121st ASEE Annual Conference & Exposition
Indianapolis, IN
June 15-18, 2014

Paper ID #9921

Measuring Qualities of Different Engineering Design Process Models: A Critical Review

James Logan Oplinger, Arizona State University
Dr. Micah Lande, Arizona State University, Polytechnic campus

Dr. Micah Lande teaches human-centered design innovation at Arizona State University and researches how engineers learn and apply a design process to their work. He is an assistant professor in the Department of Engineering on Arizona State University's Polytechnic campus.

Measuring Qualities of Different Engineering Design Process Models

Abstract

The engineering design process is a core piece of engineering education. Students are typically introduced to the process during their first semester of school and are taught many methods to improving their own design process. Several models have been introduced by professional teams, researchers, and students, each having its own particular use and qualities. A review of such models was conducted grading each of their qualities on a scale. Overall scores showed that some sort of interactive and iterative web is the best design process to use. However, other models brought forth important pieces or different perspectives that can be integrated into other designs.

1. Introduction

The engineering design process is a diverse method that engineers use in order to solve a particular problem. Throughout various studies[1] people have attempted to determine what form of the engineering design process best fits actual application. Others[2-10] have attempted to see how engineering students progress in their unique engineering design process structure. Through these particular studies the iteration is the determining factor in whether or not a student has improved. Still other researchers[11-13, 14-16] base the value of a process method off of the time saved by that particular process. Here, we will discuss what makes particular processes more valuable than others, as well as what an ideal process would be.

Through a length of research Atman and others[2-10] began studying students in engineering programs. In these studies they recorded how students solved particular problems, taking note of what modules of the design process a student was talking about. From this data[6] they would determine how sophisticated a student's methods were. As stated earlier, students were rated based on their number of iterations and transfers between different modules. An increase in iteration and interchanging of steps is often associated with an increase of design quality[11]. From this we can gather a basis for part of our own analysis of design processes.

What follows is the process of determining an ideal process. We begin the analysis by detailing tools and scales used to rate various processes. From there the types of processes are broken into four categories: structured and step models; iterative and interactive webs; abstract models; and mathematical models. In each category specific types of models are presented that branch from the broader categories. Following are analyses of each category and model types. After all of the processes are analyzed and compared a preferred model is proposed, incorporating methods from several high scoring or unique models.

2. Methods

This paper will explore a variety of models for the engineering design process and evaluate the ideal model level that each process indicates. Design process models will be graded high (1 point), moderate (0 points), or low (-1 point) on complexity, ease of use, appropriateness.

Complexity will be measured as the degree of steps, iterations, interchanges, and size of a model. Ease of use will be measured as the ability to utilize a model with little prior knowledge. Appropriateness will measure whether a model is ideal for modeling the engineering design process. The ideal model will have moderate complexity, relatively high ease of use, and be appropriate.

The three factors—complexity, ease of use, and appropriateness—were chosen based off of several engineering education studies[2-10]. The factor complexity in this context refers primarily to iteration, a quality that is used in studies from Atman and others[4-8] to rate students' learning. Ease of use refers to functionality and simplicity. This factor allows others to retrace steps that the user went through for their process. It also allows the concept of the process to be taught easily to students. Appropriateness is necessary to evaluate if the model is able to measure a variety of engineering design processes for different problems. Table 1 shows the rubric that will be used to evaluate these design processes.

Design Process Model	**Complexity**	**Ease of Use**	**Appropriateness**	**Ideal Model Level**
Model 1				
Model 2				
Model 3				

Table 1: Rubric for Rating Design Processes

Once the general analysis is conducted each process is given an ideal model level. This level will be based on results of the other scales and will be graded as low (less than -1 total factor score), moderate (between -1 and 1 total factor score), or high (more than 1 total factor score). The ideal model level will be a reflection of how a model compares to other proposed ideas and key concepts.

Coauthors collaborated in order to determine how particular models and model categories rate on each scale. In addition, qualities emphasized by Atman, Haik, and other researchers[1, 3, 5, 17] were taken into consideration before rating a particular design model. Overall, particular ratings are objectively based on experience.

The evaluated models come from various engineering journals and books. These process models are often seen in engineering courses and have applications in a real world environment. In addition to these professional models we will include examples of student design process models. The mixture of the two sources of design processes will hopefully add to insights brought from this research.

As stated in the organization of the paper models will be reviewed individually and as a group. With this evaluation it will be possible to learn what generalized view of modeling is most appropriate for students. If models result in high scores but are in low scoring categories it may be possible to apply certain aspects of these specific models to high scoring categories. From this fusion we may be able to create a new model that proves to be more appropriate than others.

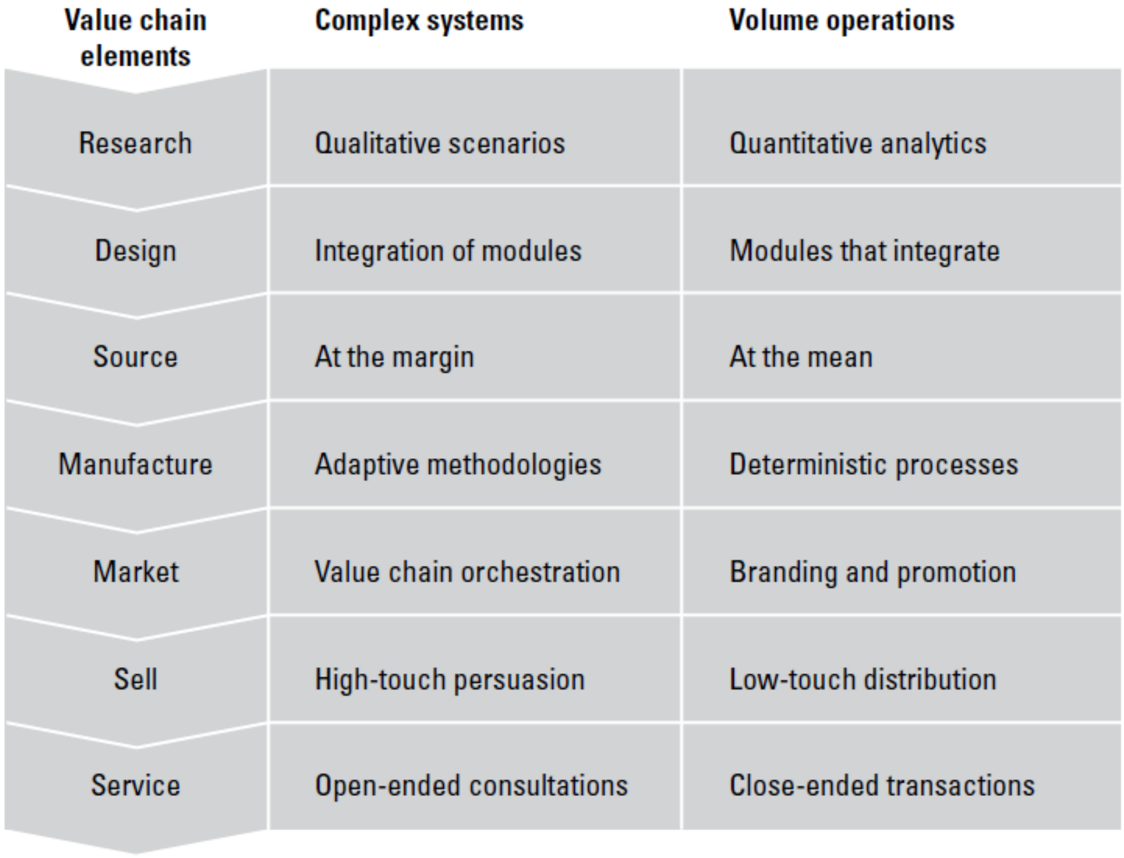
3. Structured and Step Models

Structured and step models are relatively generic fits in the design process world. Simple step models often are included as “novice” design processes. Still other models prove to be more complex in structure. These models are often used to predict time and order of production. They include linear models[18, 19], value chains[20], Design Solution Matrices[11], and evolutions of these matrices[12, 14]. The following section walks through these structured designs.

3.1 Linear Design Processes

The waterfall design process is often referred to as a novice process due to its simplicity. Designers progress through steps in a linear path, checking off each step as they pass through. The “waterfall” process is seen in many engineering freshmen representations, especially in preliminary activities where the student has little to no prior knowledge of the engineering design process.

A method almost identical to the “waterfall” process is the simple value chain[20]. Figure 1 displays the chain’s elements, complex systems, and volume operations. As designers progress through each element they must complete tasks involved with the complex systems and volume operations. Other than this added text the value chain follows exactly like the “waterfall” process.



