The Engineering Problem-Solving Process: Good for Students?

Durward K. Sobek II, Vikas K. Jain
Montana State University

Abstract

As part of an ongoing effort to better understand student problem-solving processes to open-ended problems, we have coded 14 mechanical engineering projects (representing about 60 journals) according to abstraction level, design activity, planning, and reporting. We also developed quantitative outcome measures that are reported in a separate submission to this conference. We then developed a computer model of the journal data that correlates 12 key process variables to design outcomes, and conducted a computer design of experiments to extract the effects that the process variables have on the response variables (i.e., project outcomes). In this paper we report the results of this modeling effort and discuss their implications for the general model of engineering problem-solving presented in various forms in many engineering textbooks. Our results suggest modifications to the engineering problem-solving model to make it more suitable for engineering students.

1. Introduction

Solving open-ended problems is arguably the cornerstone of the engineering endeavor. Employers look for engineers who are effective at solving open-ended problems. Engineering accreditation demands evidence that students can tackle open-ended problems proficiently. Much faculty effort is devoted to improving student skills in this area. The basic process model used for these kinds of problems starts with identification of need, then goes through information gathering, idea generation, evaluation and selection steps—in other words, a basic design process model. Our three-year study of student design processes suggests that the general model for engineering problem-solving may require some tweaking to make it a more effective model for engineering students.

Over the decades, numerous models have been proposed to describe "the engineering design process." However, few of these have been empirically validated or experimentally verified. Most have been developed through personal experience and accumulation of anecdotes. Furthermore, few models explicitly consider *student* processes relative to project outcomes. Our work attempts to further our understanding of problem-solving processes by gathering data from actual projects (one in which the participants have real stakes) in large enough sample sizes to enable statistical modeling that directly links design process to outcome.

In this study, we analyzed data collected from 14 student mechanical engineering design projects, relating design process variables to project outcomes using statistical techniques. Our aim was to better understand what process characteristics tend to be associated with good design outcomes.

Specifically, we characterized the relationship between 12 design process variables (resources spent on problem definition, idea generation, engineering analysis and design refinement activities at the concept, system or detail design levels) and project outcomes as measured by client satisfaction and design quality. The key research questions addressed are: What process variables are significantly associated with positive or negative project outcomes, and what is the magnitude of their effects?

The next section provides a brief discussion of alternative methods used to study and characterize design processes and their applicability in addressing our research objectives. Then we describe our data collection and modeling methods, followed by results, discussion, and conclusions.

2. Motivation

A design process may be defined as the series of activities that take a design problem from an initial specification to a finished artifact that meets all the requirements of the specification.[1] In general, a design process can be characterized by a sequence of fundamental operations called tasks. Many authors use flowchart representations that shows discrete tasks (or task outputs) connected by transition arcs. Individual elements within the models identify tasks, procedures, or results important to the completion of the design. The overall structure of the representation provides a qualitative definition of the design process.

Design models differ widely across authors, particularly as tasks are specified in greater detail.[2] But the models consistently identify similar types of activities, e.g., problem identification and definition, ideation, evaluation, and iteration. Furthermore, most models recognize that design projects transition through phases, or alternatively, that designers operate at different levels of scope or abstraction over the course of a design project. Again, the phases or levels can differ or have different names, but most models start with an early conceptual phase, conclude with a detail design phase, and connect them with one or more intermediate phases. Figure 1 displays a typical process, adapted from Dym and Little.[3]

In our review of design texts, we were unable to identify any models that had been empirically validated or that had explicitly correlated design process to outcome. Most authors seemed to be either expert designers writing from their work experience, or academics writing from their teaching experience. Our intention, then, was to devise a study to explicitly relate process to outcome and empirically validate a general design process model derived from the literature. We hoped to gain insight into how engineering educators can better prepare their students for professional design responsibilities. The next section presents our approach.

