Estimating Factors Affecting Project Task Size in Product Development—An Empirical Study

Katja Hölttä-Otto and Christopher L. Magee

***Abstract*—This paper addresses the methodology used to determine the amount of human resources needed to develop products. It is based on an empirical study of five firms in different industries. The methods rely on technical experts operating within budget constraints. The specific methods vary from relatively ad-hoc approaches to database-driven, centralized, and validated approaches. This variation is largely correlated with project and overall product development organization scale in a logical fashion in that larger projects and organizations move away from individual decision-makers toward centralized, objective, and validated methods. The project characteristics used by the experts and implicitly utilized in the database models to assess project complexity are also detailed in this paper. These characteristics are categorized based on a project complexity framework. We found that the scale of a project and the amount of "stretch" are the two most widely used characteristics for estimating project complexity. We found no utilization of the level of either component or task interactions in estimating project complexity by the firms studied. We also found no empirical evidence for interactions being a determinant of project difficulty despite a fair amount of theoretical concern with their importance in project complexity. In addition, we analyzed the accuracy of methods and conclude that since the accuracy at a higher level masks resource allocation problems at a lower level, the accuracy should be followed at multiple levels.**

***Index Terms*—Project complexity, project management, resource management, task size.**

I. Introduction

SINCE no two projects involving new products can be identical, estimates of resources needed are inherently uncertain. In order to execute a project successfully, a firm must be able to allocate the correct number (and type) of people to the correct projects. This paper focuses on the human resource allocation part of project management. This allocation question has direct influence on the likelihood of a given project meeting quality, time, and cost objectives.

Prior resource allocation related literature has concentrated on the stages before a project has been selected (e.g., [1]) or on how to allocate people and schedule the jobs when the sizes of development tasks are known (e.g., [2] and [3]). The project management literature (e.g., [4]–[7]) gives guidelines on how to decompose a project into tasks and how to schedule them, but this literature does not provide guidance relative to the effects the nature of tasks or product attributes have on the overall development effort, or the aspects of how many and what type of human resources are needed for a given project.

There is work on how to use a work breakdown structure (WBS) to aid in resource allocation. The WBS based estimating of the design effort, or resource need, can be based on expert judgment, alternatives analysis, published data, project management software, and/or bottom-up estimating [6]. However, the literature does not provide more detailed information on what project or product characteristics determine the design effort.

The ultimate goal of research on resource estimation is to provide methodology for more reliable forecasts of the design effort needed for a project. As a first step in such a research agenda, we conduct an empirical study in five firms concerning how they determine the number of people to deploy on their actual projects. We document here the overall procedures used and compare practices and project effort estimation at the companies. In addition, we report results on the influence of project and product attributes on needed resources.

II. Project Complexity Framework

There are a number of project complexity frameworks. For example, Shenhar and Dvir [8] developed a two-dimensional (2-D) grid for project classification based on the end product size and the level of technology. In this high-level categorization, most industrial projects, including most projects in our case study companies, fall into the middle of the grid as involving medium technology and development of a "complex collection of interactive elements," i.e., a system. This categorization cannot generally differentiate between projects within a single company.

Previous work on project complexity within a company has typically concentrated on a single dimension of complexity. For example, Liu [9] defines project complexity as the difficulty of achieving the project goals and Song and AbouRizk [10] discuss project scope complexity as a number of design items. Baccarini [11] introduces a 2-D categorization of projects, based on: 1) the variety or diversity of the tasks and 2) the interdependency between tasks or teams. Summers and Shah [12] reviewed a wide variety of complexity measures and based on this developed a multidimensional framework (Table I) to measure project complexity. They identify three aspects of a project that can be measured: the product itself (artifact), the project mission (design problem), and the tasks required to develop the product (process). In addition, they identify three metrics to measure these complexities—including the measures above: size (comparable to Song and AbouRizk [10]), interactions (comparable to Baccarini [11, eq. (2)]), and solvability (we call it stretch)

Manuscript received December 7, 2004; revised April 1, 2005 and June 1, 2005. Review of this manuscript was arranged by Department Editor J. K. Pinto. This work was supported in part by E. A. Säätiö and Tekniikan Edistämissäätiö.

K. Hölttä-Otto is with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA and also with the University of Massachusetts, Dartmouth, MA 02747 USA (e-mail: kotto@umassd.edu).

C. L. Magee is with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: cmagee@mit.edu).

Digital Object Identifier 10.1109/TEM.2005.861809

TABLE I
TYPES OF COMPLEXITY THAT AFFECT PROJECT RESOURCE PLANNING

Framework table	Size	Interactions	Stretch
Artifact Complexity			
Design Problem Complexity			
Process Complexity			

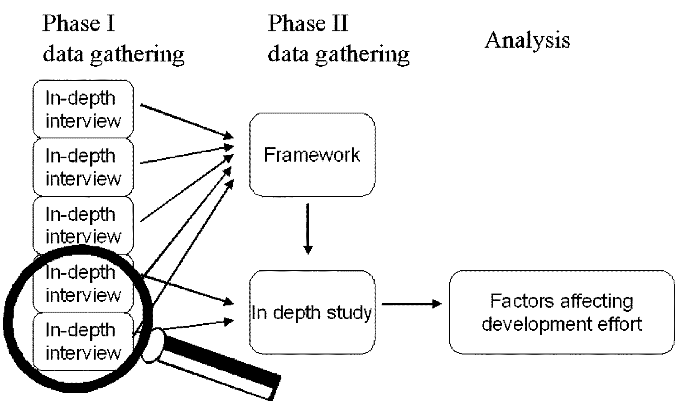

Fig. 1. Research approach.