Value chain elements	Complex systems	Volume operations
Research	Qualitative scenarios	Quantitative analytics
Design	Integration of modules	Modules that integrate
Source	At the margin	At the mean
Manufacture	Adaptive methodologies	Deterministic processes
Market	Value chain orchestration	Branding and promotion
Sell	High-touch persuasion	Low-touch distribution
Service	Open-ended consultations	Close-ended transactions

Figure 1. Value Chain Example[20]

Researchers Ishino and Jin developed a three-layer design process model[19] to evaluate knowledge used by designers (Figure 2). This design process consisted of an event-layer, an operation-layer, and a product-model-layer. The operation-layer was a linear flow of steps conducted by the designer; each action consisted of multiple events. The event-layer showed the linear progress of events that built up each operation. Along certain points of the operation-layer there would be outputs to and inputs from the product-model-layer called alternatives. Furthermore, along the operation-layer is a "black box" where the model illustrates there may be a larger amount of steps not pictured within the generic model. In many ways the three-layer model is closely related to the waterfall model; however, this model fuses a simplified and elongated version of the waterfall model, providing additional information along the process.

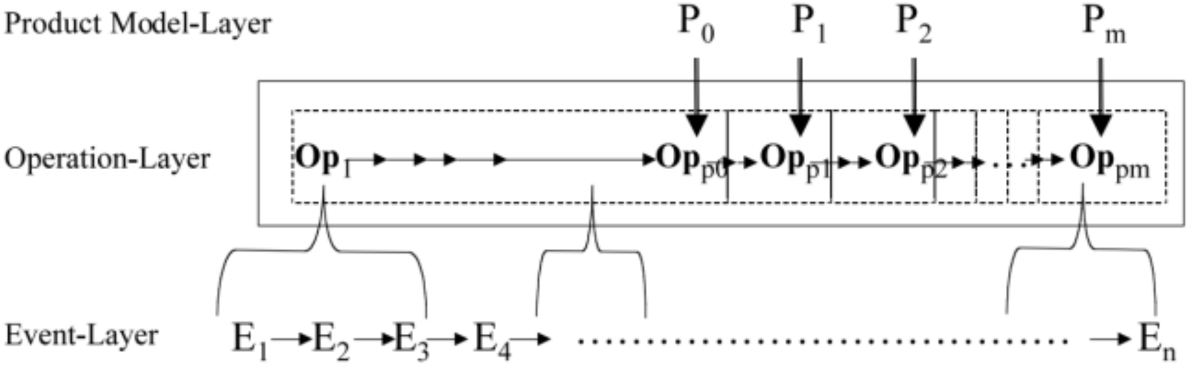


Figure 2. Ishino and Jin 3-Layer Process[19]

A common evolution of the "waterfall" design process is the generic step representation[18] (Figure 3). A generic step representation usually involves a group of tasks (Identifying the issue, generating concepts, prototyping, finalizing) and may involve detailed description of any one step (such as the different levels of prototyping). This form of step model is an elongated "waterfall" process.

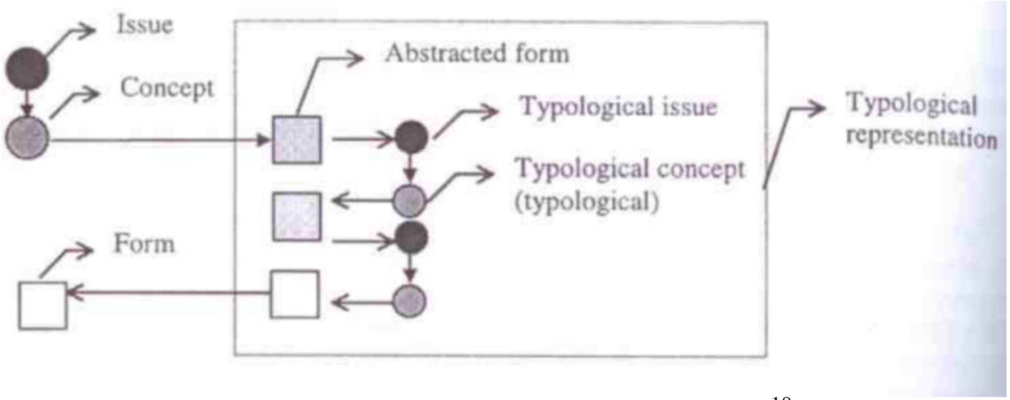
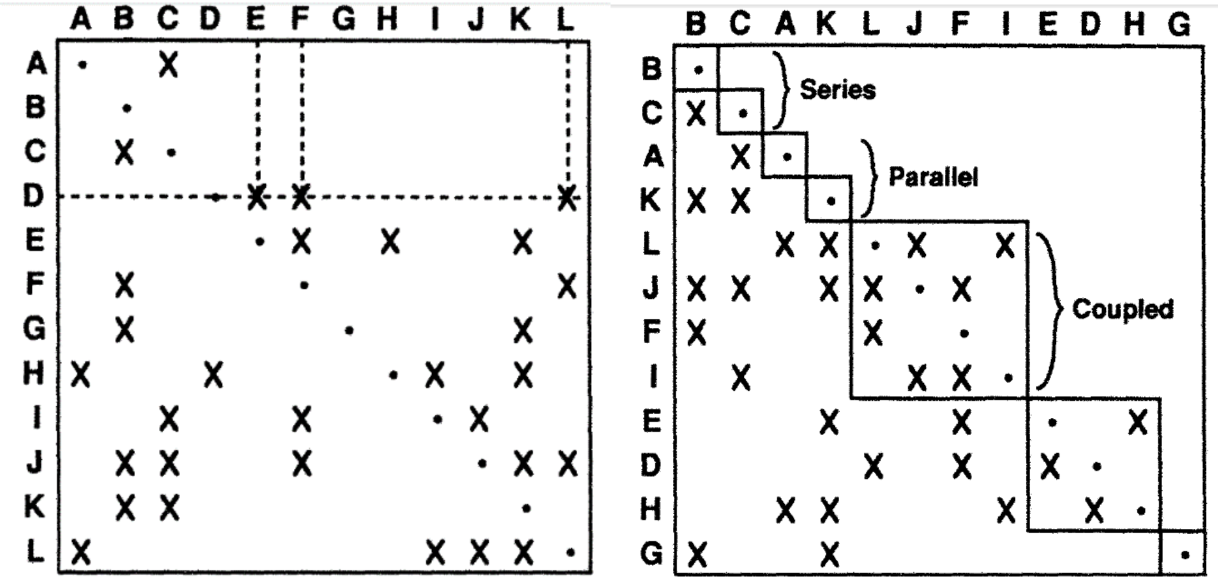


Figure 3. Simple *Waterfall* Design Process[18]

As seen even a focused section of processes like "Linear Design Processes" has multiple subsections within. In general, these models have instruction based structures.

3.2 Matrices

As an engineer progresses through the design process he or she fulfills certain prerequisites to continue on to the following step. In the industry one of the primary goals is to turn out a product in a timely manner; therefore, a format was developed, the Design Structure Matrix[11] (DSM), which could be used to signal what order of steps to use. Tasks are listed on the top of the grid along both the x and the y axis. Steps then have their dependencies marked within the grid with Xs. The goal of such an analysis is to make the process as "lower triangular" as possible by rearranging the order of steps so that little to no dependencies are in the upper right half of the grid. Figure 4 is an example that displays the change from a raw matrix to a triangulated matrix[11].



	A	B	C	D	E	F	G	H	I	J	K	L
A	•		X									
B		•										
C		X	•									
D				•	X	X						X
E					•	X		X			X	
F		X				•						X
G		X					•				X	
H	X			X				•	X		X	
I			X			X			•	X		
J		X	X			X				•	X	X
K		X	X								•	
L	X								X	X	X	•

	B	C	A	K	L	J	F	I	E	D	H	G
B	•											
C	X	•										
A		X	•									
K	X	X		•								
L			X	X	•	X		X				
J	X	X		X	X	•	X					
F	X				X		•					
I		X				X	X	•				
E				X			X		•		X	
D					X		X		X	•		
H			X	X				X		X	•	
G	X			X								•

Figure 4. Pre-Triangulated DSM and Post-Triangulated DSM[11]

As you can see it is not always possible to completely triangulate a matrix. Many times interdependent blocks or "couples" exist where steps have to be done in a back-and-forth order in order to accomplish each task. A couple may signal a large section of the design process such as concept design. In this case the smaller interwoven tasks would be along the lines of develop design concepts, determine characteristics, etc. An arguable advantage of these couples is an increase in design quality due to the interaction and reflection of design groups[11].

In addition to coupling a simplified DSM illustrates areas where tasks have to be run in series or have the opportunity to be run in parallel. In Figure 4 tasks B and C have to be run in series because task C is dependent on task B; however, tasks A and K can be run in parallel because neither is dependent on each other, nor are their prerequisites dependent on the other. The opportunity to run tasks in parallel is used to help speed production rate, a primary task of the DSM[11].

Smith and Eppinger propose an evolution of the DSM called the Sequential Iteration Model[14] (SIM). This model, based on the DSM, uses probability to determine what the order of tasks should be. Instead of dependencies being listed in the non-diagonal grid cells percentages are given. These percentages show the probability of a particular step leading to another particular step, given that both steps are in the same stage. The matrix and Markov chain in Figure 5[14] illustrate an interchange of two different tasks across two stages. Eppinger, et al, illustrates a version of the SIM that allows for variable task lengths and repetition of multiple tasks[12]. To find further discussion of the mathematics behind the SIM refer to section 6.1 "Markov Chains."