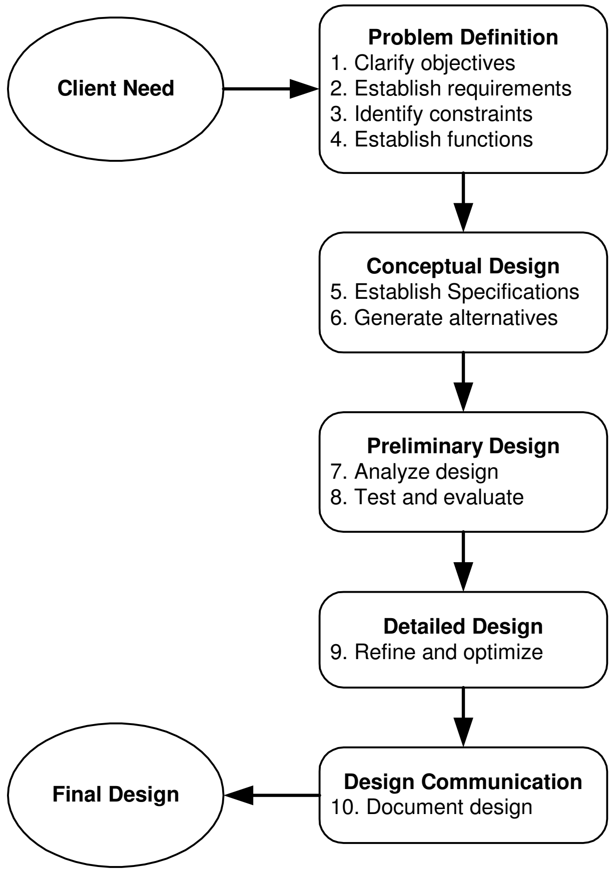

FIGURE 1: A TYPICAL DESIGN PROCESS MODEL
(ADAPTED FROM DYM AND LITTLE[3])

3. Research Methodology

This study focused on the capstone mechanical engineering design projects completed between Spring 2001 and Fall 2002 semesters at Montana State University. ME 404, the mechanical engineering capstone design class, is a 4-credit one-semester course. Students are divided into teams of 2 - 4 with a faculty member as advisor. The projects are industry sponsored so each team must interact with their client/sponsor to define the needs, devise a solution to meet those needs, and deliver a product (i.e., a set of engineering drawings and specifications, written report, oral report, and in many cases a hardware prototype) by semester's end.

3a. Data Collection: Process Variables

Researchers have used a number of techniques to collect and study data on design processes, including interviews,[1] retrospective and depositional methods,[4] protocol analysis,[5] and process observation.[6] However, for this study, we chose design journals kept by individual students as the medium by which to collect data on actual student processes. This data collection technique overcomes many of the drawbacks of other research methods. Unlike interviews, retrospective, and depositional methods, data are collected in real-time rather than retrospectively. But unlike observational approaches, the journal method does not require specially trained professionals. Like protocol analysis, the data can be readily quantified using a suitable coding scheme, but it requires little researcher intervention during data collection and therefore is a potentially more accurate representation of the actual design process. It is also more feasible to collect a relatively large sample size compared to videotaping or other approaches because the quantity of data captured, while still large, is more manageable.

Students were asked to keep individual design journals (notebooks) to document their work over the semester as a part of this project.[7] Journals were periodically evaluated using a rubric to help encourage good record keeping, and students were given specific feedback on the expectations and quality of their journals. These journals constituted 15 % of the final course grade. At project completion, journals were collected and coded according to the scheme in Table 1, with times assigned according to the start / end times recorded.

TABLE 1: CODING MATRIX

Design Activities			
	Concept (C)	**System (S)**	**Detail (D)**
Problem Definition (PD)	C/PD	S/PD	D/PD
Idea Generation (IG)	C/IG	S/IG	D/IG
Engineering Analysis (EA)	C/EA	S/EA	D/EA
Design Refinement (DR)	C/DR	S/DR	D/DR
Non-Design Activities			
Project Management	PM		
Report Writing	RW		
Presentation Preparation	PP		

Each design-related activity received two codes. The first is level of abstraction where we identify three levels. Concept design (C) addresses a problem or sub-problem with preliminary ideas, strategies, and/or approaches. Common concept design activities are identifying customer needs, establishing the design specifications, and generating and selecting concepts. System level design (S) defines the needed subsystems, their configuration and their interfaces. Detail design (D) activities focus on quantifying specific features required to realize a particular concept, for example defining part geometry, choosing materials, or assigning tolerances.

