Cost and Risk Considerations for Test and Evaluation of Unmanned and Autonomous Systems of Systems

Indira Deonandan
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA, 02139
indira@mit.edu

Ricardo Valerdi
Engineering Systems Division
Massachusetts Institute of Technology
Cambridge, MA, 02139
rvalerdi@mit.edu

Jo Ann Lane
Systems Engineering Research Center
University of Southern California
Los Angeles, CA 90089
jolane@usc.edu

Filiberto Macias
Systems Engineering Directorate
US Army White Sands Missile Range
New Mexico
Filiberto.a.Macias@us.army.mil

Abstract- *The evolutionary nature of Unmanned and Autonomous systems of systems (UASoS) acquisition needs to be matched by evolutionary test capabilities yet to be developed. As part of this effort we attempt to understand the cost and risk considerations for UASoS Test and Evaluation (T&E) and propose the development of a parametric cost model to conduct trade-off analyses. This paper focuses on understanding the need for effort estimation for UASoS, the limitations of existing cost estimation models, and how our effort can be merged with the cost estimation processes. We present the prioritization of both technical and organizational cost drivers. We note that all drivers associated with time constraints, integration, complexity, understanding of architecture and requirements are rated highly, while those regarding stakeholders and team cohesion are rated as medium. We intend for our cost model approach to provide management guidance to the T&E community in estimating the effort required for UASoS T&E.*

Keywords- Unmanned and Autonomous Systems of Systems, Test and Evaluation, Cost, Risk, Complexity

1 Introduction

The development of Unmanned and Autonomous Systems (UAS) has increased exponentially over the years and at the same time has challenged traditional testing and evaluation (T&E) processes. UAS's provide amazingly new tactical abilities to the warfighter including mission assurance, command and control. [1] In addition, they never operate alone. They are always part of an integrated and well connected network of system of systems. This means that the constituent UASs are both operationally independent (most or all of the constituent systems can perform useful functions both within the SoS and outside of the SoS) and managerially independent (most or all of the constituent systems are managed and maintained for their own purposes). [2]

The evolutionary nature and greater demand for Unmanned and Autonomous systems of systems (UASoS) acquisition needs to be matched by evolutionary test capabilities yet to be developed. For example, the Predator, which the US Air Force considers a system of four aircraft, a ground control station, and other pertinent equipment, failed Operational T&E, and still went on to be a huge success in the battlefield. Granted, T&E recommendations were overridden, and while Predator did not meet its performance requirements there is lack in T&E processes to recognize levels of effectiveness. To override T&E recommendations for every system is not advised but there is need for improved T&E processes. [1]

Testing at the SoS level focuses on the interactions between the SoS constituents and the emergent behaviors that result from the complex interactions between the constituent systems. [2] There is also no standard method to determine how much testing effort is required nor is there the ability to begin making effective contingency plans should testing requirements change. The Prescriptive and Adaptive Testing Framework (PATFrame), currently under development, uses knowledge acquisition to minimize risk through a decision support system. As part of this effort we attempt to understand the cost and risk considerations for UASoS T&E and propose the development of a parametric cost model to conduct trade-off analyses for T&E.

We use a risk and cost approach because we recognize that on a SoS level, there must be a comprehensive analysis of complexity to understand its impact on the cost of systems and to avoid unreliable estimates and unfavorable system performance. This process can also produce strategic options to improve the confidence of cost estimators and stakeholders in making better decisions, even in the face of complexity, risk, and uncertainty. [3]

Developing any cost or resource estimation model for T&E requires a fundamental understanding of existing cost estimation techniques, how they have evolved over the years and how they can be leveraged for the purpose of T&E of UASoS. This paper focuses on understanding the need for better estimation of the test effort for UASOS, what cost and risk considerations must be addressed

DOI 10.1109/SOSE.2010.448

specifically for the UASoS T&E and how other approaches may be limited in addressing the specific issues of T&E of UASoS. The work presented here is a combination of information collected from various normative and descriptive views of testing based on literature review, surveys, and interviews with members of the DoD community, both those directly and indirectly involved in the T&E process. We discuss in more detail the risks and costs we have identified, and explore the ways our findings can be merged with current cost estimation processes. We intend for our approach to provide management guidance to the DoD T&E community in estimating the effort required for test and evaluation of inter-related unmanned and autonomous systems in the context of SoS.

