

**CALIBRATING SOFTWARE COST MODELS TO DEPARTMENT
OF DEFENSE DATABASES - A REVIEW OF TEN STUDIES**

Daniel V. Ferens
Air Force Research Laboratory

David S. Christensen
University of West Florida

1 February, 1998

CALIBRATING SOFTWARE COST MODELS TO DEPARTMENT OF DEFENSE DATABASES - A REVIEW OF TEN STUDIES

ABSTRACT

There are many sophisticated parametric models for estimating the size, cost, and schedule of software projects. In general, the predictive accuracy of these models is no better than within 25 percent of actual cost or schedule, about one half of the time (Thibodeau, 1981; IIT Research Institute, 1988). Several authors assert that a model's predictive accuracy can be improved by calibrating (adjusting) its default parameters to a specific environment (Kemerer, 1987; Van Genuchten and Koolen, 1991; Andolfi *et al.* 1996). This paper reports the results of a long-term project that tests this assertion.

From 1995 to 1997, masters students at the Air Force Institute of Technology calibrated selected software cost models to databases provided by two Air Force product centers. Nine parametric software models (REVIC, SASET, PRICE-S, SEER-SEM, SLIM, SOFTCOST, CHECKPOINT, COCOMO II, and SAGE) were calibrated. Data from the product centers were extracted and stratified for specific software estimation models, calibrated to specific environments, and validated using hold-out samples. The project was limited to software development cost or effort, although the same procedures could be used for size, schedule, or other estimating applications.

Results show that calibration does not always improve a model's predictive accuracy. Although one model which uses function points did show significantly improved accuracy (Mertes, 1996), the results could not be replicated on another database (Marzo, 1997).

INTRODUCTION

Software costs are continuing to rise in the Department of Defense (DOD) and other government agencies (Mosemann, 1996). To better understand and control these costs, DOD agencies often use parametric cost models for software development cost and schedule estimation. However, the accuracy of these models is poor when the default values embedded in the models are used (Boehm, 1991; Brooks, 1975). Even after the software cost models are calibrated to DOD databases, most have been shown to be accurate to within only 25 percent of actual cost or schedule about half the time. For example, Thibodeau (1981) reported the accuracy of early versions of the PRICE-S and SLIM models to be within 25 and 30 percent, respectively, on military ground programs. The IIT Research Institute (1988) reported similar results on eight Ada programs, with the most accurate model at only 30 percent of actual cost or schedule, 62 percent of the time.

Further, the level of accuracy reported by these studies is likely overstated because most studies have failed to use hold-out samples to validate the calibrated models. Instead of reserving a sample of the database for validation, the same data used to calibrate the models were used to assess accuracy (Ourada and Ferens, 1992).

In a study using 28 military ground software data points, Ourada (1991) showed that failure to use a hold-out sample overstates a model's accuracy. One half of the data was used to calibrate the Air Force's REVIC model. The remaining half was used to validate the calibrated model. REVIC was accurate to within 30 percent, 57 percent of the time on the calibration subset, but only 28 percent of the time on the validation subset.

Validating on a hold-out sample is clearly more relevant because new programs being estimated are, by definition, not in the calibration database. The purpose of this study is to calibrate and properly evaluate the accuracy of selected software cost estimation models using hold-out samples. The expectation is that calibration improves the estimating accuracy of a model (Kemerer, 1987; Van Genuchten and Koolen, 1991; Andolfi *et al.*, 1996).

THE DECALOGUE PROJECT

This paper describes the results of a long-term project at the Air Force Institute of Technology to calibrate and validate selected software cost estimation models. Software databases were provided by two Air Force product centers: the Space and Missile Systems Center (SMC), and the Electronic Systems Center (ESC). The project has been nicknamed the "Decalogue project" because ten masters theses extensively document the procedures and results of calibrating each software cost estimation model.

The Decalogue project is organized into three phases, corresponding to when the theses were completed. Five theses were completed in 1995; two theses were completed in 1996; three

theses were completed in 1997. Lessons learned during each phase were applied to the next phase. A brief description of each phase and its results follows.

PHASE I.