The coding scheme also delineates four categories of design activity. Problem definition (PD) implies gathering and synthesizing information to better understand a problem or design idea through activities such as: stating a problem, identifying deliverables, and researching existing technologies. Activities in idea generation (IG) are ones in which teams explore qualitatively different approaches to recognized problems, such as brainstorming activities, listing of alternatives, and recording “breakthrough” ideas. Engineering analysis (EA) involves formal and informal evaluation of existing design/idea(s), e.g., mathematical modeling and decision matrices. Finally, design refinement (DR) activities include modifying or adding detail to existing designs or ideas, deciding parameter values, drawing completed sketches of a design, and creating engineering drawings using computer-aided design software.

The coding scheme also designates codes for non-design activities associated with project management and project delivery so that every entry could be assigned a code. Project management (PM) covers project planning and progress evaluation, including: scheduling, class meetings to discuss logistics and deadlines, identifying tasks, and reporting project status. The delivery category is for activities associated with interim and final report writing (RW) and final presentation preparation (PP). Even though these activities (PM, RW, and PP combined) constitute approximately 50 % of the total project time, a separate analysis found no statistically significant association between time spent on PM, PP, and RW activities and the design outcomes (client satisfaction and design quality, explained below). Thus, this study focuses only on the design activities described in the previous two paragraphs.

The process of journal coding proceeded in two stages. First, research assistants familiarized themselves with the projects by reading the final written reports, then coded data and captured times by walking through team members' journals in lock step, considering all the members' entries for a given day before moving to the next day. Simple rules were devised for allocating time, and resolving discrepancies among the different journal accounts. The principal investigator then reviewed the coding as a crosscheck on accuracy and consistency. The disagreements were solved through discussion and the process continued until mutual agreement was reached. The time data on the various process variables was then aggregated for the project by combining individual journal data. To date, we have coded 14 design projects (approx. 60 journals). The time data on the 12 design variables (3 abstraction levels across 4 activity categories) on each of these projects served as the process/input data for the model constructed in this study (see Sobek[7] for more details).

3b. Data Collection: Outcomes Data

It seems fair to define a "good" design process as one that leads to a good outcome. Thus to determine the goodness of a design process we need a way to measure the goodness of the end product. For this study we developed two outcomes measures, client satisfaction and the quality of the final designed product. Consequently, two separate instruments, the Client Satisfaction Questionnaire (CSQ) and the Design Quality Rubric (DQR), were developed, validated and deployed for measuring the client satisfaction and the design quality index quantitatively.

The CSQ was developed by the authors based in part on previously developed surveys (for details on the questionnaire development, see the companion paper by Sobek and Jain[8]). The final questionnaire was composed of 20 questions, of which six were used for the client satisfaction index of outcomes quality, as shown in Table 2. A five-point Likert scale is used for recording the responses.

TABLE 2: CLIENT SATISFACTION METRICS

Metric	No. of Measures	Measures	Cronbach's α
Quality	2	The percentage of the design objectives the client thought the team achieved The closeness of the final outcome to client's initial expectations.	0.78
Overall	4	Design's feasibility in its application and fabrication Client's opinion on implementing the design Client's opinion on students' knowledge of math, science and engineering in developing solutions Overall satisfaction with the design outcome	0.70

This survey was validated prior to implementation using content and face validation techniques. Analytic hierarchy process was used to determine weights for the metrics and the questions in each metric. The respondents were faxed a copy of the survey, then a research assistant walked them through the questions by telephone and filled in the responses by hand. Next the survey data was analyzed for statistical reliability using the Cronbach's alpha coefficient. The test illustrated that quality and overall metric displayed adequate internal consistency and inter-metric consistency (see Table 2). As a result, the satisfaction index was obtained by the summing the weighted average of each metric. The final satisfaction scores were on a scale of 1-10 with 10 being the highest.

