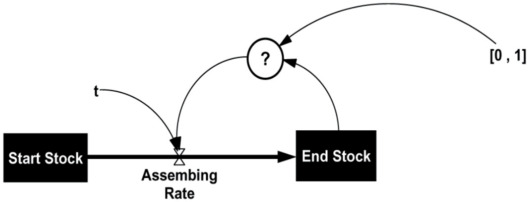


Figure 5
A System Dynamics model of an assembly of 2 parts.

incompatibility between parts (Figure 6). For simplicity this liaison graph represents a similar assembly of 18 parts: 9 horizontal and 9 vertical. All liaisons are of type ab (rule 4). From the liaison graph we can have a formal understanding of the assembly sequence: the first part can be any horizontal or vertical member; in the experiment we selected the 6th horizontal member from the bottom. In the liaison graph, the next 9 pieces can be easily installed by one liaison each. However, starting from the 11th part all other parts need to achieve 9 simultaneously non-parallel liaisons; this is impossible.

The analysis shows that assembly should jam at the 11th step because after that each next part would have to simultaneously connect with nine non-parallel installation vectors with the rest of the assembly. However, real assembly jammed later due to the looseness of the notches of the parts.

Experiment 2: Structural and dynamic analysis of Façade Panel's assembly

The second experiment refers to the design, fabrication, and assembly of a mockup of a façade panel. Design development took place in a parametric 3D

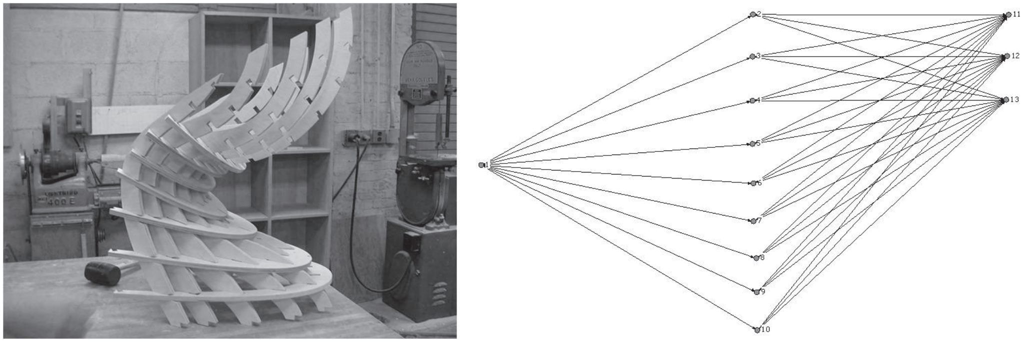


Figure 6
Physical model of the chair (left), and liaison graph (right,) showing the step where assembly jammed

CAD modeling software (CATIA V5 R18). In this case, while the assembly was successful, it proved to be difficult, and took significantly more time than the designer expected. While this example is relatively simple, including a small number of parts, it clearly demonstrates the lack of tools that designers need to understand assembly process.

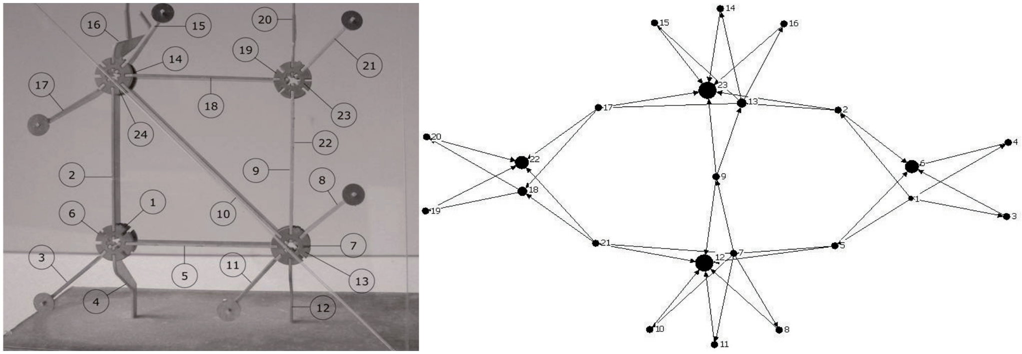
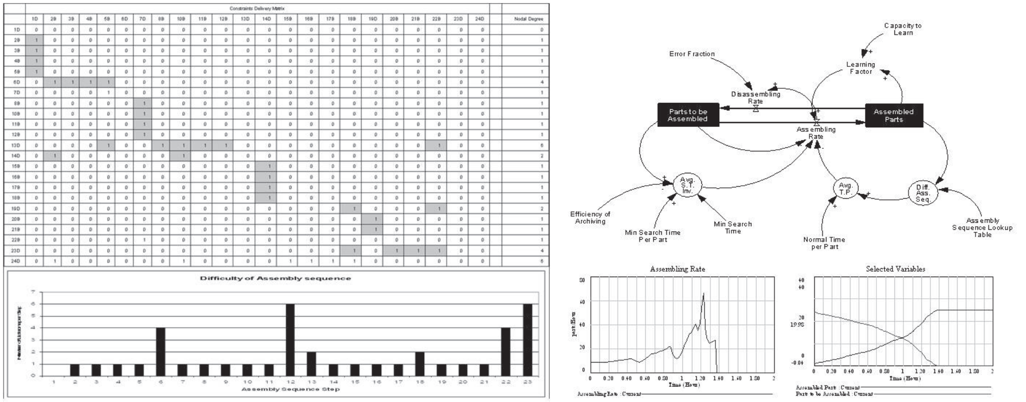