Five theses were completed in 1995. Each thesis student calibrated a specific software cost model (Revised Enhanced Intermediate Version of COCOMO (REVIC), Software Architecture Sizing and Estimating Tool (SASET), PRICE-S, SEER-SEM, and SLIM) using the SMC software database. REVIC and SASET are owned by the government. The remaining models are privately owned.

The SMC database was developed by Management Consulting and Research, and contains detailed historical data for over 2,500 software programs (MCR, 1995). The database includes inputs for REVIC, SASET, PRICE-S, and SEER-SEM for some of the 2,500 projects, but none specifically for SLIM.

The details of each thesis project are described in the separate thesis reports (1995) of Weber, Vegas, Galonsky, Rathmann, and Kressin. Each is available from the Defense Technical Information Center. Additional detail is also available from Ferens and Christensen (1997). Here, only the highlights are provided, and include a short description of the software models, the calibration methodology, and the results.

REVIC. This model is the Air Force Cost Analysis Agency's computerized variant of the Constructive Cost Model (COCOMO), developed by Dr. Barry Boehm. REVIC is calibrated the same way as COCOMO (Boehm, 1981). The nominal intermediate equations for REVIC are of the form $E = A (KDSI)^B$ where E is effort in person-months, KDSI is thousands of delivered source instructions, and A and B are the constants to be calibrated. The equations can be modified by calibrating A, B, or A and B. In calibrating the model, the product of nineteen effort adjustment factors is computed for each program and used to adjust for program variation. A large database is highly desirable if both A and B are calibrated.

PRICE-S. As discussed in the *User's Manual* (PRICE-S, 1993), this model is calibrated by running the model in the ECIRP (PRICE backwards) mode. In this mode, the actual cost or effort, and all inputs except the model's productivity factor (PROFAC), are entered into the model. The inputs include program size, language, application mix, hardware utilization, integration difficulty, platform, and several complexity factors. The output is a value of PROFAC for each project analyzed. PROFAC, which captures the skill levels, experience, efficiency, and productivity of an organization, is a very sensitive parameter; small changes in PROFAC result in relatively large effort estimation differences.

SEER-SEM. There are several versions of SEER-SEM; Rathman (1995) used version 4.0 for his thesis project. According to the *User's Manual* (SEER-SEM, 1994), this version of the model can be calibrated in either of two ways. The first way is to calibrate an "effective

technology rating" (ETR), a parameter that reflects relative productivity. To calibrate ETR, the user must enter values for size, effort or schedule, and "knowledge base" parameters. Knowledge base parameters include information about the platform, application, acquisition method, development method, and development standard used.

Instead of calibrating ETR, the user may calibrate effort and schedule adjustment factors from historical data. These factors are multipliers for which the nominal value is 1.0. Factors greater than 1.0 result in longer schedules and greater effort. The factors, like the ETR, can be included in a custom knowledge base for future programs. While the factors are easier to understand and work with than ETR, more input data are needed. Rathman (1995) used this latter method in his thesis.

SLIM. Version 3.2 of the Software Life Cycle Model is calibrated by entering actual size, effort, schedule, and number of defects on historical programs. The model outputs a "productivity index" (PI) and a "manpower buildup index" (MBI) for each program. Since the user cannot directly enter MBI into the model, the calibrated PI is of most interest to the user. Like PROFAC in PRICE-S, PI, which measures the total development environment, is also very sensitive.

SASET. This model was developed by Martin Marietta under contract to the United States Navy and Air Force (Ratliff, 1993). A calibration tool, the Database Management System Calibration Tool (Harbert, *et al.*, 1992), is available with the model. The tool adjusts the model's "productivity calibration constants" (PCCs), for the type of software (systems, application, or support) using the size, effort, and complexity of past programs. The calibration can be further refined by adjusting for different classes (avionics, ground, manned space, etc.) of software. As usual, there are default values for these constants if the user cannot calibrate the model.