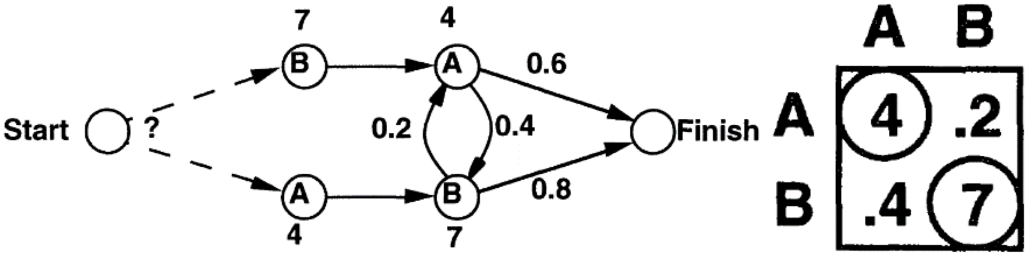


Figure 5. Markov Chain and associated SIM[14]

In addition to the SIM another matrix was derived from the DSM, the Work Transformation Matrix[16] (WTM). This model determines what tasks or sets of tasks will take significantly more work than other tasks[1]. Due to the WTM's more mathematical principles it is further examined in section 6.2 "Work Transformation Matrices."

The Design Structure Matrix can be evolved and added onto in order to meet a particular need. In addition these matrices allow design teams to divide tasks or form subgroups based off of the results of triangulating and coupling. Timelines can be formed after a matrix is triangulated. The DSM and other matrices allow milestones to be placed and task orders be made.

4. Iterative and Interactive Webs

As an engineering student progresses through the program they are taught a new form of the engineering design process, iterative design. Basic iterative design involves looping through the design process, showing that it is never ending, but rather builds upon past projects/designs (Figure 6). From there the individual builds upon the process, evolving it into their own personal machine. The focus of iterative design processes is to increase design quality due to increased transition and collection time[5].

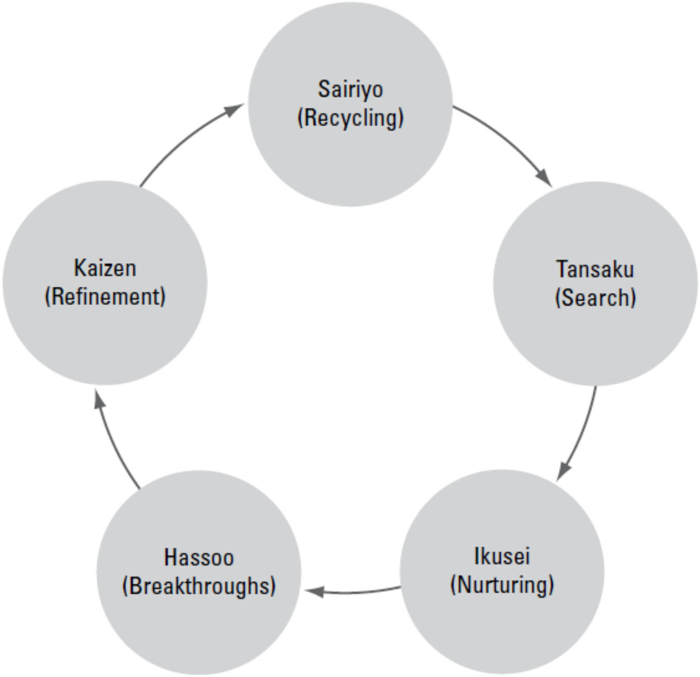


Figure 6: Basic Iterative Loop[20]

4.1 PERT Modeling

The Programming Evaluation and Review Technique[21] (PERT) evolved from the critical path model used to find the quickest solution to a project. PERT separates tasks into nodes and activity along arcs or arrows. These charts can have many, many tasks or only a few. Usually PERT diagrams have numbers associated with each node that are higher than the nodes they lead to[21]. NetMBA[21] gives the following steps to planning with a PERT system:

1. "Identify the specific activities and milestones."
2. "Determine the proper sequence of the activities."
3. "Construct a network diagram."
4. "Estimate the time required for each activity."
5. "Determine the *critical path*."
6. "Update the PERT chart as the project progresses.

This form of diagramming allows designers to see what resources are needed to complete what tasks along with a generalized order and communication line. When key concepts are followed designers can ensure a project will be completed on time and that milestones are completed on time. In Figure 7 a processing network[22] is based off of PERT modeling to show the flow of activities between resources. PERT proves to be a simple model of tasks.

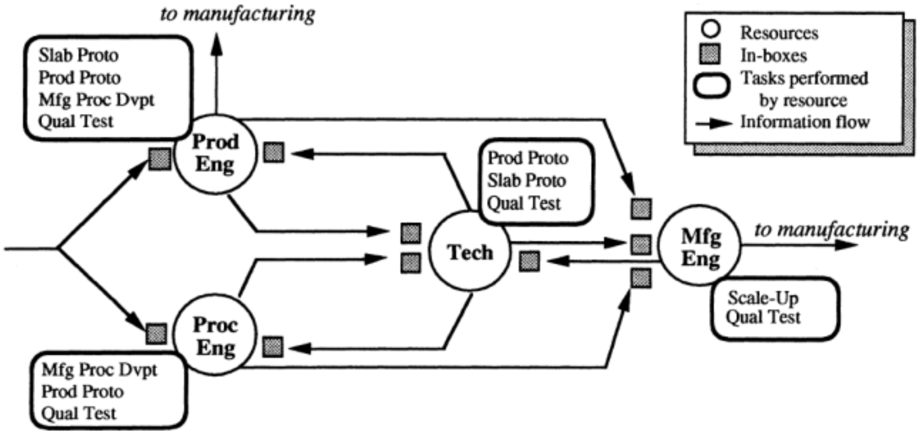


Figure 7. A process diagram based off of PERT[22].

4.2 Flowcharts and Networks

Many of the web models are similar to a simple flowchart. Often similar to the "waterfall" models, the primary difference is that these models usually have some sort of feedback loop within or throughout (Figure 8). This extra path is an improvement path where the designer, or a new designer, builds upon the created product or service to create a new or improved one. Figure 8 is an example where a design team works through a series of steps, often looping back between stages to improve design quality[20].

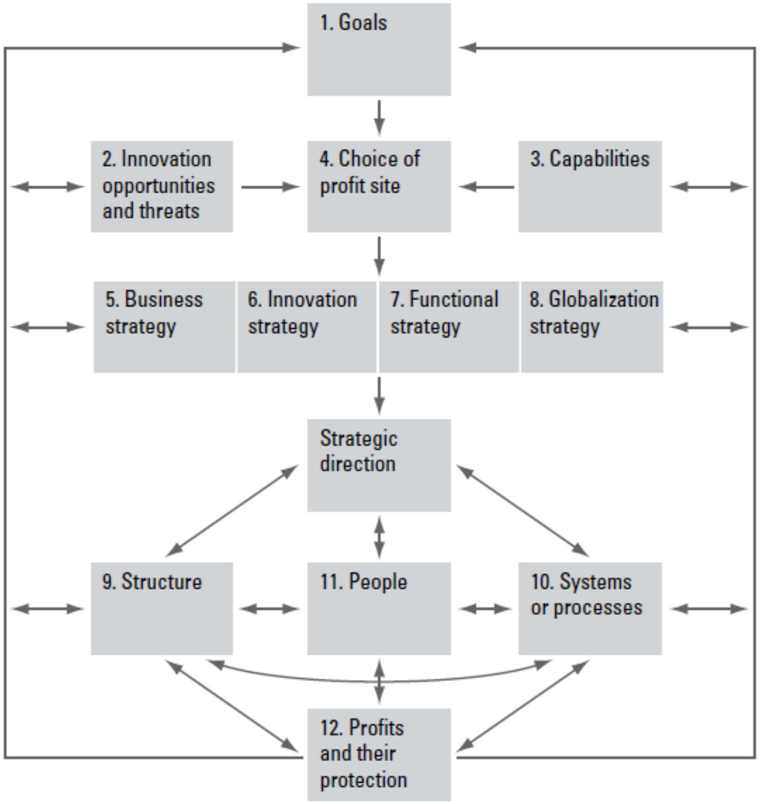


Figure 8: A basic flowchart example[20]

These flow processes can range from the simple model in Figure 9 to intricate groups of tasks like that of Figure 8. This flexibility allows personal design teams to clump common tasks into one group. When tasks are collected together it becomes easier to divide responsibility between teams and team members.