(comparable to Liu [9]). This framework is the most complete and incorporates the other measures, thus, it is not surprising that we found this framework to be useful in understanding our empirical findings.

In addition to the above framework, we categorize the analyzed projects into platform and derivative projects [13], [14]. A platform project is defined as a project that initiates a new product family. The project can involve a clean sheet design or a significant change from a previous product. A derivative project is defined as a project, where a new version of a past product is developed.

III. Approach

This study consists of two phases (Fig. 1). The first part is a review of the current state of resource allocation and project effort estimation in five broadly differing firms. This was done in a two-step interview process: We first sent an open-ended questionnaire and after the responses to these were analyzed, we then conducted a phone interview.

The second part, based on the results of the initial effort, consists of a deeper look into the methods at two of the case companies that have the most sophisticated methods. We performed additional phone interviews and conducted more detailed on-site interviews with not only the person responsible for the projects but also the experts that provide the data for them. During the on-site visits, we also obtained detailed data for two projects from each company, including a detailed list of tasks, task properties, task sizes, estimation accuracy, etc. This allows us to ascertain the influence of certain variables on project (and task) resources which we analyze from the viewpoint of a project complexity framework.

TABLE II
FIVE CASE STUDY COMPANIES. B AND C WERE LATER CHOSEN FOR A MORE DETAILED ANALYSIS

	A	B	C	D	E
#people in PD	1500	1550	7500	215	170
#projects at a given time	100	20	155	17	40
avg project size (#people/#projects)	15	78	48	12	4
#functional areas	6	19	14	5	5
project duration (months)	2-26	12-30	21-52	12-60	24
#products sold / year	millions	2K-3K	4 500 K	1K – 100 000 K	thousands to millions
product price ($)	200-200 000	500 000-1 500 000	12 000 – 60 000	0.5 - 2000	10-100 000s
allocation method	Ad hoc	Database 1	Database 2	Project manager	Functional Manager

We combine the in depth interview data with the data of the past projects to analyze the influence of project type on the effort needed for a project, the relative importance of several task types on overall resources, and the accuracy of estimation as influenced by task hierarchy.

IV. Current State

A. Overview of Methods Used in Five Companies

In the first phase, we investigated at the current state of resource allocation and design effort estimation in five divisions of large corporations competing in different industries in the North American market. All companies develop assemblable products, not software. To maintain confidentiality, we refer to these companies as A, B, C, D, and E. Table II summarizes relevant information about the companies and the projects they pursue.

1) Decision Maker and Subjectivity of the Estimate: We identified three types of decision makers: a corporate team (C), functional units (D, E), and project managers (B). Regardless of the centrality of the decision maker a typical decision maker in a company is a technical expert, but the decision is bounded by a budget constraint.

Fig. 2 shows how the different companies map in terms of the centrality of the decision and objectivity of the methods deployed as a function of project size. The objectivity is defined as the level of personal judgment involved in the method in scale from 1 to 10, 1 being the least objective involving purely personal judgment and 10 being a structured method, where personal judgment is not needed. Larger projects tend to be associated with more central methods than the smaller projects and this seems consistent as experience of any single expert becomes more unreliable as project size increases. We also note that the centrality of deployment of a method is correlated with the objectivity of a method.

2) Project Types: Four of the five companies use the same method for all project types whether it was a derivative or a platform project. Company B has a separate method for technology development, but other than that, they use the same methodology for all project types. All five companies use the same

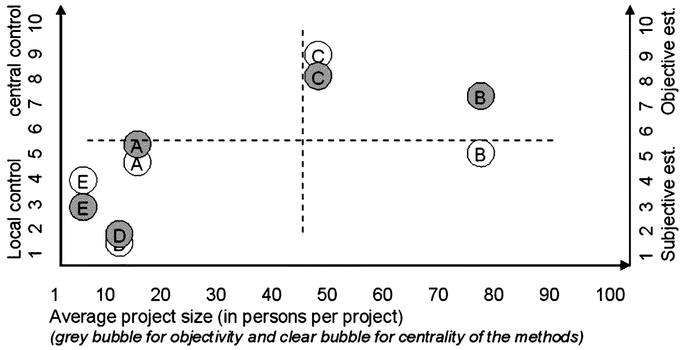

Fig. 2. Centrality and objectivity of decision making compared with the project size.

method to estimate the need for people in projects independent of the disciplines.

3) Unit of Measurement: Human resource needs are measured in time. A common way of estimating the time needed for each task is the percentage of hours in a given period. In our preliminary study, companies D and E use a rough percentage of a person year and company B a more detailed percentage of a person month. Company C has the most detailed time estimations. They use total hours to complete a task. Company A uses a unique time scale "person phase," which is analogous to a person year, but in which the time can vary in length from a few weeks to several months depending on the length of the project phase.

4) Satisfaction: We find that the two companies with the database-based methods (B, C) are fairly satisfied with their human resource allocation method, but also companies A (ad hoc) and D (project manager estimate) feel their method is acceptable, but see some need for improvement. Only company E (functional manager estimate) expresses a strong desire for a better system. It may seem surprising that no single approach appears best. However, it should be noted that the value of central and objective methods will tend to be less with small projects that have very different task content. In cases with small projects (and particularly with very differing nature of projects as seems true with company A), local experts could be the most cost-effective and reliable method available as database application would be problematic at best.

Nonetheless, the database methods developed by companies B and C are the most sophisticated methods, and the databases allow for further probing of the nature of variables affecting project resource estimation. Thus, further research with companies B and C was the second part of the research reported here. We use their databases to probe which factors (project, task, and product attributes) affect the amount of people needed for a project to more objectively assess the complexity framework presented earlier.