2 T&E of UASoS

UASoS represent a new type of technology with a new engineering genus. They offer the flexibility for additional capabilities, which manned systems or SoS are not capable of due to combined safety and effectiveness considerations. Autonomous, intelligent systems, and the operators within the SoS network, will execute outside of predictable, stable behavior within carefully optimized situations. In order to be useful to the warfighter, a UASoS must have the capacity for adaptation and change and be able to perform the unexpected no matter what mission it has to perform.

2.1 Risks Associated with UASoS

UASoS provide new challenges, which dictate very different developmental and operational testing than that required for traditional systems. Currently, systems designed under traditional means are expected to perform predictable tasks in bounded environments and are measured against their ability to meet requirements, while UASoS function and operate in open, non-deterministic environments and are more focused on interactions between components, both manned and unmanned. The interconnected parts have more properties, and control and operator interfaces can be drastically different.

There are a number of other technical and organizational risks associated with testing UASoS, which we have divided into two categories: those which occur at the SoS level and those which affect the testing process. Many of the SoS level risks and challenges have been identified by numerous researchers in the past; however, none of these past projects have focused specifically on the risks associated with testing SoS. [4, 5] These risks are prioritized based on stakeholder responses in Section 4.

System of System Level Risks: Many factors can increase the integration complexity of the SoS including the number of systems to be integrated, number of interfaces involved and technology maturity of the SoS. Many UASoS have never even existed in the past making it very difficult to predict any emergent properties. A UASoS requires the ability for manned and unmanned systems to co-operate with each other to fulfill its purpose. In addition, the number of requirements of the SoS is a key driver of risk, as well as changes in requirements throughout SoS development and operation. Many times it is unclear what the SoS needs to do in order to fulfill its mission and without the appropriate metrics to evaluate the performance of the UASoS, it is difficult to determine whether the mission is successful or not. But, not only do requirements change within a mission setting; missions and operational platforms also change resulting in changing requirements to reflect the warfighter's needs. For example, the Brigade Combat Teams Modernization Program, formerly known as Future Combat Systems, is a typical SoS integrating a number of operational platforms, a versatile mix of mobile, networked systems that will leverage mobility, protection, information and precision. To conduct effective operations across such a spectrum requires careful planning and co-ordination of space, air, land domain platforms and networks, understanding of the SoS architecture and capabilities, as well as interoperability across all components of SoS. Further, the individual systems within a SoS may have varying levels of maturity and may enter the SoS at different stages of the SoS lifecycle. Ensuring that these systems can still work together and merging newer more advanced technologies with more traditional technologies can present a significant challenge to development and validation of the SoS. In addition, the degree of autonomy of the individual systems can result in cost savings in some areas and additional costs in other areas. From an operational perspective, it may be less costly to operate a set of systems with a higher degree of autonomy because the systems are more developed and capable of fulfilling the missions while maintaining safety to the warfighter. From the development perspective, the higher the degree of autonomy, the more significant the costs especially when ensuring that the UAS meets its specified requirements and is capable of maintaining the safety of the warfighter.

Testing and Network Risks: Unmanageable combinatorial problems can result when a large number of tests need to be performed on a large number of systems, and especially in the DoD, there is need to prioritize tests to ensure the systems meet schedule requirements. The type of test and amount of each type of test to be performed will also be a driver of costs. For example, field tests require considerable resources, labor, and scheduling, and is significantly more costly than a simulated test which can be done in a virtual environment. While it is impossible to eliminate all risks through computer simulations; the more failure scenarios that can be predicted and tested in a simulated environment, the less costly it will be during the fielding process, especially in the case of communication failures and loss of equipment. Multisite coordination for testing also becomes an issue especially when multiple stakeholders are involved and individual systems are located in many different places. Testing systems in specific domains can also be difficult especially in the space and undersea arenas which are primarily UAS environments and access becomes exponentially more difficult and expensive. As UAS are merged with other systems to form SoS, there is need for a better understanding of the risks associated with testing in multiple domains as well as the platforms necessary to ensure effective testing in space, air, land, sea and undersea domains at once. When systems are integrated, it is difficult to predict how the test process needs to adapt to account for emergent properties, especially when dealing with UAS as these place additional demands on effort, scheduling, and budgeting. The maturity level of the test which defines how mature the test and test process are, can influence the ability of a test to predict whether the SoS has met its expected requirements and capabilities.