Since clients do not always have the background to objectively assess the engineering validity of design recommendations, we also obtained third party assessment of design quality on each project. A design quality rubric (DQR) was developed to address this issue with an objective to quantify the final "quality" of the designed projects.

To develop this rubric, we first obtained evaluation schemes from mechanical engineering capstone course instructors at 30 top ranking schools and several design contests (again, see companion paper[8] for details). We then extracted 23 metrics that were common across the evaluation schemes of the various universities and design contests. These 23 metrics were aggregated into the five metrics shown in Table 3. A seven-point scale was used for each question/metric and three anchors provided. A brief rationale was requested from each evaluator on each response for the purpose of inter-reviewer comparisons to evaluate consistency among the evaluators.

TABLE 3: DESIGN QUALITY RUBRIC

	Metric	Definition
Basic	**Requirements**	The design meets the technical criteria and the customer requirements
	Feasibility	The design is feasible in its application and fabrication / assembly
Advanced	**Creativity**	The design incorporates original and novel ideas, non-intuitive approaches or innovative solutions
	Simplicity	The design is simple, avoiding any unnecessary sophistication and complexity, and hence is: Practical Usable Reliable Ergonomic Serviceable Safe
	Overall	Overall impression of the design solution

Four engineering professionals were hired to evaluate the design projects. Three were licensed professional engineers, each with over 10 years of experience in design and manufacturing. The fourth had 5 years of mechanical engineering experience and had taken the exam to be professionally licensed at the time of the study. These evaluators were asked to evaluate the project outcomes as if they were evaluating actual industry designs while taking into consideration the project time and budget constraints. The final reports of each project served as the means for the evaluation. Specific instructions were provided to assess the design projects on their outcomes, not on the process. Each evaluator was assigned a number of reports in such a way that each report was evaluated twice to provide redundancy in the measurement. All four evaluators looked at two reports in order to determine inter-evaluator consistency. The quality index for each project was calculated by averaging the scores of the individual metrics, then averaging across evaluators. The quality score is on a scale of 1-7.

The CSQ and DQR measures demonstrate a weak correlation (0.52) implying that the two could not be combined. Therefore, to study the design processes, two models were constructed with satisfaction and quality as their respective responses. A complete description of the techniques used to code the responses, missing values analysis, descriptive question analysis, and other issues on these instruments can be obtained from Jain.[9]

<u>3c. Data Analysis</u>

The small sample size and high dimensionality of the data in this study pose significant challenges. To address these concerns, we built a principal components artificial neural network from the data currently available (so-called happenstance data). This is a special class of neural networks designed for data with high dimensionality. This hybrid architecture helped reduce the dimensionality of the data to compensate for the small sample size, and allowed us to predict the output in terms of the original process variables. Two neural network models were constructed using the twelve design-related coding pairs from Table 1 as the model inputs. One neural network used the satisfaction scores as the output variable, while the other used quality rubric scores as the output variable. A subset of the sample (11 exemplars) was used to train each network and the remaining sample was used to cross-validate the networks. Several different network architectures were constructed and trained using Neurosolutions software, with the MSE on the training and cross validation set as the judging criteria.

If the neural network is reliable (tested and validated), it should imitate the actual design processes in a manner consistent with those of our sample. Then we can use this model to generate responses in a virtual design of experiments (VDOE), and draw conclusions about the cause and effect relationships within the system.[10] We developed a 2^{12-4} fractional factorial design for each outcome measure (client satisfaction and quality), with the process variables as the input factors. The data for the runs dictated by the design grids was obtained by inputting the factor levels into the two artificial neural networks previously developed, and obtaining the predicted response levels.