Calibration rules. The five models were calibrated to a portion of the SMC database. The database was divided into the following subsets: military ground, avionics, unmanned space, missiles, and military mobile. The military ground subset was further divided into command and control programs and signal processing programs. Each subset was then divided into calibration and holdout samples using three rules:

- (1) If there were less than nine data points, the subset was considered too small for a hold-out sample and could not be validated.
- (2) If there were between nine and eleven data points, eight were randomly selected for calibration and the rest were used for validation.
- (3) If there were twelve or more data points, two-thirds were randomly selected for calibration and the rest were used for validation.

The accuracy of each model was evaluated using criteria proposed by Conte, *et al.* (1986) based on the following statistics:

$$\text{Magnitude of Relative Error (MRE)} = |\text{Estimate} - \text{Actual}| / \text{Actual} \quad (1)$$

$$\text{Mean Magnitude of Relative Error (MMRE)} = (\text{MRE}) / n \quad (2)$$

$$\text{Root Mean Square (RMS)} = [(1/n) (\text{Estimate} - \text{Actual})^2]^{1/2} \quad (3)$$

$$\text{Relative Root Mean Square (RRMS)} = \text{RMS} / [(\text{Actual}) / n] \quad (4)$$

$$\text{Prediction Level (Pred (.25))} = k/n \quad (5)$$

For Equation 5, n is the number of data points in the subset and k is the number of data points with MRE # 0.25. According to Conte, *et al.* (1986), a model's estimate is accurate when MMRE # 0.25, RRMS # 0.25, and Pred (.25) < .75.

Results. Table 1 summarizes the results of Phase 1. Due to an oversight, not all five of these reported RRMS. Thus, only MMRE and PRED (.25) are shown. "Validation sample size" is the number of data points in the holdout sample used for validation. For some models, the military ground subsets (signal processing and command and control) were combined into an overall military ground subset to obtain a sufficiently large sample size for validation.

TABLE 1

REVIC, SASET, PRICE-S, SEER-SEM, AND SLIM CALIBRATION RESULTS (1995)

<u>Model</u>	<u>Data Set</u>	<u>Validation Sample Size</u>	<u>Pre-Calibration</u>		<u>Post-Calibration</u>	
			<u>MMRE</u>	<u>PRED (.25)</u>	<u>MMRE</u>	<u>PRED (.25)</u>
REVIC	Military Ground	5	1.21	0	0.86	0
	Unmanned Space	4	0.43	0.50	0.31	0.50
SASET	Avionics	1	1.76	0	0.22*	1.00*
	Military Ground	24	10.04	0	0.58	0
PRICE-S	Military Ground	11	0.30	0.36	0.29	0.36
	Unmanned Space	4	0.34	0.50	0.34	0.50
SEER-SEM	Avionics	1	0.46	0	0.24*	1.00*
	Command and Control	7	0.31	0.43	0.31	0.29
	Signal Processing	7	1.54	0.29	2.10	0.43
SLIM	Military Mobile	4	0.39	0.25	0.46	0.25
	Command and Control	3	0.62	0	0.67	0

* Met Conte's criteria

As shown in Table 1, most of the calibrated models were inaccurate. In the two instances where the calibrated models met Conte's criteria, only one data point was used for validation. Thus, these results are not compelling evidence that calibration improves accuracy. In fact, in some cases the calibrated model was less accurate than the model before calibration.

These results may be due in part to the nature of the databases available to DOD agencies. In the SMC database, the developing contractors are not identified. Therefore, the data may represent an amalgamation of many different development processes, programming styles, etc., which are consistent within contracting organizations, but vary widely across contractors.

Furthermore, because of inconsistencies in software data collection among different DOD efforts, actual cost data and other data may be inconsistent and unreliable.¹

PHASE II.

In 1996 two additional models, SoftCost-OO and CHECKPOINT, were calibrated by two masters students. Details are provided in their thesis reports (Southwell, 1996; Mertes, 1996). A brief description of each model, the calibration procedures, and the results of Phase 2 follow.