2.2 Future of T&E of UASoS

Testers and evaluators have much work to do develop test procedures, develop test facilities, and develop evaluation methods and criteria to address the unique characteristics, operation, and missions of unmanned systems. There needs to be a move away from traditional boundary lines placed between developmental testing (DT) and operational testing (OT). Currently, developmental testing entails identifying the technical capabilities and limitations, assessing the safety, identifying and describing technical risk, assessing maturity, and testing under controlled conditions. Operational testing is a fielding activity, measuring the capability of the system to perform in the field. [6] On a SoS level, especially with a UASoS, both these spectrums of testing are required simultaneously and even existing programs currently do joint integration testing emphasizing the need for the involvement of both operational and developmental testing throughout the life cycle of the UASoS.

Testing is a method of risk mitigation: but we must know why we are testing to determine what we should test, how long and how much effort is required. By using a risk based testing approach, we identified the risks that need to be mitigated, and what priorities need to be made in the testing process based on these risks. We will combine these results into a cost model, which we will use to estimate the amount of test effort required for a given level of confidence. The remainder of this paper focuses on our progress thus far in developing the model, based on inputs from stakeholders within the DoD and other cost estimation techniques.

3 The Cost Estimation Approach

A parametric cost model is defined as a group of cost estimating relationships used together to estimate entire cost proposals or significant portions thereof. These models are often computerized and may include many interrelated cost estimation relationships, both cost-to-cost and cost-to-non-cost. [7] The use of parametric models in planning and management serves as valuable tools for engineers and project managers to estimate effort.

While cost models have not been specifically applied to testing and evaluation in the past, they have been an essential part of DoD acquisition since the 1970s. Hardware models were first to be developed and were followed by software models in the 1980s. Various cost models have subsequently been developed to focus on specific categories of systems; however none of them have been singled out for the testing and evaluation phase of the system life cycle. In fact, previous studies on systems engineering cost models have shown that developers are so convinced that T&E is such a small proportion of the total life cycle cost, that much more emphasis is placed on the cost of the other phases of the life cycle as opposed to T&E. [8] However, further analysis of T&E in the SoS environment with recent reports of unexplained behaviors in complex systems (e.g., Lexus cars speeding out of control) are leading experts to re-evaluate these ideas. [9]

From a warfighters' perspective, testing UASoS is absolutely critical and in fact because many of these systems are being fielded for the first time and testing is so integrated with both development and operations, T&E contributes significantly to the cost of the system especially given the risks and uncertainties associated with UASoS. This coupled with the fact that during the initial testing, finding and fixing problems after delivery is often 100 times more expensive than finding and fixing it during the requirements and design phase, makes an accurate cost estimation tool even more necessary in the DoD. [10]

The budget, both in terms of cost and effort, is currently determined based on similar projects that have been conducted in the past, coupled with extrapolations to account for the new system under test. However, UASoS do not have a significant history, but are in such high demand that there is the need to understand how much effort is required for testing.

Testing is often reduced to a purely technical issue leaving the close relationship between testing and business decisions unlinked and the potential value contribution of testing unexploited. [11] There comes a point at which the amount of effort invested does not minimize risk at a justifiable rate. Neither does it offer enough of a return on the amount of resources invested into the test. We take into consideration the notion that all risks can never be eliminated, though a valuable effort can be made to significantly reduce the impacts of the risks of UASoS. We are developing a methodology to determine what the optimal test stopping point should be and how to estimate when that stop point has been reached.

3.1 Related Cost Modeling Work and their limitations for UASoS T&E

Today, there are fairly mature tools to support the estimation of the effort and schedule associated with UASoS T&E. For software development activities, there are the COCOMO II, Cost Xpert, Costar, PRICE S, SLIM, and SEER-SEM cost models. At the single system level, there is the Constructive Systems Engineering Model, COSYSMO, to estimate the system engineering effort and for definition of the SoS architecture, the solicitation and procurement process for the SoS components, and the integration of the SoS components into the SoS framework there is the Constructive System-of-Systems Integration Cost Model, COSOSIMO. [9]