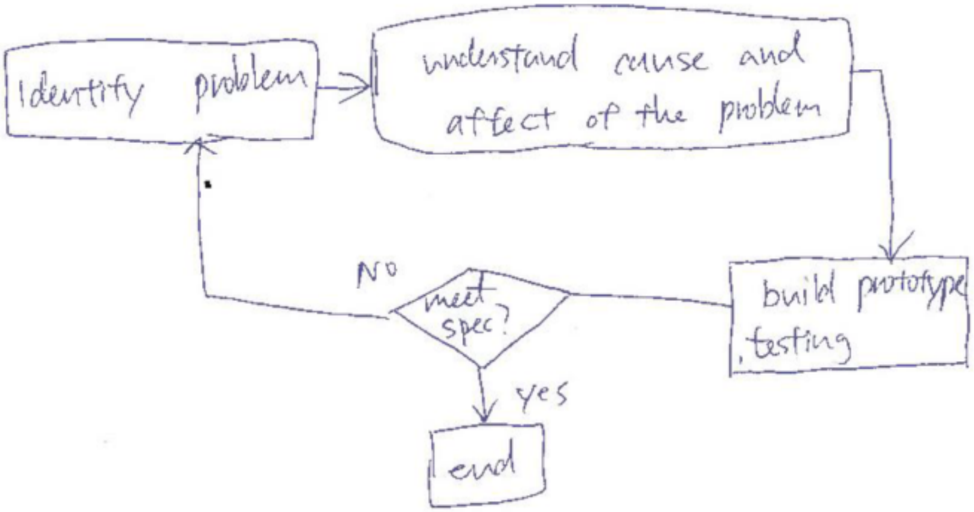


Figure 9. Student submitted Flowchart

5. Abstract Process Models

Not all process models have a straightforward structure to them. These abstract models come in a variety of different ways, from symbolic to textual. Abstract models represent the design process as not being constrained to any sort of template. If an abstract model has steps they usually are ambiguous, not in order, or extremely generic. The following models are examples of abstract forms.

5.1 List Model

A list model is a grocery-list of tasks. Tasks and goals are given in a text format such as that shown in Figure 10[20]. An advantage to this model is that before preparing a project a designer can jot down tasks he or she needs to accomplish. From there they can form a more concrete order or process. In the example below[20] the list is given several different tasks and how to accomplish them (asking, prototyping, filtering). This example allows a designer to jump to whatever task they are available to work on.

In order to generate breakthrough innovation . . .

Explore the world at large and open your mind to new possibilities
Immerse yourself in the lives of your customers
Ideate about how to challenge the status quo
Envision a breakthrough way to improve people's lives
Hypothesize a thoughtful, creative strategy and business model
Design the customer experience and supporting business model
Refine the design to make it truly breakthrough
Market your innovation to realize its full potential

Proceed through each stage of the process by . . .

Asking	Come up with the ideas that will challenge the status quo by asking the right questions.
Prototyping	Make the ideas more real—at any stage in the process—so that they can be experienced, evaluated, improved, or reconsidered.
Filtering	Challenge assumptions and conclusions to draw out the breakthrough from the banal.

Figure 10: A list model[20]

List models are great for brainstorming how to accomplish a problem. Tasks can be listed as a design team thinks of them. After the list is created the team then goes back and modifies the order and fine details of the process. The flexibility here allows list models to act as stepping stones toward other process methods.

5.2 Quadrant Map

Another form of abstract modeling is the quadrant map[20] (Figure 11). In the center of this map is a general goal (in the case of Figure 11 this goal is the intent). Along the x-axis are the extremes "know" and "make." "Know" refers to the knowledge that the design team has or discovers through research. "Make" refers to what the design team has to discover, such as concepts, prototypes, capabilities. The y-axis has the extremes "abstract and "real." "Abstract" means the intangible data such as plans or insights. "Real" concerns physical data such as research and models. A design team travels throughout the quadrant map to complete different tasks that fall into two categories (abstract and make, know and real, etc.).

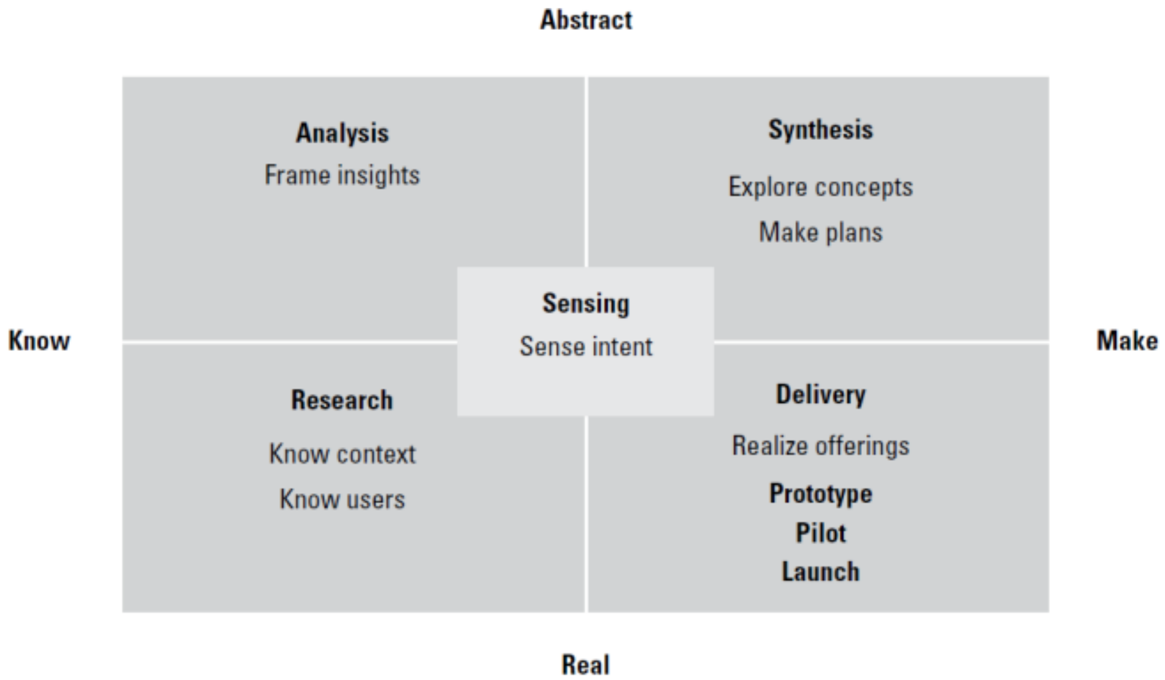
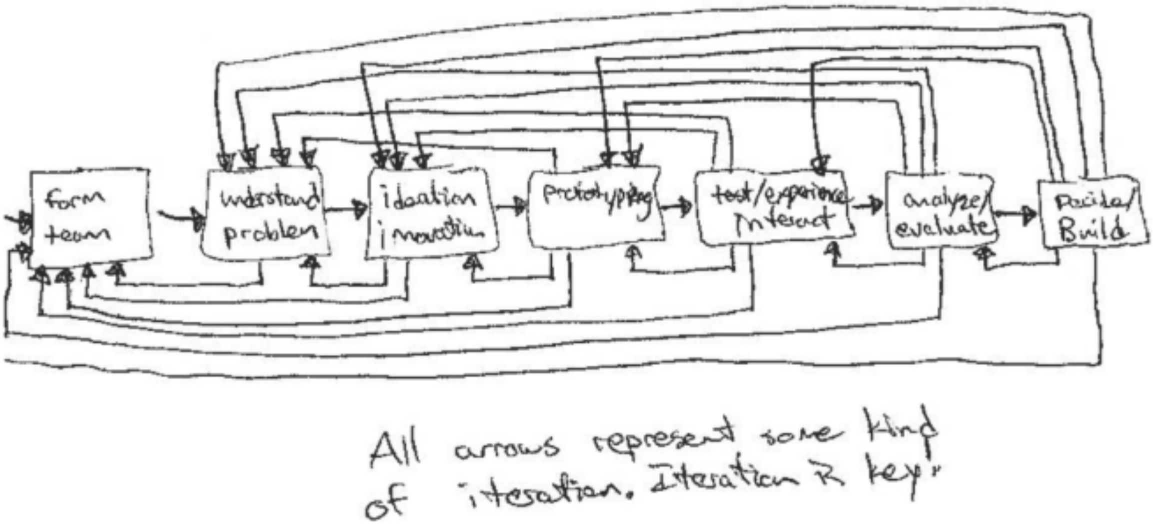


Figure 11. A quadrant map[20]

This model allows the team to see the roots and aspects of each step. Similar steps and tasks may require a particular order, or can be run at the same time by a particular design team. When designers know the aspects of certain tasks they have comparisons and a better understanding of the nature of certain tasks.

5.3 Iterative "Snake"



All arrows represent some kind of iteration. Iteration is key.

Figure 12. Student Submission, Iteration is Key

During a study to see the change of students' design processes one had submitted a model that was similar to the "waterfall" method but with iterative cycles between all of the steps (Figure

12). The student had made a note in this first submission that “Iteration is key!” This ideal reveals itself again in his/her second submission (Figure 13). The student submitted a process model showing a path between start point A and end point B. Along this path there are several iterative cycles.