B. More Details Concerning Resource Allocation at Companies B and C

Companies B and C follow a generic project management process, and the human resource allocation is done in the planning phase of the process. Both companies have a three-level WBS, where the highest level is the project, then task groups or work packages, and a task is at the lowest level, and it is these lowest level tasks that are used in resource allocation database. Company B decomposes the tasks based on deliverables and company C based on subsystems and the amount of redesign they need. In the end, both approaches result in tasks. The size of the lowest level task was similar at companies B and C. The average task size at B was 7200 person hours and the median (of these lowest level tasks) was 2100 person hours showing the average is affected by a few very large tasks. Similarly, for company C, the average was 6400 hours, while the median task size was 2600 person hours.

1) Company B follows typical approaches found in the project management literature (e.g., [5]). The project managers are responsible for the project plan and resources. The project managers discuss their project with subsystem experts who are then able to give their expert estimation on how much effort a task needs.

In practice, each subsystem expert is given a project description and they make a preliminary estimate based on the past projects and their estimates of how much more or less effort the new project takes than similar past projects. Individual experts often have simple models or heuristics they use in effort estimation. These expert estimates form a lump sum of, e.g., 20 000 hours, and this is then entered into a system that distributes the hours to the tasks according to a standard distribution based on the past projects. This standard distribution is important since the resources all have a different cost. Later on, as more accurate data become available about the specific task that will have to be done, a bottom up approach can be used.

Company B starts with a rough estimate, but they have built a systematic control process to adjust the resources on a regular basis. Company B has a monthly meeting, where the top four resource usage and schedule variations from projections are analyzed carefully. Task areas using both more and less resources than planned are investigated. A typical project's resources are adjusted three times per year and the estimations are done throughout the execution phase.

Since the database at company B consists of data from past projects, it cannot be used effectively if the project is very different from any previous one by including radically new technology. In these cases, the database estimation is supplemented by expert estimation.

2) Company C has a group of resource development specialists who work together with subject matter experts from different functional areas. This group is responsible for the estimation of effort needed for each project, as well as individual task estimates. They work with a database that consists of tasks related to the components and subsystems of the product. Each task has 1–6 degrees of difficulty that affect the hours required to perform the task.

The subject matter experts evaluate the project at hand and decide which subsystems and components need to be selected for modification or development and at what level of difficulty. These tasks are selected from the database. In addition to selecting the appropriate task, the tasks can be given more or less resources by adjusting a count number that reflects how easy or difficult the task is. For example, if a design task at difficulty level 2 is usually worth 500 hours and at level 3 is 1000 hours,

TABLE III
TYPES OF COMPLEXITY THAT AFFECT PROJECT RESOURCE PLANNING

Framework table	Size	Interactions	Stretch
Artifact Complexity	B C D E		b D E
Design Problem Complexity	B C E	C	B C E
Process Complexity	A B C d	a b	A B d

Letter identifies the company, a lower case letter means company only partially in that group or moving toward it, size of the bubble relates to the size of an average project in that company.

but the experts feel that in this specific project the time required will be closer to 600 hours, the experts give the task a count number of 1.2 instead of standard 1.0. The level of difficulty and count number decisions are made between the subject matter experts and the project people.

The tasks, or subsystems to be developed or tested, in the database are common tasks done in most projects. If a totally new type of task is needed, such as a component based on a new technology, a special technology development group can add the task to the database when the technology is estimated mature enough to be included into the regular product development.

The tasks are at the third level of the WBS. In the very early phases, the estimates can be done also at the second level, in which case the second-level subsystem and system tasks consist of a predetermined combinations of third-level tasks. The estimates are reviewed by the four set check points during the project.

V. FACTORS AFFECTING DESIGN EFFORT

A. Project Complexity

In order to probe the project complexity framework introduced in Section II, we asked each company what attributes they use to estimate project complexity. These responses from the five companies are mapped in Table III.

At first glance, it is clear that the size and stretch attributes are more common foci in estimating the development effort than is the level of interactions. All companies employ measures of the number of tasks and/or the number of subsystems involved as part of their size estimation methodology. Moreover, most experts also introduce stretch into their considerations of projecting the necessary resources.

The companies with smaller projects do either artifact and design problem-based effort estimation (D and E) or process complexity base estimation (A), whereas the companies with larger projects (B and C) seem to have a larger mixture of estimation attributes. Table IV takes a closer look at the complexity attributes used by companies B and C.

A specific example of using artifact complexity and size in resource estimation involves decomposing a subsystem that is undergoing change, in order to decide what subsystems or components require redesign and what tasks, such as design and testing, each component requires.

Design problem and size are common measures at both B and C. An example is the level of customer expectations. If the customer wants a highly customizable and controllable product, it will make the design task a lot more complex than if the customer had no such expectations.

An example of design problem stretch is a case where an old customer ordered a new product that was similar to the one they had ordered earlier. The experts took a look at the last project and evaluated how similar the task would be and gave a quote to the customer. One weakness of this approach was pointed out by an expert: the estimate was done without including the suppliers due to time restraints, which may lead to problems later on.

An example of a process stretch measure is a case where the company moved from a 2-D modeling software to a three-dimensional (3-D) modeling software between projects. The new version of software had many benefits that would speed up the process over time, but after only a recent introduction to the new software, the engineers required much more hours than before for similar tasks.

In Table IV, we also show possible additional measures mainly related to interactions based on the literature. These were added in order to probe if any interactions were considered at all. The only responses to such suggestions were related to matters that cause the most rework or iteration. In software design at one of the companies, for example, the code needs to be changed as more detailed information becomes available. Changes in power requirements cause the most changes in software design according to an expert.