But, while COSOSIMO addresses the development of a SoS and normative integration and testing in the SoS environment, there has been little work done with respect to the needed evolution of SoS T&E (prescriptive) or the evaluation of the flexibility and emergent behaviors of complex systems and SoS (adaptive limits). How do you know when testing is done and you have minimized sufficient risk so that the SoS is safe for deployment in the field? Li et al propose a value-based software testing method to better align investments with project objectives and business value. [11] This method could provide decision support for test managers to deal with resource allocation, tradeoff and risk analysis, and time to market initiatives and software quality improvement and investment analysis. While Li's value based testing techniques do give a good foundation on which we can build our methodologies, this method is more applicable for business critical projects focused on return on investment and not suitable for safety critical domains. It also requires detailed cost estimation to assist the test planner and does not account for emergent properties as those frequently found in UASoS. From a warfighter's perspective, a risk based testing approach may be more relevant as it focuses resources on those areas representing the highest risk exposure. Li also applies a costing methodology which defines costs of tests relative to each other as opposed to the absolute cost of test. PATFrame methodology attempts to calculate the absolute cost of test rather than relative cost because this will allow us to estimate and predict what strategies can be used to optimize the test process on a case by case basis.

In a paper entitled "Managing your way through the integration and test black hole", George also tries to address both testing and integration from a software perspective. [12] She claims that the integration testing phase is a black hole, which the systems never seem to escape. George calculates integration effort as a product of the number of predicted defects and the average time to find and fix a defect plus the product of number of test cases and the average time to run a test case. While this is a very simple model and could be expanded to other phases of a life cycle as opposed to just software testing, it assumes that the main problem with integration testing is defects. We build on George's process, as using only defect analysis can be very limiting since there are a number of other cost drivers which define the stopping point of a test. In fact, in a recent workshop, representatives from the army indicated that "defects" are not of that much of a concern in the SoS environment, but rather identification and evaluation of emergent behaviour is of more importance.

George also assumes that we know what these defects are, can find them easily, and can estimate the amount of effort to remove the defects. For UAS, we need to not only be able to identify and understand these single-system defects but also have a firm grasp of the risks involved in integrating multiple UAS to form a complex system of systems, and determining the cost drivers associated with those risks.

We also intend on expanding on the fundamental methods presented by Aranha and Borba to include the complexity and sizing of tests for UASoS. Their work attempted to estimate the size of a software test which is required to determine the test execution effort. This is because test managers have difficulties using existing cost models, since the effort to execute tests are more related to the characteristics of the tests rather than characteristics of the software. Their method focuses on using the specifications of the test to determine the size and complexity, which is used as an input for test execution effort estimation models. [13] Such methodology is very relevant to us because as a UASoS increases in size so does the testing complexity and thus the required test effort. We begin with the UASoS and calculate the test effort based on the complexity of the SoS.

However, whereas test size is defined as the number of steps required to complete the test, and the complexity is defined as the relationships between the tester and the tested product, we must take many more factors into consideration to determine the size and complexity of UASoS. These range from the number of requirements of the SoS, to the interactions between individual systems, individual systems at various levels of maturity, operation platform diversity, maturity level of the test given emergent UASoS, etc. There are also organizational factors that can increase the complexity of the interactions between systems, including understanding of the integration requirements depending on how well defined they are, the number of organizations or individual stakeholders managing the systems, understanding the overall architecture of the SoS etc.

3.2 Cost Model Development Methodology

To derive good cost estimating relationships from historical data using regression analysis, one must have considerably more data points than variables; such as a ratio of 5 to 1. [7] It is difficult to obtain actual data on testing and evaluation costs and the factors that influence these costs especially when programs of record do not exist. Therefore, we have embarked on the Seven Step Modeling Methodology created by Barry Boehm and used for a number of cost estimation models [14]:

Step 1: Analyze existing literature
Step 2: Perform behavioral analysis
Step 3: Identify relative significance
Step 4: Perform expert judgment, Delphi analysis
Step 5: Gather project data
Step 6: Determine Bayesian a-posteriori update
Step 7: Gather more data, refine model

For Steps 1 and 2, we have used the interpretivist approach, which focuses on complexity of human sense making as the situation emerges. It enables us to learn as much as possible about UASoS T&E and arrive at qualitative conclusions as to the most important factors. The interpretivist approach was used when developing the size and cost driver definitions with the PATFrame group and affiliates. Through a series of interviews, surveys, and working group meetings we identified and defined the most significant drivers of cost. We had to ensure that there was credibility in establishing a match between the constructed realities of UASoS T&E and the respondents, and confirm that these cost drivers were grounded in the theory of cost estimation as well as testing and not just a product of our imagination. Steps 3 and 4 are ongoing, and the remaining steps will be tackled as the project progresses.