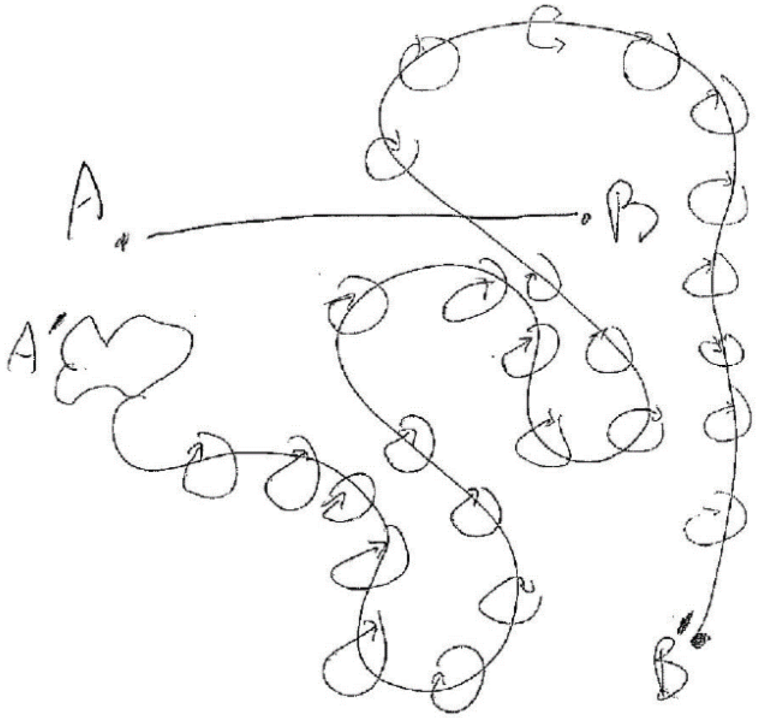


Figure 13. Student Submission, Iterative Snake

What makes this model abstract is its lack of definitive steps. The designer simply shows that the process has a start and an end, but how the user gets there is ever-changing and iterative. Concepts like these are held at high value according to some researchers[5].

5.4 Ambiguous Paths

Similar to the iterative “snake” model two students submitted design processes with little or no verbal components. These models, as pictured in Figure 14 and Figure 15, have nodes or obstacles that an engineer must navigate in order to complete a project. How a designer completes these tasks is based on their tools, knowledge, and resources. These methods often vary from one designer or design team to another, the models in the associated figures are clear to demonstrate that.

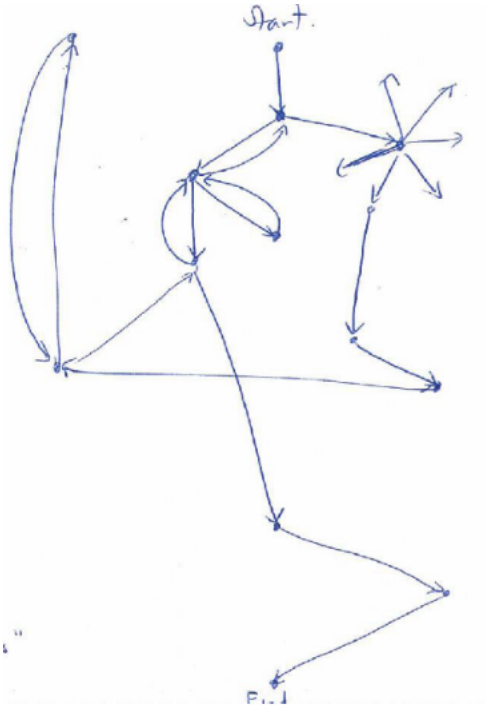


Figure 14. Student Submission, Abstract Nodes

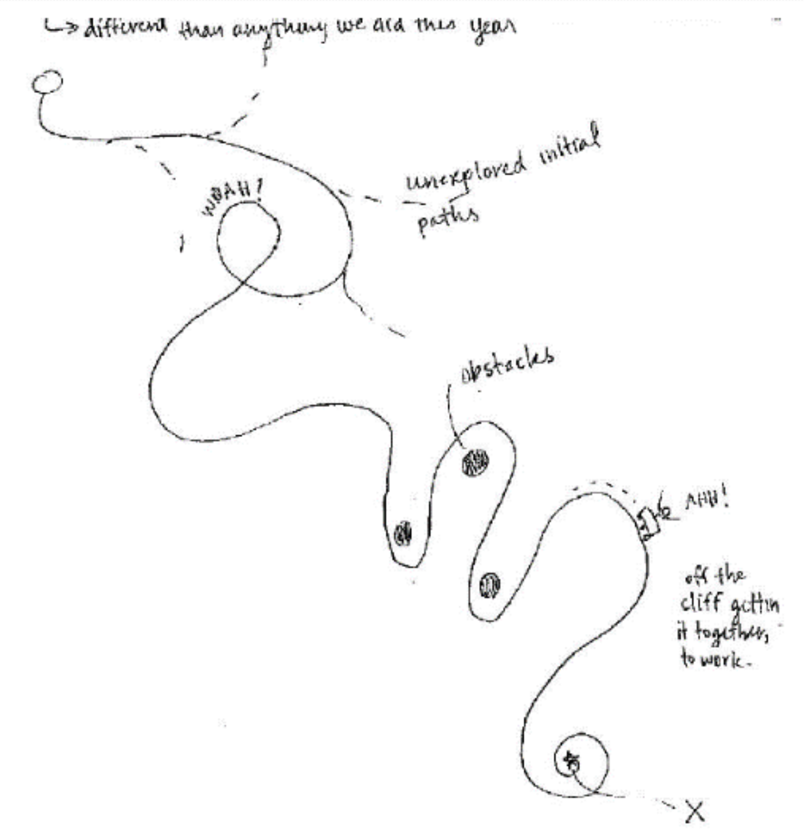


Figure 15. Student Submission, Abstract Obstacles

6 Mathematical Models

Most process models focus on how to progress through the design process via visual or verbal aid. Some models use equations and probability to determine the best or most probable route of design, these methods are known as mathematical models. An advantage of mathematical models is their ability to give insight on efficiency and outcome. These models can often be paired up with software in order to improve calculation accuracy or speed.

6.1 Markov Chains

As discussed in section 3.2, Smith and Eppinger used Markov chains to illustrate the mathematical structure behind their SIM[14]. The following equations gives time estimates for stage three of Figure 16:

$r_A = .4r_B + .3r_C + 4$ (1)
$r_B = .2r_A + .1r_C + 7$ (2)
$r_C = .5r_B + 6$ (3)

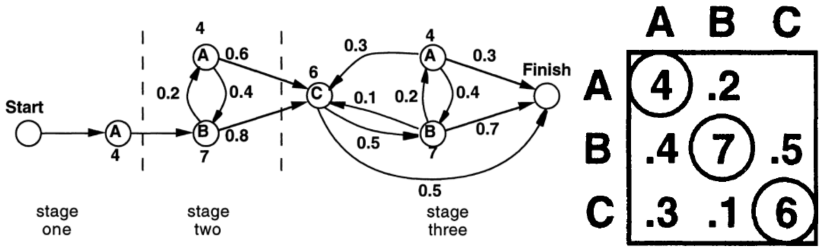


Figure 16. 3x3 Markov and the Associated SIM[14]

From Equations (1), (2), and (3) a matrix equation is created (Figure 17). Through Gaussian elimination we find the reduced matrix equation (Figure 18). Since the third stage of the chain in Figure 16 began with task C we calculate r_C from this reduced matrix. From the reduced matrix we discover that stage three takes approximately 11.21 time units.

$$\begin{bmatrix} 1 & -0.4 & -0.3 \\ -0.2 & 1 & -0.1 \\ 0 & -0.5 & 1 \end{bmatrix} \begin{bmatrix} r_A \\ r_B \\ r_C \end{bmatrix} = \begin{bmatrix} 4 \\ 7 \\ 6 \end{bmatrix}$$

Figure 17. Not Reduced Matrix[14]

$$\begin{bmatrix} 1 & -0.4 & -0.3 \\ 0 & 0.92 & -0.16 \\ 0 & 0 & 0.91 \end{bmatrix} \begin{bmatrix} r_A \\ r_B \\ r_C \end{bmatrix} = \begin{bmatrix} 4 \\ 7.8 \\ 10.23 \end{bmatrix}$$

Figure 18. Reduced Matrix[14]

The same process is repeated for stage two (Figure 19) and stage one uses only task A, which is four units. From this Markov chain we find that the total expected time is given by:

Total time = $r_C + s_B + t_A = 11.21 + 8.48 + 4 = 23.69$ (4)

$$\begin{bmatrix} 1 & -0.4 \\ -0.2 & 1 \end{bmatrix} \begin{bmatrix} s_A \\ s_B \end{bmatrix} = \begin{bmatrix} 4 \\ 7 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -0.4 \\ 0 & 0.92 \end{bmatrix} \begin{bmatrix} s_A \\ s_B \end{bmatrix} = \begin{bmatrix} 4 \\ 7.8 \end{bmatrix}$$

Figure 19. Step 2 matrices[14]

This example demonstrates how the SIM is adopted into a Markov chain to estimate the time. This method would be beneficial for discovering whether or not a team can make it to a deadline, if a new process method must be developed, or if the project must be abandoned.