It is most interesting that even though interactions between teams, tasks, and subsystems are an important topic in the research literature (e.g., [3] and [16]–[18]), it appears that interactions are not considered during resource allocation in practice. This could indicate that the methods in practice could be significantly improved by considering interactions. A second possibility is that the respondents are not able to articulate the role of interactions, while such effects are actually present in the historical databases. Another explanation could be that the interactions do not differ significantly within a given system and interactions can be estimated by the size of the product. In fact, an empirical study of 14 products [19] shows that the number of parts (or artifact size) is as good an estimator of effort as the interactions within a given system. We look for evidence of interaction effects in the following section.

B. Effect of Project and Product Attributes

We analyzed two projects more deeply from each company to further examine what project or product attributes affect the design effort. For company B, we received data for two different derivative projects (projects 1 and 2). The two projects chosen from company C involve the same product family. One is a new platform (project 3) and the other a derivative (project 4) of that platform. We looked at four attributes: 1) project type; 2) task type; 3) subsystem type; and 4) functional area. The analysis is discussed in the following sections and the effects of the first three to the total resource need and the number of tasks is summarized in Fig. 3.

1) Effect of Project Type: By project type, we refer to whether the project involved a platform or a derivative design. This is related to the complexity attributes of similarity to past projects but also affects the project scale as a new platform usually involves more newly designed components. Projects 3 and 4 for company C give us an opportunity to make this

TABLE IV
EXAMPLES OF ATTRIBUTES CONSIDERED AT COMPANIES B AND C (AND OTHER POSSIBLE MEASURES IN ITALICS)

Framework table	Size	Interactions	Stretch
Artifact Complexity	End product type End product End application #parts *#lines of code* [a]	*Density of sub-system interactions*	(End application)
Design Problem Complexity	#government regulations #customer requirements Customer expectations Type of task	Tasks' effect on other sub-systems	Level of government reg. Level of difficulty Similarity to past projects
Process Complexity	#government regulations #analysis needed # of tasks *# specialties* [b] Type of analysis needed Type of testing needed	(#suppliers involved) (#distant offices involved) *Density of task interactions*	Complexity of task Similarity to past projects Analysis tool capability

[a] *see [15],* [b] *see [11]*

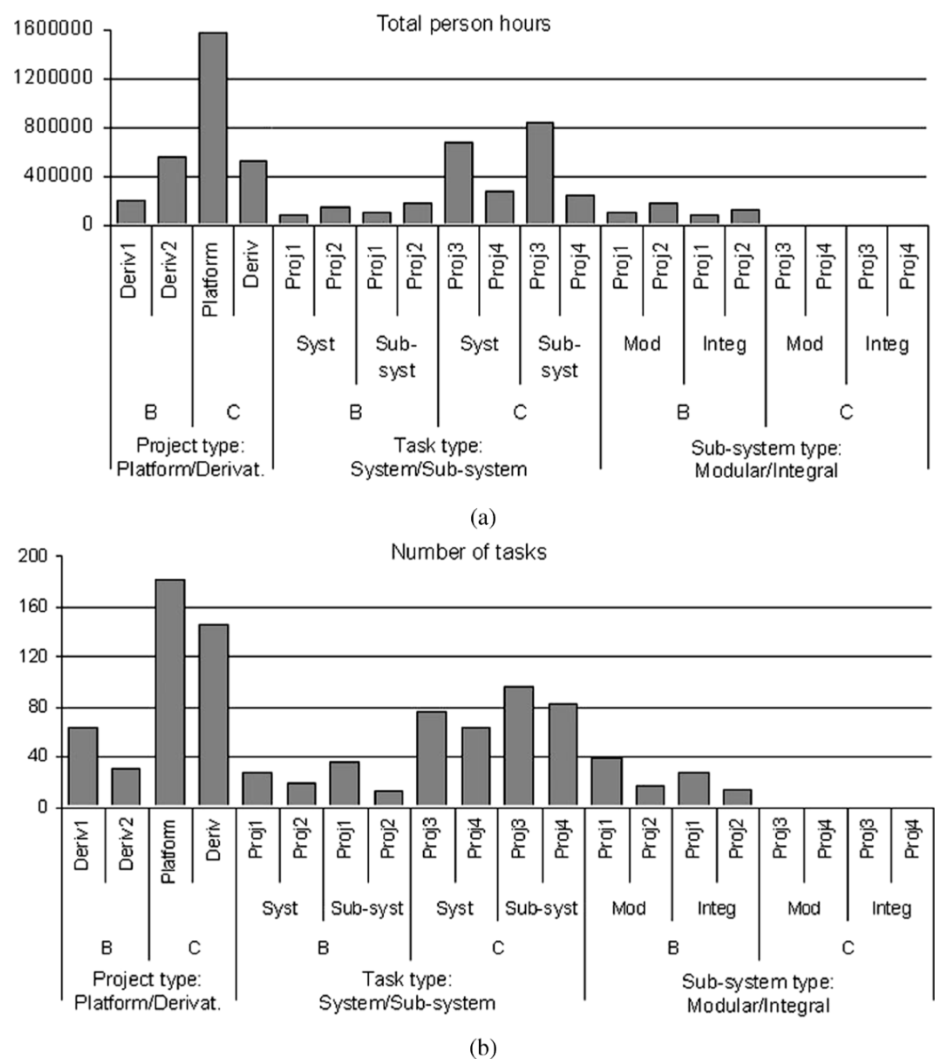

Fig. 3. (a) Effect of project, task, and subsystem types on the resource need and (b) number of tasks.

comparison directly. We find that the derivative project (project 4) requires fewer tasks and less resources (Fig. 3, left) than the platform project (project 3). We also noticed that the number of functional areas involved in the development is same for the platform project as for the derivative project.