We have begun collecting the opinions of experts involved in the T&E of UASoS on the initial technical and organizational cost drivers we identified as inputs to our cost model. Everyone we interviewed or solicited ideas from has been involved in the T&E process for at least 10 years, either as a tester, test engineer, test planner, evaluator or program manager. We have received the responses from 10 survey respondents so far. They were asked to rate the identified risks on a scale of 1 to 5, with 5 having the greatest impact on the effort required for UASoS T&E and 1 having the smallest impact. We gathered their inputs to help prioritize our cost drivers, which are a combination of factors affecting SoS, individual systems and the testing process. In addition, we gathered feedback on what drivers need to be changed, reworded, eliminated or added.

4 Cost Driver Prioritization

The following charts represent the inputs of subject matter experts in the area of testing and evaluating unmanned and autonomous systems of systems. A score that is 3.5 and above represents a high impact driver, 2.5 to 3.49 represents a driver of medium impact and a driver with a rank below 2.5 is a low impact driver. Figure 1 shows the responses to the technical drivers presented to respondents and the average score rating for each driver.

Our results confirm our hypothesis that the T&E community prioritizes tests based on how complex the task is. Number of systems, integration complexity, number of requirements, technology maturity, synchronization complexity, requirements changes test complexity and diversity are all rated very high in their impacts on effort for SoS testing. Power availability was rated with least impact and conversations with respondents confirm that power issues can be easily remedied as opposed to the other factors that need to be considered. Additional cost drivers identified include emergent behaviours, data analysis tool capabilities and instrumentation requirements and changes.

Figure 2 shows the responses to the organizational drivers presented to respondents and the average score rating for each driver.

From the organizational perspective we see that understanding of the SoS requirements and architecture as well as the personnel availability and capability are rated as higher cost drivers compared to multisite coordination of stakeholder team cohesion. "Time constraints" is the most significant organizational driver of cost in T&E of UASoS.