Due to the deterministic nature of the neural network model, classical notions of experimental unit, blocking, replication and randomization were irrelevant in the experimental design. The final factorial was a resolution V design with 299 runs. Data transformation, model fitting, analysis of variance (ANOVA), model reduction and model adequacy checking were all performed in Design Expert software to obtain the response curves for various factors and factor interactions. Response was predicted under various process settings within the range of the data utilized to construct the model. Results of the analysis are reported in the next section.

4. Modeling Results

Table 4 reports the mean and standard deviations of the process and outcomes data used in the modeling. The process data are aggregate hours for each project, reported as a percentage of total design hours. A correlation analysis of the 12 variables found that only 2 pairs of variables out of a possible 72 were significantly correlated at 1 % significance level.

TABLE 4: SUMMARY STATISTICS

	Mean (Hrs)	Standard Deviation
Process Variables		
C/PD	13.14	9.28
S/PD	2.16	3.27
D/PD	8.68	6.10
C/IG	4.41	2.45
S/IG	2.83	1.90
D/IG	2.78	2.87
C/EA	2.94	3.82
S/EA	0.80	0.75
D/EA	24.44	16.72
C/DR	1.39	2.55
S/DR	3.54	3.48
D/DR	32.93	16.90
Outcome Variables		
CSQ	8.14	1.42
DQR	4.42	1.06

Table 5 presents the architecture summary of the two neural network models constructed. The principal components network reduced the original 12 variables to six independent components explaining 99 % of the variation in the data.

TABLE 5: NETWORK ARCHITECTURES

Parameter	Satisfaction Model	Quality Model
Number of input Variables	12	12
Number of Principal Components	6	6
Number of hidden layer	1	1
Number of hidden neurons	3	2
Training set	11	11
Testing Set / Cross Validation	3	3
Learning Rate	1.75	1.75
Momentum	0.7	0.7
Step Size	0.1	0.1
Number of iterations	1000	1000
MSE (Training Set)	< 0.01	< 0.01
MSE (Cross Validation Set)	< 0.11	< 0.21

The best performing networks (based on the judging criterion and production data) were the ones with a single hidden layer and 3 and 2 hidden neurons respectively for the satisfaction and the quality models. From the learning results, it was observed that the network architectures had a good "memory" and the trained matrices of weights and bias reflected the hidden functional relationship well. Thus the models can serve as a reasonable surrogate to reality. Finally, because the testing and validation errors (MSE) were small and the R-Sq values low, the models developed can be considered reliable for the prediction of the response scores under any combination of the process parameters as long as they are within the range investigated.

Next, Table 6 presents the analysis of variance (ANOVA) results for the satisfaction and quality virtual design of experiments (VDOE) models. The insignificant factors ($p > 0.05$) are not included. The large values of the F-ratios and the small p-values suggest that the model includes terms significantly affecting the responses.

TABLE 6: ANOVA RESULTS

Source	Sum of Squares	*df*	Mean Square	F Value	Prob > F
Satisfaction Model					
Model	206.81	25	8.27	36.32	< 0.0001
C/PD	67.32	1	67.32	295.54	< 0.0001
S/PD	19.48	1	19.48	85.51	< 0.0001
C/IG	9.66	1	9.66	42.43	< 0.0001
S/IG	2.04	1	2.04	8.95	0.0030
D/IG	6.71	1	6.71	29.48	< 0.0001
C/EA	4.50	1	4.50	19.74	< 0.0001
S/EA	6.53	1	6.53	28.69	< 0.0001
C/DR	21.87	1	21.87	96.01	< 0.0001
S/DR	3.46	1	3.46	15.20	0.0001
D/DR	15.78	1	15.78	69.27	< 0.0001
Quality Model					
Model	209.95	22	9.54	24.06	< 0.0001
C/PD	3.11	1	3.11	7.84	0.0055
S/PD	40.97	1	40.97	103.32	< 0.0001
D/PD	20.52	1	20.52	51.74	< 0.0001
C/IG	22.86	1	22.86	57.63	< 0.0001
S/IG	6.78	1	6.78	17.11	< 0.0001
S/EA	22.72	1	22.72	57.28	< 0.0001
C/DR	43.47	1	43.47	109.61	< 0.0001
D/DR	1.78	1	1.78	4.50	0.0348

Within interactions, the individual variables follow the same trend as the primary effects, save that some variables insignificant as primary effects appear significant in interactions (D/PD, D/EA for the satisfaction model and C/EA and D/EA for the Quality model).