SoftCost-OO. The SoftCost Object-Oriented (OO) model is a commercial model originally developed by Don Reifer, and marketed by Resource Calculations, Inc. The model is a modification of SoftCost-Ada developed by Reifer during the late 1980s. In addition to size, SoftCost-OO uses twenty-eight parameters in four categories (product, process, personnel, and project) to adjust effort and schedule for a particular program. Key parameters include system architecture, application type, OO program experience, analyst capability, and reuse costs and benefits. The model is calibrated by simultaneously adjusting two factors: an average work force factor, and a productivity factor of thousands of executable source lines of code per person-month. Currently, these factors must be calibrated off-line using an electronic spreadsheet. An on-line capability is envisioned for the future (SoftCost-OO, 1994).

CHECKPOINT. The CHECKPOINT model is a commercial model marketed by Software Productivity Research (SPR) and is based on the work of Capers Jones. It is unique among the models calibrated in this study because the internal algorithms are based on function points instead of lines of code.² If a user inputs lines of code and language, the model converts lines of code to function points using pre-set values for the language specified. A user can obtain a basic estimate by specifying (1) the nature and scope of the project, (2) the project class and type, and (3) complexity ratings for design, code, and data. The complexity ratings are used to adjust the function point count. A user may also enter values for more than one hundred detailed parameters in five categories (process, technology, personnel, environment, and special factors). In addition, a user can calibrate CHECKPOINT by creating templates from historical programs (SPR, 1993). These templates are used to set default values for new programs for selected input parameters.

Calibration rules. With a few exceptions related to the subsets to calibrate and the hold-out sample rules, the two models were calibrated and validated using the same methods that were used in Phase I. A seventh subset of the SMC database, ground in-support-of-space (designated

¹ This problem was addressed in Phase 3 of the Decalogue project, where the ESC database was used. The ESC database contains an identifier for each contributing contractor.

² Function points are weighted sums of five attributes or functions of a software program (inputs, outputs, inquiries, interfaces, and master files). Based on their analysis of more than 30 data processing programs, Albrecht and Gaffney (1983) report that function points may be superior to SLOC as predictors of software development cost or effort.

“Ground Support” in Tables 2, 3, and 4) was used for both models. For SoftCost-OO, three additional subsets for European Space Agency programs were added since SoftCost-OO is used extensively in Europe. For CHECKPOINT, the missile subset was not used, and no European programs were used. In addition, data were obtained on Management Information System (MIS) programs written in COBOL from a local contractor, and a subset for COBOL programs was added to determine if stratification by language would provide better results. Finally, the rules to determine the sizes of the calibration and holdout samples were changed to avoid the problem of "single-point validations" experienced in Phase 1. Specifically, if there were eight or more data points in a subset, half were used for calibration, and the other half for validation. If there were fewer than eight data points, that subset was not used.

Results. The following three tables show the results of calibrating each model. For SoftCost-00 (Table 2), calibration almost always improved the accuracy of the model, although none of the subsets met Conte’s criteria. For CHECKPOINT, all but one subset met the criteria when predicting development effort (Table 3), but none met the criteria when predicting schedule (Table 4).

TABLE 2
SOFTCOST CALIBRATION RESULTS (1996)

Data Set	Validation	Pre-Calibration			Post-Calibration		
	Sample Size	MMRE	RRMS	PRED (.25)	MMRE	RRMS	PRED (.25)
Ground Support	15	2.73	3.13	0.13	1.80	1.96	0.20
Ground Support (Europe)	25	3.05	3.61	0.08	0.67	0.84	0.36
Unmanned Space	5	0.56	1.05	0.20	0.48	0.92	0.20
Unmanned Space (Europe)	7	1.79	0.79	0.14	1.27	0.84	0.14
Avionics	5	0.71	0.76	0.20	0.85	0.56	0.20
Command and Control	6	1.90	3.43	0.17	0.52	0.87	0.50
Signal Processing	9	0.43	0.61	0.11	0.28	0.64	0.44
Military Mobile	5	0.63	0.51	0.20	0.42	0.40	0.20

Since CHECKPOINT uses function points as a measure of size, they were used when sufficient data points were available for the subsets; otherwise, source lines of code (SLOC) were used. For three function point effort subsets, there was substantial improvement in accuracy after the model was calibrated for other programs in these subsets, especially for the MIS COBOL subset. Except for the Command and Control subset, the SLOC effort subsets met Conte’s criteria both before and after calibration. Although calibration did not significantly improve accuracy for these subsets (primarily because SLOC are an output, not an input, to CHECKPOINT), the accuracy was very good even without calibration. The CHECKPOINT results for effort estimation are especially noteworthy because the inputs for this model were not even considered when the SMC database was developed.