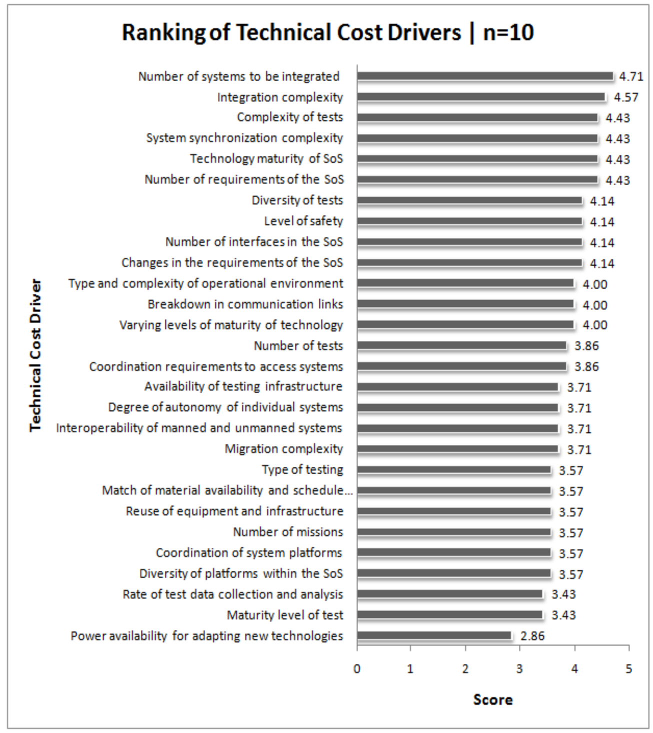

Fig. 1 Prioritization of Technical Cost Drivers for UASoS T&E

Fig. 2 Prioritization of Organizational Cost Drivers for UASoS T&E

5. Conclusion

In this paper, we have addressed the need for better testing and evaluation tools of unmanned and autonomous systems of systems and presented the Prescriptive and Adaptive Testing Framework as one of the tools which will take into consideration the risks and costs surrounding the complexity of emerging UASoS. There comes a point at which the amount of effort invested in testing to minimize the risk of the SoS is significantly greater than the rate at which risk is minimized and does not provide enough of a return on the amount of resources invested into the test. Therefore it is necessary to understand how much testing is enough to deem a SoS safe. The work presented here is just the beginning of the development of a cost and risk model for UASoS T&E to address this problem. We highlight the limitations of applying existing cost models to the testing process of UASoS and propose a new model which takes into consideration both risks and costs involved in UASoS testing. One of the important elements is ensuring that the definitions of the drivers are consistent so they can be rated from similar perspectives. We presented the prioritization of both technical and organizational cost drivers. We note that all drivers associated with time constraints, integration and complexity, understanding of architecture and requirements are rated highly, while those regarding stakeholders and team cohesion are rated as medium. We will continue to develop this model based on continued investigation into the test and evaluation process of the DoD, the inputs from subject matter experts, and the quantitative and qualitative data from ongoing projects to be used for our parametric model development. In future work, we will assign appropriate ratings to the cost drivers as identified by our testing experts in the DoD field, and present a cost model to determine the effort required for testing.

6. Acknowledgements

This material is based upon work supported by the Department of Defense, United States Army, White Sands Missile Range, NM under Contract No. W9124Q-09-P-0230. The authors would like to thank the Test Resource Management Center (TRMC) Test and Evaluation/Science and Technology (T&E/S&T) Program for their support. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Department of Defense, United States Army, White Sands Missile Range, NM.

References

[1] Fil Macias, "The test and evaluation of unmanned and autonomous systems," *ITEA Journal*, vol. 29, pp. 388–395, 2008.

[2] Department of Defense. 2008. *Systems Engineering guide for system of systems,* version 1.0.

[3] Leone Young, John Farr and Ricardo Valerdi, *"The Role of Complexities in Systems Engineering Cost Estimating Processes",*8th Conference on Systems Engineering Research, March 2010

[4] Alex Gorod, Brian Sauser and John Boardman, *"System-of-Systems Engineering Management: A Review of Modern History and a Path Forward",* IEEE Systems Journal, Vol 2, No. 4, December 2008

[5] Jo Ann Lane, "*Impact of System of System Management Strategies on System of System Capability Engineering Effort*," PhD Dissertation, University of Southern California, CA, May 2009.

[6] *Basic Test Training Course,* Army Test and Evaluation Center, 2010

[7] Ricardo Valerdi, "*The Constructive Systems Engineering Cost Model*," PhD Dissertation, University of Southern California, CA, August 2005.

[8] Ricardo Valerdi and M. Wheaton, *ANSI/EIA 632 as a standardized WBS for COSYSMO*, AIAA-2005-7373, Proceedings of the AIAA 5th Aviation, Technology, Integration, and Operations Conference, Arlington, Virginia, 2005

[9] JoAnn Lane and Barry Boehm. 2007. *Modern tools to support DoD software intensive system of systems cost estimation*: A DACS state-of-the-art report, DACS Report Number 347336. New York: Data and Analysis Center for Software.

[10] Barry Boehm and Basili, *"Software Defect Reduction Top 10 List*", Software Management, 135-137, January 2001

[11] Qi Li, Mingshu Li, Ye Yang, Qing Wang, Thomas Tan, Barry Boehm,and Chenyong Hu,*"Bridge the Gap between Software Test Process and Business Value: A Case Study*", ICSP 2009, LNCS 5543,. 212–223, 2009.

[12] Ellen George, *"Managing Your Way through the Integration and Test Black Hole",* Methods and Tools [Online], http://www.methodsandtools.com/archive/archive.php?

[13] Eduardo Aranha1 and Paulo Borba, *"Sizing systems tests for estimating test execution effort",* First International Symposium on Empirical Software Engineering and Measurement, Madrid, Spain, pp 107-116, 20-21 Sep. 2007.

[14] Barry Boehm, Chris Abts, A. Winsor Brown, Sunita Chulani, Bradford K. Clark, Ellis Horowitz, Ray Madachy, Donald J. Reifer, and Bert Steece, *Software cost estimation with COCOMO II*, Englewood Cliffs, NJ:Prentice-Hall, 2000