Next, Table 7 presents an estimate of the relative importance of the significant factors in each model. The slopes of each variable versus the response variables were taken from the response plots of the ANOVA, then divided by the absolute value of the smallest magnitude slope (D/DR for both models). This calculation yields an estimate of the relative impacts that the independent variables have on the response variables. Thus, for example, in the satisfaction model, system level engineering analysis (S/EA) has an effect that is approximately 21 times stronger than D/DR, and in the positive direction.

TABLE 7: RELATIVE FACTOR SLOPE SCALING

* Insignificant at $p \leq 0.05$

Factor	**Relative Slope Estimates** *Quality Model*	**Relative Slope Estimates** *Satisfaction Model*
Concept Problem Definition (C/PD)	4.96	8.20
Concept Idea Generation (C/IG)	- 36.50	8.16
Concept Engineering Analysis (C/EA)	*	- 4.09
Concept Design Refinement (C/DR)	- 48.97	-11.83
System Problem Definition (S/PD)	40.46	9.46
System Idea Generation (S/IG)	31.61	*
System Engineering Analysis (S/EA)	114.51	21.06
System Design Refinement (S/DR)	*	- 4.13
Detail Problem Definition (D/PD)	-14.82	*
Detail Idea Generation (D/IG)	*	- 7.71
Detail Engineering Analysis (D/EA)	*	- 6.06
Detail Design Refinement (D/DR)	- 1.00	- 1.00

Table 7 indicates that conceptual and system level problem definition, and system level idea generation and engineering analysis have significantly positive impacts on project quality; whereas (somewhat counter-intuitively), concept level idea generation and design refinement, and detail problem definition and design refinement have significantly negative impacts. Concerning client satisfaction, problem definition and idea generation at the concept level, and problem definition and engineering analysis at the system level have significant positive impacts, whereas engineering analysis at the concept and detailed levels, design refinement at any design level, and detailed engineering analysis have significantly negative effects.

However, these results are only valid over the range of the variable values in our sample. For example, S/EA ranges from zero to 10 hours of activity (per project over the semester) across our sample. Thus, the results do not mean that a team should spend all of their time doing system level engineering analysis and no detailed engineering analysis. Nonetheless, even slight evidence that S/EA is two orders of magnitude more important to the design's end quality as D/DR is interesting and provocative.

5. Discussion

Table 8 below displays the general trends in the relationships of individual process variables to the two outcome measures as determined by the virtual experimental design. The plus and minus signs represent positive and negative effects of the independent variable on the response variable respectively. The left-most symbol of each pair is from the satisfaction model, while right-most symbol of each pair is from the quality model. A single plus or single minus indicates a significant factor at a 5% significance level, on the same order of magnitude as D/DR. Double plus or double negative indicates at least one order of magnitude greater impact than D/DR as reported in Table 7. Blanks denote the insignificant factors.

TABLE 8: COMBINED RESULTS

	PD	IG	EA	DR
C	+ +	+ – –	–	– – – –
S	+ + +	– + +	+ + + +	–
D	– –	–		– –

Table 8 shows a fair amount of consistency across the two models even though quality and satisfaction scores themselves were weakly correlated. Except for C/IG and S/IG, none of the variables change direction. Some of the results are somewhat counterintuitive. We will discuss them in terms of five trends evident from Table 8.

First, somewhat surprisingly, idea generation has contrasting effects on client satisfaction and design quality in this sample. Idea generation at the concept level is positive for client satisfaction, but negative for quality; whereas idea generation at the system level is negative for client satisfaction, but positive for quality. Idea generation at the detail level has either a negative or insignificant effect. Engineering design texts implore design students to "be creative" and to generate "lots of ideas," offering a multitude of techniques to break through mental blocks and come up with that real gem of an idea. So we were somewhat surprised that idea generation was not an unquestioningly good thing to do. This is possibly explained by the trend explained next.