Figure 7
Physical model of the façade panel assembly (left), and liaison graph (right)

A representation of the assembly with the liaison graph shows that while the assembly is possible, there are two steps in the assembly sequence of high difficulty because they need simultaneous connections. The nodal degree distribution along the actual assembly sequence shows the difficulty of each step as a function of the number of connections that have to be achieved with the rest of the assembled artifact. The nodal degree sequence is then inserted as input in the simple System Dynamics model that represents the assembling process. The model clearly shows that assembling rate will significantly drop at the 12th and 23rd step of the assembly sequence.

Explanation of the System Dynamics model

The structure of the System Dynamics model (Figure 8, right) consists of two stocks, the Parts to be Assembled and the 'Assembled Parts'. Parts move from one stock to the other through the 'Assembling Rate'; the faster the Assembling Rate, the less time will take for the assembly to be completed. However, due to errors some parts will need to be disassembled and reassembled. Therefore there is a Disassembling Rate



Constraints Delivery Matrix

	1D	2B	3B	4B	5B	6D	7D	8B	10B	11B	12B	13D	14D	15B	16B	17B	18B	19D	20B	21B	22B	23D	24D		Nodal Degree
1D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
2B	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
3B	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
4B	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
5B	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
6D	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		4
7D	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
8B	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
10B	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
11B	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
12B	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
13D	0	0	0	0	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0		6
14D	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0		2
15B	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		1
16B	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		1
17B	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		1
18B	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		1
19D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0		2
20B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0		1
21B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0		1
22B	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1
23D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0		4
24D	0	1	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0		6

Figure 8
Adjacency matrix with in-degree distribution (left), and System Dynamics model with simulations (right)

that removes parts from the Assembled Parts stock back to the Parts to be Assembled stock.

The Assembling Rate depends on the following factors: first, the Learning Factor and the Capacity to Learn; the more we assemble the more skillful we get which improves our assembling rate. Second, the Average Search Time in Inventory (Avg.S.T.Inv); average search time depends on Efficiency of Archiving, which is how well organized the parts are in the inventory. Third, on the difficulty of the assembly sequence that is given by the Assembly Sequence Lookup Table. The lookup table returns the in-degree of each step of the assembly sequence. The Disassembling Rate depends on the Error Factor and on the Assembling Rate.

Conclusion

This paper presented a theoretical framework to evaluate assemblability that consists of 2 steps: first, structural analysis of assembly design and definition of a valid assembly sequence; second, dynamic analysis of assembly process in executing the assembly sequence. The presented method is applied on planar part assemblies; however the theory can provide the basis for studying other kinds of assemblies, such as manufactured 3D component.

Application of System Theory methods in Digital Fabrication has benefits in both education and practice. In education it provides a formal way to study assemblability and it can provide a generative rule-based method for assembly design. In practice, it provides a powerful tool for augmenting evaluation and management of digital fabrication projects. Finally, another benefit is the high level of abstraction; it can be used relatively early in the design process.

Acknowledgments

Experiment 1 was a team project in class 4.580: Inquiry into Computation and Design (Prof Terry Knight, Prof Lawrence Sass) at the Massachusetts Institute of Technology in fall 2006. Team members: Joshua Lobel, Magdalini Pantazi, Dimitris Papanikolaou.

Experiment 2 was an individual project in class 4.592 Special Problems in Digital Fabrication (Prof Lawrence Sass) at the Massachusetts Institute of Technology in Spring 2007. Team member: Dimitris Papanikolaou.

References

Eppinger, S. D., Whitney D. E., Smith, R. P., and Gebala, D.: 1994, A Model-Based Method for Organizing Tasks in Product Development, Research in Engineering Design. 6(1), pp. 1-13.

Forrester, J. W. (ed.): 1961, Industrial Dynamics. Pegasus Communications.

Sass, L.: 2006, A Wood Frame Grammar; A generative system for digital fabrication, International Journal of Architectural Computing, 01(04).

Sterman, J. D. (ed.): 2000, Business Dynamics: Systems Thinking and Modeling for a Complex World, McGraw-Hill, Boston.

Steward, D. V. (ed.): 1981, Systems Analysis and Management: Structure, Strategy and Design, Petrocelli Books.

Whitney, D. E. (ed.): 2004, Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development, Oxford University Press, New York.

Whitney, D. E.: 2006, Network Models of Mechanical Assemblies, International Conference on Complex Systems, Boston.