2) Effect of Task Type: By task type, we refer to system and subsystem tasks, which we define as follows. A system task is defined as a task that involves the entire, or most of, the end product of the project. A subsystem task is defined as a task that involves a chunk of the end product and the task does not

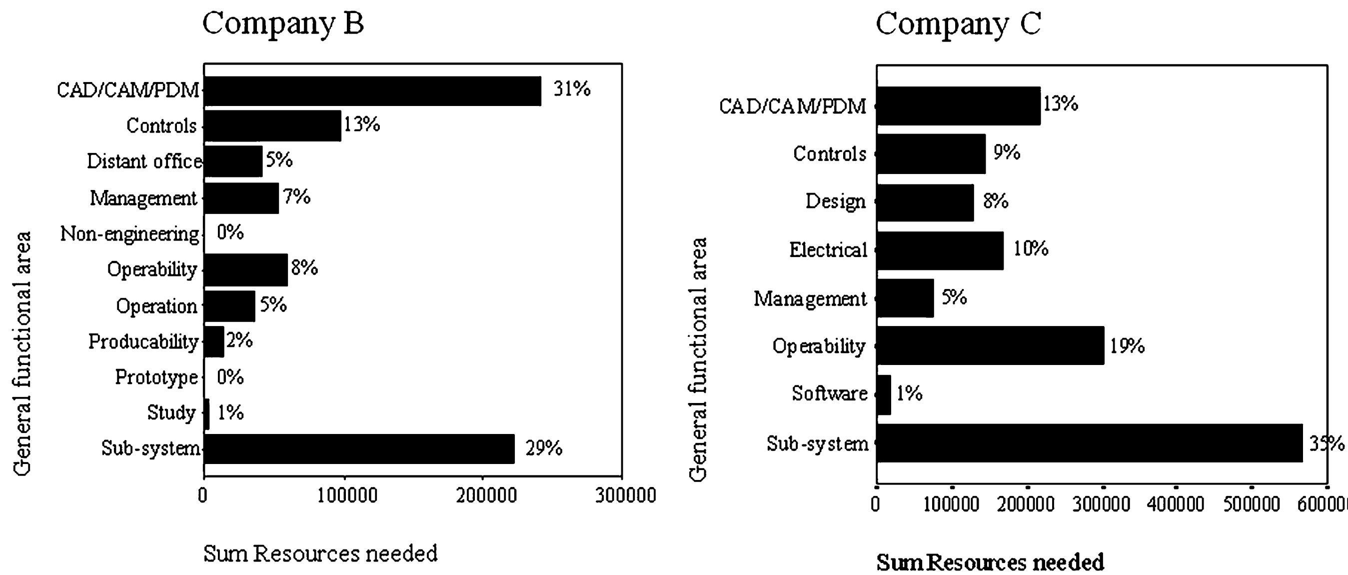

Fig. 4. Functional area type contributions to resource need (testing removed from C).

require knowledge of the entire (or most of) the end product. Thus, system tasks by definition would deal with the interaction complexity that was found above to be relatively unimportant in interview responses by the five companies.

We first looked at the total person hours system and subsystem tasks consumed in each project. The total effort for both types of tasks is shown in Fig. 3(a) (center). The ratio of subsystem to system resources is approximately the same for each project. For the most difficult project (project 3), the subsystem tasks require more effort in total than system tasks. Thus, looking more closely at the system tasks indicates (in agreement with the interviews) that directly accounting for interaction complexity may be unimportant in determining project difficulty.

Effect of Subsystem Type: The third attribute tested was subsystem type, namely, modularity or integrality. This is another mechanism that would allow interactions to be important (even if not mentioned) if we find that tasks on modular systems are significantly reduced relative to those on integral systems. In order to test this, we defined the modules for company B in the same way as in [18]. The data obtained from company C did not allow for the definition of modular and integral systems.

We find that modular systems require somewhat more resources in total than integral systems (Fig. 3, right). This is partly due to the larger number of modular tasks. In addition, we investigated whether modular system require involvement of less functional areas in the development, but again, the difference to integral systems is insignificant.

The fact that we find no clear differences in resources need for modular and integral systems is surprising in light of some previous publications [18] and [20]–[22]. It may indicate that these specific modules were designed poorly relative to ease of design. The problems may be due to the company defining the modules according to the functional areas. This can lead to excess coordination problems, as shown in [23]. However, it is also possible that for complex electromechanical products, Whitney's [24] rationale for why such products are more difficult to design to benefit from modularizing, is supported. In any case, it is important to note that the effort equality between development of modular and integral systems is consistent with the surprisingly low importance for interaction complexity now indicated in three ways in this empirical research.

3) Effect of Functional Areas: In this section, we explore the effect of variation in functional areas on resource allocation. As shown in [25], different skill types such as mechanical or electrical engineering typically require different amounts of redesign effort. We tested here if the skill types, or functional areas in this case, have an effect in the case companies.

We first look at the total person hours needed for each functional area per project. For both companies, we observe that the proportion of resources needed for each functional area is similar in both projects within a company. This indicates that certain jobs, such as CAD/CAM/PDM related work may consistently require more resources than other jobs such as producability related work.

The two companies are different and operating in different industries. This fact leads one to presume that the fractions of different types of resources would be different but interestingly many commonalities can be found. In order to compare the two companies more closely, we removed the testing hours from company C, since company B does not record them in the resource allocation database, and analyze the percentages of hours needed by different functional area types (Fig. 4).