6.2 Work Transformation Matrices

As briefly discussed in the end of section 3.2 "Matrices" Smith and Eppinger developed a method known as the Work Transformation Matrix (WTM) that should "predict rapid and slow convergence of iteration" and predict couples where iteration is needed[16]. This model is made with the following assumptions:

1. All tasks are done in every stage -- fully parallel iteration.
2. Rework performed is a function of the work done in the previous iteration stage.
3. The work transformation parameters in the matrix do not vary with time.

Assumption 1 refers to an ideal where tightly coupled tasks are manageable, usually with a close, fixed team. Assumption 2 is usually followed according to previous studies by Smith and Eppinger[16]. Generally if time varies (Assumption 3) it decreases rather than increases. After reviewing the assumptions Figure 20 shows the structure of the WTM.

(a) Strength of Dependence Measures **(b) Task Times**

Figure 20. Breakdown of a WTM[16]

Once set up the WTM allows equations involving matrices and eigenvalues to reveal completion time[16]. From further evaluation Smith and Eppinger give insight on how to increase iteration speed and lower unnecessary iteration. An increase in speed is accomplished through management and organization changes while the lower of unneeded iteration is achieved through team changes.

Later on Smith and Eppinger developed a subsystem of the WTM that allows designers to predict work required when assumption 1 is not followed[15]. The researchers describe the model as allowing processes like those given in Figure 21. This adjustment to the WTM gives a wider range of possibilities for engineering designers as well as more accurate estimations of completion time.

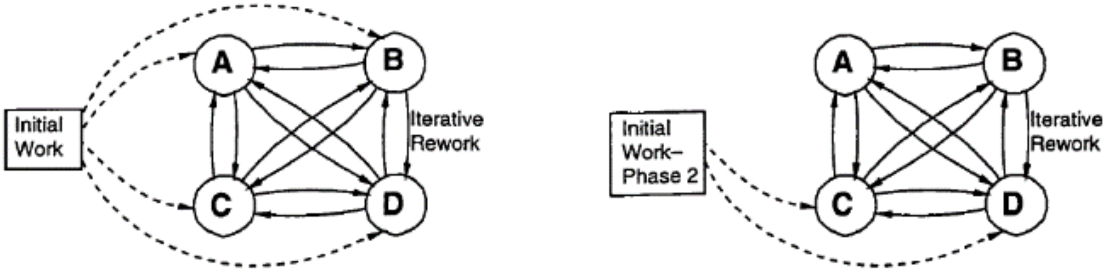


Figure 21. processes allowed by WTM extended[15]

6.3 Frequency of Reviews Method

Ha and Porteus propose a mathematical method to determine when to review a product during the design process[13]. The intention here is to realize flaws earlier in the process to prevent the need of excessive rework; however, if reviews are done too often time is lost on a project. The model proposed by Ha and Porteus works to find a middle ground and when reviews are appropriate. The model[13] takes into account variables from time spent during setup, evaluation time, and time required.

7 Analyzing the Design Models

The models discussed in this article are designed with specific intent. Some, like the DSM, are meant to cut production time. Others are meant to improve design quality by increasing iteration, like the webs discussed in section 4. What follows is an analysis of each model within its category and an overall review of each category.

These analyses will be based off of personal experience and insight, as well as observances and qualities found in other works[1, 3, 5, 17]. From Adams we learn that iteration is the key to a successful design process[3]. During one of the many studies done by Atman and other researchers the value of a student's design process is based off of how many transitions that student makes[5]. Haik proposes many forms of the design process that are cyclical, iterative, or interwoven[17]. In a study done by Smith and Tjandra[1] many of the models discussed in sections 3.2 and 6 were compared to behaviors of student design teams. Each of these sources provides different perspectives when analyzing each design process model.

7.1 Breaking Down the Structured and Step Models

The first of the structured models, the linear design processes described in section 3.1, are not seen as expert models. They received a low level of complexity due to the small amount of steps and novice level interactions. Inversely, the processes are rated high in ease of use due to the basic 1-2-3 setup that many people are familiar with. These processes are given a moderate score in appropriateness due to lack of features such as iteration and feedback. All-in-all the linear design processes were graded as moderate design processes ultimately due to simplicity.

When evaluating the DSM, SIM, and WTM processes it becomes evident that they were designed with production speed in mind. The complexity of these models is rated as high due to the time spent preparing a DSM and the complex system of interchanges. Ease of use was low because without advanced software a designer may go through several attempts to find the most simplified DSM. The appropriateness of the matrices was also graded low; this was due to the focus on turning out products in a manufacturing sense, rather than emphasizing innovation. The overall score for the matrix processes was moderate due to the required level of understanding but the lack of appropriateness.

Structured models provide a suitable design process model in many scenarios. Their high qualities include a known sense of direction and the ease of communication of plans. Such models are commonly seen in freshmen college courses, especially in preliminary activities. Large manufacturing companies may use such models to focus on production speed rather than new designs. Structured models like the Design Structure Matrix and the linear process emphasize these qualities.

On the other hand, such structured models fall short in a few important areas. Depending on their length they may become over complicated models. These structures often have a low to medium appropriateness due to their cemented or by-the-book steps. Most structured process models focus on a manufacturing sense rather than an innovation sense. The more streamlined a process is the less iteration there is, and with less iteration means little improvement or innovation during the design process. For these reasons structured models are rated at the moderate level.

Design Process Model	**Complexity**	**Ease of Use**	**Appropriateness**	**Ideal Model Level**
Linear	-1	1	0	Moderate
DSM	1	-1	-1	Moderate
SIM	1	-1	-1	Moderate

Table 2. Comparison of Structured and Step Models

7.2 Reviewing Iterative and Interactive Webs

First, we begin evaluating the PERT systems. These systems show the beginnings of excellent design process models. These systems were given a moderate in complexity due to the ability to create elaborate charts displaying multitudes of information such as the processing network in Figure 7. Ease of use was graded at high due to the easy to follow step-by-step flow of the diagrams. Appropriateness was rated at moderate because it shows the beginning steps toward the “ideal” design process we are looking for, such as iteration and communication. The overall score of PERT is moderate due to its progress towards simple setup and the focus of communication.

Continuing from PERT models we examine the flow chart and network diagrams discussed in section 4.2. Regardless of the exact type of network diagram they hold pretty well in evaluation. Complexity is given a moderate score due to the versatility of whatever the user wants. Ease of use is rated high because simple arrow and module graphics allow anyone to read the diagram to some degree. Appropriateness is also rated high because the adaptive nature of such networks allows them to be modified for almost any project. The level score of the flowchart and network design model is given high because it shows how a student or professional has evolved their design-thinking to the next higher level.

Models like the PERT diagram and the flowchart process are praised for several reasons. One of these is their adaptability; such webs are able to skip between steps in order to achieve different outcomes. They also have many areas where iteration is encouraged, often between concept generation and modeling stages. Webs are relatively easy to use as engineers basically choose how complicated they want the process to be. These models are often seen in later years of engineering students.

The value of web models has almost no downsides. The information received from these models entirely depends on the user. These models can have probability calculations added to them or a mathematical model can be used to complement the primary model. This versatility and the inclusion of key concepts are why web models receive a high rating.

Design Process Model	**Complexity**	**Ease of Use**	**Appropriateness**	**Ideal Model Level**
PERT	0	1	0	Moderate
Flow Chart	0	1	1	High
Network Diagrams	0	1	1	High

Table 3. Comparison of Iterative and Interactive Webs

7.3 Analyzing Abstract Models

The most basic of abstract models, a list model is simple, but it may not be the best of use. This model was graded low on complexity as it is just a group of tasks and goals listed down. The model was given moderate on ease of use because although it gives a lot of freedom to the design it has no clear sense of direction. This lack of direction may add confusion when trying to explain to an outside party. Appropriateness was given a low because the essential model is better suited as a complement to a design process rather than its own design process model. The overall score of the list model was low due to over simplicity and lack of coordination.

Quadrant maps also allow engineers to see information in a new way, allowing a new method of modeling. These maps were given a low complexity score as the users only need to understand the basic roots of a step to place it in the correct quadrant. The map has a moderate ease of use as there is little direction with the map, similar to the issue with the list model. Appropriateness was low because the map is not necessarily perfect on its own, and is better used as a different perspective to an existing process model. The overall score of the quadrant map model is low due to its simple and complementary nature.

The iterative "snake" model is a unique design. Its complexity is low due to its adaptability to any situation. Similarly, the ease of use is high due to the ability to plug in what steps a user needs. The appropriateness of the process model is moderate because it provides use of the iteration and cyclical thinking that many programs focus on teaching. The overall score of the iterative "snake" model is moderate as it progresses towards a model that we aim for.