A second trend is the overwhelmingly positive effect of problem definition activities at the concept and system levels. These activities involve such things as: understanding the client's needs, gathering additional information about the problem, searching the internet to learn about existing technologies, studying a textbook to learn about the behavior of a certain material, and so forth. In other words, problem definition as we coded it included anything where students went outside the team to gain a better grasp of the problem space. Putting this trend together with the first, it seems that it may be more productive for students to learn about existing technologies, and to learn about existing solutions to similar or analogous problems, than to brainstorm ideas. This interpretation agrees with the findings of recent studies comparing expert and novice problem-solvers. Novice designers simply do not have the repository of knowledge to draw from, so it's especially critical for them build up some knowledge in order to have a true appreciation of the possibilities. Expert designers when working in an unfamiliar domain spend a good deal of their design effort gathering information.

A third trend is that work performed at the system level appears to be very productive. The variables S/PD, S/IG, and S/EA have the most strongly positive effects in the models. The exceptions are idea generation for client satisfaction, and design refinement. System level work entails planning out the architecture of the systems, deciding upon subsystems and their configuration, working out interface issues, and so forth. Amazingly, students spend the least amount of time on these activities (9% of their design time, as opposed to 22% at the concept level and 69% at the detailed level). Also, most design texts have very little to say about system level design.

A contrasting trend is that design refinement activities are not terribly productive. The teams spending more time in design refinement tended to get worse quality and client satisfaction compared to those spending less (within the range of our sample, of course). Much of the time coded as DR was CAD work and prototype building, but it also included design changes based on new information or analysis.

A final trend evident from Table 8 is that work at the detailed level is also not terribly productive, it seems. This agrees with the general consensus in the product development literature that the early design decisions have high leverage in terms of cost and quality, and severely constrain the gains that can be made in later stages of the project. It seems design teams should be careful not to dive into detailed design too quickly.

These trends must be taken with the caveat that they apply only to the range of our data. We have no projects, for example, that spent zero time in detailed design refinement. In fact, most projects spent dozens of person-hours on design refinement activities at the detailed level. So a negative impact does not necessarily imply that teams should try to eliminate the category. Rather, the data indicate a tendency for students to spend too much time here, and so student teams would be wise to do more of their work upfront so as to decrease the amount of time needed in detailed and/or design refinement modes.

6. Conclusions

While this study is far from conclusive (data are limited to a group of mechanical engineering students at one institution in the Rocky Mountain West), it does seem to strongly suggest that the general engineering problem-solving model advocated in many introductory engineering texts should be modified or at least have a modified interpretation. We feel our data suggest at least two significant modifications.

First, while many acknowledge that understanding the problem is important, we think more is needed with an additional emphasis that is not traditionally emphasized. Since students have little experience, they would be well-advised to seek out solutions to similar problems in the past, and understand why/how they work in order to ascertain their applicability to the problem at-hand. The problem definition activity should extend to, and perhaps partially merge with the generating alternatives phase. Students need tools and appropriate representations to help assimilate information and come to a cohesive and deep understanding of the problem, and by proxy potential solutions. They need strategies that are more productive than trial-and-error, which seems to be the default strategy when faced with a new problem.

The second modification is to recognize that, particularly with complex problems, problem-solving can occur at different levels of abstraction, and that this is actually something that should be done. The typical engineering problem-solving model seems to imply that one generates alternatives, analyzes them, selects the best one, then "iterates" until done. This would seem to suggest a process whereby the young engineer generates a number of conceptual ideas, analyzes them, then selects one to detail out. Our data suggest that such a strategy is not as effective as one that includes problem definition, idea generation, and analysis at intermediate levels. It seems more work needs to be done to determine how the general engineering problem-solving model applies to complex problems.