We notice that subsystem development requires a significant fraction, approximately a third of the total resources at both companies. Of the common functional areas, management requires the least resources ($\sim$5%). Controls requires approximately 10% of the total resource at both companies. CAD/CAM/PDM shows a strong difference, but nonetheless indicates it is a significant resource need independent of the industry. The resources needed for operability, however, is different at the companies reflecting the different industries. The results suggest that the contribution of a few functional areas (management, subsystems, and controls) is constant from project to project and from company to company. Notably, the subsystem development takes a third of the resources and the

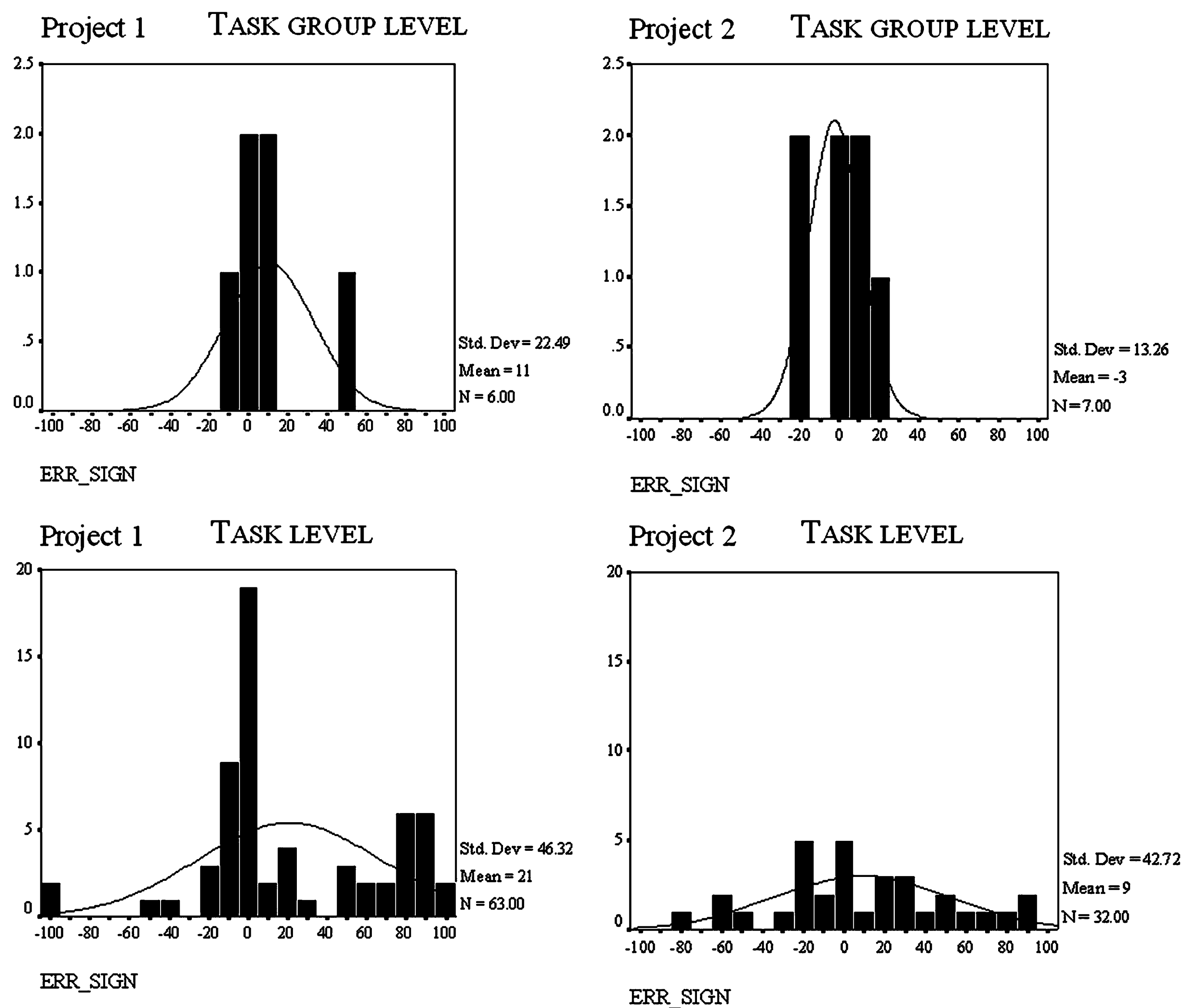

Fig. 5. Estimation error distributions for projects 1 and 2 at task group and task level.

rest go to more or less supporting functions. This can serve a rough cut estimation tool in evaluating design effort for any project.

C. Accuracy

One of the key concerns using estimates of the task sizes and the number of people needed for a project is the estimation accuracy. All five companies in the first phase say they are more likely to underestimate than overestimate the resource need. Only company B has detailed past accuracy data at the task level. Company C and B have accuracy data at the project level.

The accuracy at the project level was surprisingly uniform. Project 1 at company B at 3.2% over and project 2 at 4.2% under. At company C, project 3 was 3.7% under and the data for project 4 is not yet available. It seems that 4% is an achievable level of estimation error at the project level. However, this approximately 4% estimation accuracy at the project level does not tell the whole story, as a closer look at the accuracy at different levels of detail at company B shows. Company B records accuracy per task group and per task. In addition, they record the accuracy for each functional area at both task group and task level. The histograms (Fig. 5) show how the accuracy at task group level is much lower (~20% versus 4%) than at the project level for the same project data. Further, we see that the accuracy is even worse at the task level (~40%). Also, the distribution curve flattens as we move down to a more detailed level indicating that the deviation from an accurate estimate increases as one moves to a lower, more detailed, level. Most tasks are far from accurate even if the average is close to accurate.