Similar to the iterative snake, the ambiguous path methods allow a designer to be imaginative with how he or she accomplishes certain tasks. Their complexity is rated as low due to the ease of modifying a path to one's particular need. The ease of use is high because a particular path is only as complicated as the user wants it to be. An ambiguous path's appropriateness is given a moderate because depending on the type of path it gives a different perspective and some insight on a design problem. The overall rating of these types of models is moderate due to its general--but inventive--structure.

Overall, abstract models are relatively difficult to evaluate. Their ambiguous qualities allow them to appear with many different "faces." From reviewing abstract models a new understanding can be obtained through the different perspective. These models are useful particularly when used as complementary devices. A list model may be rather bland on its own, but it can be used early on in a design process to brainstorm what steps and methods need to be taken to complete the end goal. A quadrant map can be applied to an existing process to divide team responsibilities. The iterative "snake" and ambiguous path models offer skeletons for more specific processes. These models receive a moderate score for their ability to adapt to any required need.

Design Process Model	**Complexity**	**Ease of Use**	**Appropriateness**	**Ideal Model Level**
List Model	-1	0	-1	Low
Quadrant Maps	-1	0	-1	Low
Iterative "Snake"	-1	1	0	Moderate
Ambiguous Path	-1	1	0	Moderate

Table 4. Comparison of Abstract Models

7.4 Reviewing Mathematical Models

The Markov chain mathematical model is seen as complementary. It received a high on its complexity because as more tasks are added to the process the chain becomes increasingly complex. The ease of use was rated as moderate because as long as a user understands basic linear algebra they can understand the meaning behind the chain. The Markov chain received a moderate on appropriateness because it is useful when necessary, but isn't required for most projects. The Markov chain also works best as a complementary model, adding additional information to existing methods. For these reasons the model was graded as moderate due to the expertise required to use such a model, but the lack of application the model has on its own.

The WTM gives a different perspective to the design process, but not in an easy way to understand. This model was graded high on the complexity chart because it goes beyond that of the Markov chains discussed in section 6.1. The WTM also received a low on ease of use due to the level of mathematics required and the lengthy process to accomplish an outcome. This matrix received a low on appropriateness due to its similar focus on manufacturing as the DSM and SIM models. Overall the WTM is graded at moderate due to its high complexity and unsuitable nature.

The frequency of reviews method allows design teams to predict when to review a product ahead of time in order to catch flaws before they become overwhelming. The complexity of the model is rated at high due to the vast number of equations and iterations required in order to receive an accurate estimation. The ease of use is rather low based off of complexity and lengthiness. The method's appropriateness was moderate because such a model is innovative and may prove helpful in some applications, especially after further refinement. Overall the model proposed by Ha and Porteus was graded at moderate due to its required background knowledge but its narrow applicability.

Mathematical models allow a statistical and informative view on the design process. These models can be put to use in order to determine how likely a designer is to repeat certain steps. The models are used to determine if a system is flawed and when to check for such flaws. Other models can be implemented in order to predict production time or calculate the fastest route. The high level of insight is why engineering designers benefit greatly from these models.

Mathematical models, while beneficial, can only offer a limited amount of information. A physical web or structure gives the backbone that these models need; therefore, mathematical procedures are more useful as complements rather than standalone models. These methods are rated at moderate for their usefulness.

Design Process Model	**Complexity**	**Ease of Use**	**Appropriateness**	**Ideal Model Level**
Markov Chain	1	0	0	Moderate
WTM	1	-1	-1	Moderate
Frequency of Reviews	1	-1	0	Moderate

Table 5. Comparison of Mathematical Models

8 Choosing and Developing an Ideal Design Process Model

The primary goal of this paper was to find the most appropriate design process by evaluating various process models from a several sources: structured models[11, 12, 18, 20], iterative and interactive webs[17, 18, 20-22], abstract models[20], and mathematical methods[13-16]. Throughout this discussion key features of each type of model have been detailed. From these features the most appropriate models were highlighted. Finally, models were given an ideal model value—low, moderate, or high—to determine how ideal a design process was. The overall models are compared in Table 6. From these analyses it was determined that web models are a sign of expert level of thinking.

Design Process Model Category	**Average of Complexity**	**Average of Ease of Use**	**Average of Appropriateness**	**Ideal Model Level**
Structured and Step Models	0.33	0.33	0	Moderate
Iterative and Interactive Webs	0	1	0.67	High
Abstract Models	-1	0.5	-0.5	Moderate
Mathematical Models	1	-0.67	-0.33	Moderate

Table 6. Comparison of Model Categories

Along the way other models held key features. Mathematical models allowed insight into production time, when to review a product, and statistical aspects. Abstract models increased brainstorming and gave a new perspective to the process. Even some of the more structured models had some sort of value that could be brought to the table. After reviewing these process methods it is possible to incorporate key features into web models. Some researchers have already used mathematical probabilities in conjunction with web modeling to determine when iteration is likely to occur[22] (Figure 22). By picking and choosing select aspects or even whole processes an engineering designer can improve the value of the web process.

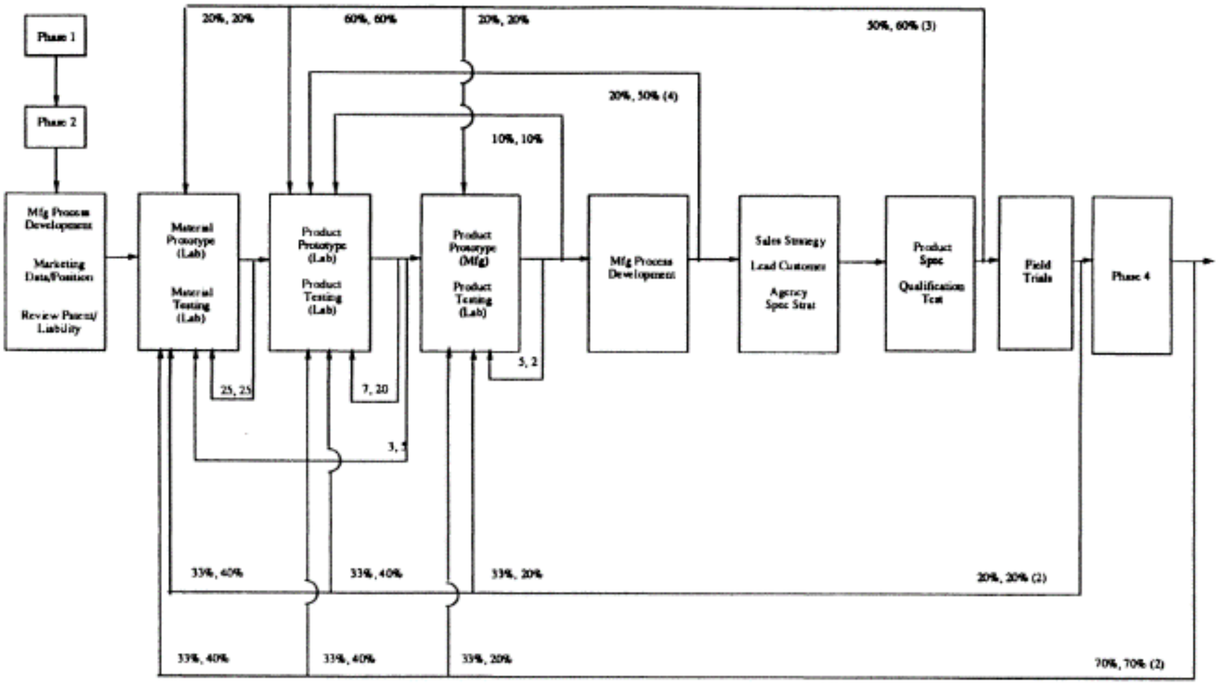


Figure 22: Incorporating probability to a flow diagram[22]

The final model proposed is very similar to the ones found in Figure 7 and Figure 22. The process has a webbed structure showing interactions and iterations throughout the process (Figure 23). In addition to the bare-bones setup mathematical probabilities can be added in order to estimate where the process is likely to go, as well as how long the process may take. In many ways the final model is just a web process. What we propose is taking this base model and implementing other models to complement it, such as abstract brainstorming and mathematical estimations.

When evaluating the proposed model based off of the same scales and sources used to evaluate the other models it scores rather well. In complexity the model scores moderate despite the amount of interwoven connections. This is due to the easy to follow arrow graphics, as well as the user's ability to omit any unneeded steps. Ease of use is rated high because of the relationship with other web models. Appropriateness is also given a high score. This is because the model highlights communication and iteration, key concepts as proposed by Adams[3] and Atman[5]. The model is also able to adapt to various situations depending on the users need; whether this includes omitting steps, adding steps, or applying other methods to complement the base model. Based off of these factors the proposed model is given a high rating for its ideal model level as shown in Table 7.