Acknowledgements

This work was sponsored by National Science Foundation grant #REC-9984484. Many thanks to Brian Berg, Vic Cundy, Grey Lerner, Clark Mozer, Thad Orosz, Michael Wells, and the ME 404 students for their assistance and cooperation.

Bibliography

1. Johnson, E. W. (1996) *Analysis and Refinement of Iterative Design Processes*, PhD dissertation, University of Notre Dame.

2. Finger, S., and Dixon J. R. (1989) "A Review of Research in Mechanical Engineering Design, Part I: Descriptive, Prescriptive, and Computer-Based Models of Design Processes," *Research in Engineering Design*, Vol. 1, No. 1, pp. 51-68.

3. Dym, C. L., and Little, M. (2000), *Engineering Design: A Project-Based Introduction*, John Wiley & Sons, New York.

4. Waldron, M., and Waldron, K., (1992) *Mechanical Design: Theory and Methodology*, Springer-Verlag, New York.

5. Atman, C. J., Bursic, K. M. and Lozito, S. L. (1996) "Application of Protocol Analysis to the Engineering Design Process," *Proceedings of the 1996 American Society of Engineering Education Conference*.

6. Bucciarelli, Louis L., *Designing Engineers*, 1994

7. Sobek, II, D.K., (2002) "Preliminary Findings from Coding Student Design Journals," *Proceedings of the 2002 American Society of Engineering Education Conference*, Montreal, Canada.

8. Sobek, II, D. K., and Jain, V. K. (2004), "Two instruments for Assessing Design Outcomes of Capstone Projects," to appear in the *Proceedings of the 2004 American Society of Engineering Education Conference*.

9. Jain, V., K., (2003) Relating Design Process To Design Outcome In Engineering Capstone Projects, Master's Thesis, Montana State University.

10. Sacks, J., Welch, W., Mitchell, T. J., and Wynn, H. P. (1989) "Design and Analysis of Computer Experiments", Statistical Science, 4:409-435.

11. Brackin, M. P., and Gibson, J. D., "Methods of Assessing Student Learning in Capstone Design Projects with Industry: A Five Year Review," Proceedings of the American Society for Engineering Education Conference, 2002.

12. Brackin, P., and Gibson, J.D., "Techniques for Assessing Industrial Projects in Engineering Design Courses", Proceedings of the ASEE Annual Conference, Albuquerque, NM, 2001.

13. Lewis, W.P. and Bonollo, E., "An analysis of professional skills in design: implications for education and research", Design Studies, Vol. 23, No 4, pp 385-406, 2002

14. Shelnutt, Buch, Muffo, and Beasley, "Capstone Experience Draft Assessment Instruments" Prepared by a subgroup of SUCCEED's Outcome Assessment Focus Team, July, 1997

15. Spurlin, J.E., Rajala, S.A., Lavelle, J.P., & Hoskins, O.J., "Outcomes Assessment: Developing an Electronic Assessment Database as a Model for Collection and Analysis of Data", Proceedings of the American Society for Engineering Education Conference, 2002.

16. Pearson NCS, Survey Research Notes, http://www.pearsonncs.com/research-notes/subject.htm (accessed Dec. 31, 2003).

17. Saaty, Thomas, L., "The Analytic Hierarchy Process", McGraw Hill, 1980.

18. Jain, V. K., Relating Design Process to Design Outcomes in Engineering Capstone Projects, M.S. Thesis, Montana State University, 2003.

19. Yu, C. H., "An Introduction to computing and interpreting Cronbach Coefficient Alpha", Proceedings of 26th SAS User Group International Conference, 2001.

DURWARD K. SOBEK, II
Durward Sobek is currently Assistant Professor of Industrial and Management Engineering at Montana State University. He holds a B.A. degree in Engineering Science from Dartmouth College, and the M.S. and Ph.D. degrees in Industrial and Operations Engineering from The University of Michigan. His current research interests are in the areas of new product development, engineering design education, and health care.

VIKAS K. JAIN
Vikas Jain was a graduate student in Industrial and Management Engineering at Montana State University. He graduated with his M.S. degree and is currently looking for full-time employment.