The error distributions in Fig. 5 are for each task group or task, but the task group and task estimations consist of separate functional area estimations. We notice that the error, and especially the deviation, increases if the functional areas are considered separately. At the task group level, the accuracy decreased from 11% to 22% for project 1 and from 3% to 23% for project

2 as compared with not separating the functional areas. The differences at the task level for functional areas lumped or separate are also clear (21% to 27% for project 1 and 9% to 20% for project 2). This shows that the functional areas have trouble estimating their resources needed for actual projects. From the project manager's point of view, a resource is just dollars that can be moved from one area to another to improve the overall accuracy. But in reality, the resources often consist of highly specialized people who cannot be transferred from one task to another. There is a less significant difference at the functional area level between the task and task group levels. This suggests that the task group level is good enough for resource tracking purposes at the functional area level.

The fact that accuracy deteriorates when one looks at more detailed levels indicates that company B's practice of following the estimation accuracy at lower levels is appropriate.[1]

In addition to the functional areas, we checked if any of other factors that influence resources also affect the estimation accuracy. The project type, whether a derivative or new platform development project, could not be tested since company C does not have accuracy data for the second project. The tasks type's effect, whether a task is a system or subsystem task was tested. The mean error for system tasks was slightly higher than for the subsystem tasks, but the difference was insignificant. Further, we found that integral systems make the estimation slightly more inaccurate, but again, the difference is not significant.

VI. Conclusion

We found that the human resource allocation in product development is typically done by a technical expert under budget constraints. The technical experts can be supplemented by either local project managers or a central group of resource allocation specialists. Companies with larger projects tend to have a more central resource allocation methodology than firms who naturally have smaller projects. In addition, centralized methodologies are less subjective than localized methodologies and are more likely to employ a formal validation of the methodology.

We find that a central resource allocation method is somewhat preferred over a local method based on the satisfaction levels of the interviewees, but may not be effective for firms with small and disparate projects.

We identified multiple factors that experts use to estimate the development effort and project complexity. The most common factors are related to the scope or scale of the artifact, design problem, or the process (for example, the number of components, attributes, or tasks). Also, stretch (or the perceived difficulty of meeting objectives) was considered an important factor affecting the number of resources needed. Consistent with these drivers, we found that derivative projects require fewer resources than platform development projects.

Regardless of the method type in the five companies, interviews suggested that subsystem interactions are not used in effort estimation. The estimations are based on the scale and stretch of the project. We further probed for the importance of interaction complexity in driving project resources by studies of the databases employed in the objective methodologies found as part of this research. We analyzed the relative efforts between system tasks and subsystem tasks. If interactions were important, system tasks would require relatively more effort in more difficult projects. However, this was not the case as there is no difference in the proportion of development effort between the two task types as a function of project difficulty. Further, if interactions are important, modular subsystems should require less effort and fewer functional areas to be involved in the development than integral subsystems. We found that, in fact, both modular and integral subsystems require as much effort and as many functional areas. We believe this is due to three reasons: 1) the modularizing of assemblable products is difficult; 2) the modules are defined according to the functional areas and not *vice versa*; and 3) the interaction differences between the systems are not significant. The full modularization of the system in question will probably not be possible due to the first reason, but if the modules were defined based on the system characteristics, and then the development teams accordingly, more benefits might be seen. Nonetheless, the relative unimportance of directly accounting for interaction complexity is again consistent with this result however it occurs.

Within each company, each functional area requires a similar amount of resources, in respect to the other areas, from project to project. Further, attempting to compare resource allocations in differing functions across companies in very different industries shows—not surprisingly—significant variation. There are nonetheless some similarities for a few general functional areas that require the same fraction of resources independent of the company. For example, subsystem development requires about a third of all resources, and management only about 5% for projects in the two companies with quantitative databases.

Following the accuracy of the estimations is important. We noticed, moreover, that the accuracy can vary significantly from one level of detail to another. Typical inaccuracy at a project level is approximately 4%, whereas it approaches 40% at a detailed functional task level. The project level results may seem reassuring but are misleading as technical resources are not fungible and the lower level inaccuracy is what matters to project success. According to the findings in this paper, we recommend following accuracy at a task group level for each functional area. This corresponds to the second level of a three-level WBS. This level is more detailed than the project level but gives nearly as detailed data as a level below, but following at the level below would be more laborious.

From a more academic perspective, an interesting finding is that directly accounting for interaction complexity is relatively unimportant as a determinant of project difficulty/resources. This is different than expected based on prior theoretical-based work. Thus, we must be cautious in concluding that interactions are unimportant as they may still be present even though our probing of the data did not uncover evidence of them. In addition, our empirical findings only apply to the five companies studied here and further studies of the type done here will be needed to assess their generality. In the interim, it seems appropriate to consider the general possibility that directly accounting for interaction complexity is relatively unimportant

[1]Company C is starting to follow accuracy at a lower level, which indicates they also have suspicions about reduced accuracy at the more detailed task levels.

estimating development effort in real-world product development projects.

In summary, we identified multiple factors that affect the resource usage and in addition a few that do not, such as the interactions. Ultimately, we would like to be able to recommend the ideal factors to use in resource estimation for different industry environments, but at this point our recommendation is to track the factors shown useful in this research, be cautious about the interactions, and follow accuracy at a level equal to the second level of a WBS.

REFERENCES

[1] R. G. Cooper, S. J. Edgett, and E. J. Kleinschmidt, "New product portfolio management: Practices and performance," *J. Prod. Innova. Manage.*, vol. 16, pp. 333–351, 1999.

[2] T. Roemer, "Concurrent product development," Sloan School of Management, Cambridge, MA, Working Paper, 2004.