Design Process Model Category	**Average of Complexity**	**Average of Ease of Use**	**Average of Appropriateness**	**Ideal Model Level**
Structured and Step Models	0.33	0.33	0	Moderate
Iterative and Interactive Webs	0	1	0.67	High
Abstract Models	-1	0.5	-0.5	Moderate
Mathematical Models	1	-0.67	-0.33	Moderate
Proposed Model	0	1	1	High

Table 7. Comparison of Proposed Model and Model Categories

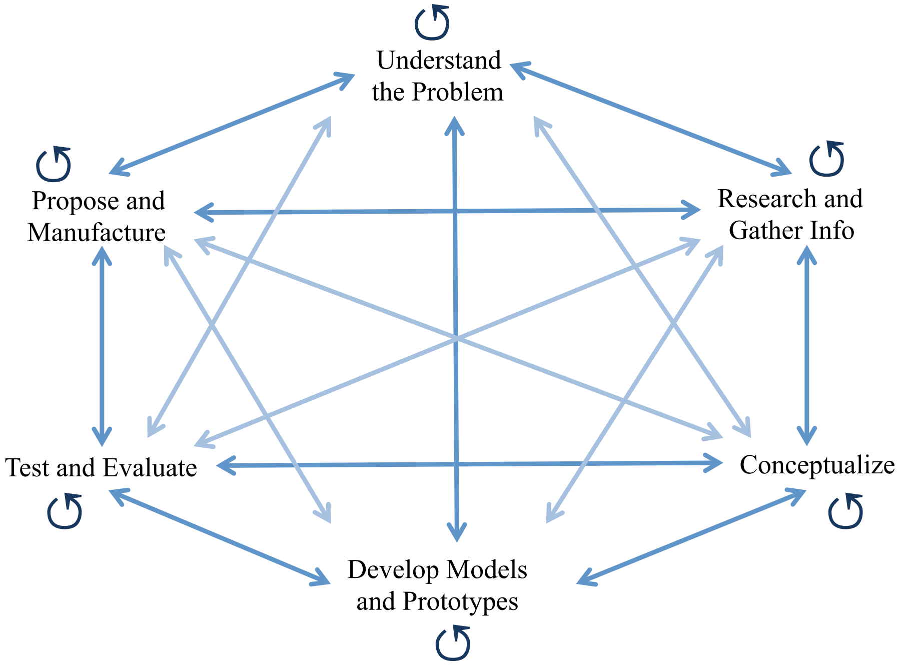


Figure 23: Proposed Web Model.

Ultimately, the way particular engineers or engineering teams complete a design is unique to their methods. The design process is a base idea that is then tailored by different people to meet different goals. Depending on the situation the proposed model may be suitable, or a new model may fit best. In the end it is the designer's choice for how they proceed through the engineering process.

9 Conclusion

Through the analysis each process model was detailed and rated on several scales. Structured models proved to be simple and linear in nature. Web models were more evolved versions of structured models, focusing on the concept of iterative processes and intercommunication between steps. Abstract models provided different perspectives and complementary additions for other processes. Mathematical models allowed technical information to be relayed from a design process to the design team. Finally, the proposed model used web models as a base structure and pulled the strengths from other types of models.

After evaluating sources and models it is apparent that there are many ways for the design process to be conducted. Some researchers emphasize the importance of iteration and communication[2-10]. Other resources look for the best way to streamline the design process[11-16]. The true purpose of engineering design process models is to communicate to workers and customers how that particular team or company operates. The proposed model in this article is a solution for just one of the many reasons a design process is needed.

10 Future Work

For future work studies could be done to expand upon the knowledge gathered here. This could include bringing in additional evaluators to rank the different processes. Studies can be done to evaluate engineering student's design processes against those presented here. In this case it may be appropriate to add additional sections or subsections of engineering models. Furthermore, through online and textual researcher other design models may be considered unique enough to add to the data found here. This study can also be used as a basis for further insight into why certain processes are considered "expert" models and others are considered novice.

11 References

1. Smith, Robert P., and Primanata Tjandra. "Experimental Observation of Iteration in Engineering Design." *Research in Engineering Design* 10.2 (1998): 107-17. *Springer Link*. Web. 28 June 2013.
2. Adams, Robin S., et al. "Educating Effective Engineering Designers: The Role of Reflective Practice." *Design Studies* 24.3 (2003): 275-94. *Science Direct*. Web. 27 June 2013.
3. Adams, Robin. "Cognitive Processes in Iterative Design Behavior." Diss. University of Washington, 2001. Ann Arbor: Dissertations, 2001. ProQuest Dissertations & Theses Full Text. Web. 25 June 2013.
4. Atman, C. J., and K. M. Bursic. "Teaching Engineering Design: Can Reading a Textbook Make a Difference?" *Research in Engineering Design-Theory Applications and Concurrent Engineering* 8.4 (1996): 240-50. *ILLiad*. Web. 27 June 2013.

5. Atman, C. J., et al. "A Comparison of Freshman and Senior Design Processes." *Design Studies* 20.2 (1999): 131-52. *Science Direct*. Web. 27 June 2013.
6. Atman, C. J., et al. "Comparing Freshmen and Senior Engineering Design Processes: An In-Depth Follow-Up Study." Design Studies 26.4 (2005): 325-57. Science Direct. Web. 27 June 2013.
7. Atman, Cynthia J., et al. "Characterizing Design Learning: A Mixed-Methods Study of Engineering Designers' Use Of Language." *Journal of Engineering Education* 97.3 (2008): 309-326. *Education Full Text (H.W. Wilson).* Web. 25 June 2013.
8. Atman, Cynthia J., et al. "Breadth Problem-Scoping: A Comparison of Freshmen and Senior Engineering Students." *International Journal of Engineering Education* 24 (2008): 234-45. *ILLiad*. Web. 25 June 2013.
9. Borgford-Parnell, et al. "From Engineering Design Research to Engineering Pedagogy: Bringing Research Results Directly to the Students." *International Journal of Engineering Education* 26 (2010): 748-59. *ILLiad*. Web. 25 June 2013.
10. Cardella, ME. "Students with Differing Design Processes as Freshmen: Case Studies on Change." *International Journal of Engineering Education* 24 (2008): 246-59. *ILLiad*. Web. 25 June 2013.
11. Eppinger, S. D., et al. "A Model-Based Method for Organizing Tasks in Product Development." *Research in Engineering Design-Theory Applications and Concurrent Engineering* 6.1 (1994): 1-13. *ILLiad*. Web. 1 July 2013.
12. Eppinger, Steven D., et al. "Generalised Models of Design Iteration Using Signal Flow Graphs." *Research in Engineering Design* 9.2 (1997): 112-23. *Springer Link*. Web. 1 July 2013.
13. Ha, Albert Y., and Evan L. Porteus. "Optimal Timing of Reviews in Concurrent Design for Manufacturability." *Management Science* 41.9 (1995): 1431-447. *JSTOR*. Web. 2 July 2013.
14. Smith, Robert P., and Steven D. Eppinger. "A Predictive Model of Sequential Iteration in Engineering Design." *Management Science* 43.8 (1997b): 1104-120. *JSTOR*. Web. 1 July 2013.
15. Smith, Robert P., and Steven D. Eppinger. "Deciding Between Sequential and Concurrent Tasks in Engineering Design." *Concurrent Engineering, Research and Applications* 6.1 (1998): 15-25. *Sage Journals*. Web. 3 July 2013.
16. Smith, Robert P., and Steven D. Eppinger. "Identifying Controlling Features of Engineering Design Iteration." *Management Science* 43.3 (1997a): 276-93. *JSTOR*. Web. 3 July 2013.
17. Haik, Yousef. *Engineering Design Process.* Pacific Grove: Brooks/Cole-Thompson Learning, 2003. Print.
18. Eastman, Charles M., et al. Newstetter. *Design Knowing and Learning: Cognition in Design Education*. Amsterdam [etc.]: Elsevier Science, 2001. Print.
19. "PERT Chart." *NetMBA*. Internet Center for Management and Business Administration, Inc., n.d. Web. 17 July 2013.
20. Dubberly Design Office, and Ryan Reposar. *[Beta] Innovation Models, Prepared for Lance Carlson of the Institute for the Creative Process, Alberta College of Art and Design*. Comp. Satoko Kakihara and Sean Durham. 1.3 ed. San Francisco: Dubberly Design Office, 2006. *Dubberly Design Office*. Hugh Dubberly, 15 Sept. 2006. Web. 15 July 2013.
21. "PERT Chart." *NetMBA*. Internet Center for Management and Business Administration, Inc., n.d. Web. 17 July 2013.
22. Adler, Paul S., et al. "From Project to Process Management: An Empirically-Based Framework for Analyzing Product Development Time." *Management Science* 41.3 (1995): 458-84. *JSTOR*. Web. 1 July 2013.
23. Koushik, Murlidhar V., and Vijay S. Mookerjee. "Modeling Coordination in Software Construction: An Analytical Approach." *Information Systems Research* 6.3 (1995): 220-54. *JSTOR*. Web. 2 July 2013.