[3] A. Kusiak, *Engineering Design: Products, Processes, and Systems*. New York: Academic, 1999.

[4] J. K. Pinto, Ed., *The Project Management Institute—Project Management Handbook*. San Francisco, CA: Jossey-Bass, 1998.

[5] M. D. Rosenau, *Successful Project Management*, 3rd ed. New York: Wiley, 1998.

[6] *A Guide to the Project Management Body of Knowledge PMBOK*, 3rd ed., Project Management Institute, Newton Square, PA, 2000.

[7] R. Klein, *Scheduling of Resource-Constraint Projects*. Norwell, MA: Kluwer, 2000.

[8] A. J. Shenhar and D. Dvir, "Toward a typological theory of project management," *Res. Policy*, vol. 25, pp. 607–632, 1996.

[9] A. M. M. Liu, "A research model of project complexity and goal commitment on project outcomes," *Eng., Construct. Arch. Manage.*, vol. 6, no. 2, pp. 105–111, 1999.

[10] L. Song and S. M. AbouRizk, "Quantifying engineering project scope for productivity modeling," *J. Construct. Eng. Manage.*, vol. 131, no. 3, pp. 360–367, 2005.

[11] S. M. Baccarini, "The concept of project complexity—a review," *Int. J. Proj. Manage.*, vol. 14, no. 4, pp. 201–204, 1996.

[12] J. D. Summers and J. J. Shah, "Developing of complexity for engineering design," in *Proc. ASME DETC*, Chicago, IL, 2003. Paper DTM-48633.

[13] M. V. Tatikonda, "An empirical study of platform and derivative product development projects," *J. Prod. Innov. Manage.*, vol. 16, pp. 3–26, 1999.

[14] S. C. Wheelwright and K. B. Clark, "Creating project plans to focus product development," *Harvard Bus. Rev.*, pp. 70–82, 1992.

[15] O. Benediktsson and D. Dalcher, "New insights into effort estimation for incremental software development projects," *Proj. Manage. J.*, vol. 35, no. 2, pp. 5–12, 2004.

[16] A. Yassine, N. Joglekar, D. Braha, S. Eppinger, and D. Whitney, "Information hiding in product development: The design churn effect," *Res. Eng. Design*, vol. 14, no. 3, 2003.

[17] T. R. Browning and S. D. Eppinger, "Modeling impacts of process architecture on cost and schedule risk in product development," *IEEE Trans. Eng. Manage.*, vol. 49, no. 4, pp. 428–442, 2002.

[18] M. E. Sosa, S. D. Eppinger, and C. M. Rowles, "Designing modular and integrative systems," in *Proc. ASME*, Baltimore, MD, 2000. Paper DTM-14571.

[19] M. Van Wie, J. Greer, M. Campbell, R. Stone, and K. Wood, "Interfaces and product architecture," in *Proc. ASME DETC*, Pittsburgh, PA, 2001. Paper DTM-21689.

[20] C. Y. Baldwin and K. B. Clark, *Design Rules: The Power of Modularity Design*. Cambridge, MA: MIT Press, 2000.

[21] P. Rajan, M. Van Wie, K. Wood, K. Otto, and M. Campbell, "Empirical study on product flexibility," in *Proc. ASME DETC*, Salt Lake City, UT, 2004. Paper DTM-57389.

[22] S. Thomke and D. Reinertsen, "Agile product development: managing development flexibility in uncertain environments," *Calif. Manage. Rev.*, vol. 41, no. 1, pp. 8–30, 1998.

[23] T. D. Miller, "Modular engineering," Ph.D. dissertation, Dept. Mech. Eng., Section for Engineering and Product Development, Tech. Univ. Denmark, Lyngby, Denmark, 2001.

[24] D. E. Whitney. (1996) Why mechanical design cannot be like VLSI design. [Online]. Available: http://web.mit.edu/ctpid/www/Whitney/papers.html#P1

[25] K. Holtta and K. Otto, "Incorporating design complexity measures in architectural assessment," in *Proc. ASME DETC*, Chicago, IL, 2003. Paper DTM-48648.

Katja Hölttä-Otto received the M.Sc. (Eng.) and D.Sc. degrees in mechanical engineering from the Helsinki University of Technology, Helsinki, Finland.

During her studies, she spent 2001–2005 as a Visiting Scholar at MIT Center for Innovation in Product Development. She started as an Assistant Professor in the Department of Mechanical Engineering, University of Massachusetts, Dartmouth, in 2005. Her current research interests are product architecture and methods for product platform development including evaluation of proper degree of modularity and the fit between the organization and the platform. She is also interested in product and platform flexibility.

Prof. Hölttä-Otto is a member of the Institute for Operations Research and the Management Sciences (INFORMS), Society for Women Engineers (SWE), and the American Society of Mechanical Engineers (ASME).

Christopher L. Magee received the B.S. and M.S. degrees, and the Ph.D. degree in materials science and engineering from Carnegie-Mellon University, Pittsburgh, PA, and the M.B.A. degree from Michigan State University, East Lansing.

He is the Director of the Center for Innovation in Product Development and a Professor of the Practice of Engineering Systems and Mechanical Engineering all at the Massachusetts Institute of Technology (MIT), Cambridge. He has held these positions since January 2002. Prior to this, he had 35 years of experience at Ford Motor Company ranging from early research and technology implementation work to later executive positions in product development emphasizing vehicle systems and program initiation activities. He has been a participant on major National Research Council Studies whose topics span design research to materials research. His current research focuses on the innovation and change process in complex systems. His teaching subjects include product development, complex system modeling, and systems engineering.

Dr. Magee is a member of the National Academy of Engineering, a Fellow of ASM, and a Ford Technical Fellow.