TABLE 3

CHECKPOINT CALIBRATION RESULTS (EFFORT, 1996)

<u>Data Set</u>	<u>Validation Sample Size</u>	<u>Pre-Calibration</u>			<u>Post-Calibration</u>		
		<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>	<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>
<u>Effort - Function Points</u>							
MIS - COBOL	6	0.54	0.10	0.67	0.02*	0.01*	1.00*
Military Mobile - Ada	4	1.38	0.41	0.25	0.19*	0.06*	0.75*
Avionics	4	0.82	0.68	0.50	0.16*	0.11*	0.75*
<u>Effort - SLOC</u>							
Command and Control	6	0.19*	0.14*	0.50	0.16*	0.16*	0.50
Signal Processing	10	0.09*	0.08*	1.00*	0.09*	0.08*	1.00*
Unmanned Space	5	0.05*	0.05*	1.00*	0.04*	0.06*	1.00*
Ground Support	4	0.05*	0.06*	1.00*	0.05*	0.06*	1.00*
COBOL Programs	4	0.05*	0.05*	1.00*	0.05*	0.05*	1.00*

Met Conte's Criteria

TABLE 4

CHECKPOINT CALIBRATION RESULTS (SCHEDULE, 1996)

<u>Data Set</u>	<u>Validation Sample Size</u>	<u>Pre-Calibration</u>			<u>Post-Calibration</u>		
		<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>	<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>
MIS - COBOL	6	0.31	0.37	0.17	0.29	0.72	0.33
Unmanned Space	5	0.60	0.62	0.00	0.50	0.68	0.00
Ground Support	4	0.60	0.62	0.00	0.60	0.62	0.00
COBOL Programs	4	0.60	0.60	0.00	0.60	0.60	0.00

Although these results are promising, it should not be assumed that CHECKPOINT will do as well in other environments. The best results for the CHECKPOINT model were for the MIS COBOL data set, which was obtained from a single contractor. Data from multiple contractors, which often characterize DOD databases, are more difficult to calibrate accurately. Furthermore, CHECKPOINT is a function point model. If the user wants to input size in SLOC (which is usually the case), the user or model must first convert the SLOC to function points. Unfortunately, the conversion ratios are sometimes subject to significant variations. Thus, the SLOC effort results for CHECKPOINT may not work out as well elsewhere.

PHASE III.

In 1997 three models (COCOMO II, SAGE, and CHECKPOINT) were calibrated. COCOMO II, the successor to Boehm's COCOMO model (1981), was calibrated to the SCM database. SAGE model, a commercial model developed by Randy Jensen, was calibrated to the SMC and ESC databases. Finally, CHECKPOINT was calibrated to the ESC database to determine whether the unusually high accuracy reported by Mertes (1996) could be achieved on a

different database. As before, the details are documented in the 1997 thesis reports (Bernheisel, Marzo, and Shrum). Here, only the highlights are described.

The COCOMO II "Post-Architecture" model, the long-awaited successor to COCOMO, is expected to have an on-line calibration capability; however, it was not available for this study. Instead, the model was calibrated using the procedure described in Phase I for REVIC. Since the data sets were relatively small, only the coefficient of the effort equation was calibrated. The exponent was set to an "average" value of 1.153.

SAGE is a commercial model developed by Dr. Randy Jensen (1996). It currently has an on-line calibration capability where effort and schedule equations are calibrated simultaneously by adjusting a "basic technology constant" (Ctb) for effort, and a "system complexity" (D) factor for schedule. Due to time limitations, only Ctb was calibrated for this study, and a value of 15, typical for the types of subsets calibrated here, was used for D. The basic technology constant accounts for an organization's personnel capability and experience, use of modern practices and tools, and computer turnaround and terminal response times. Higher values of Ctb represent higher productivity, and result in relatively lower costs and shorter schedules.

The SMC database was stratified into the seven categories used in Phase II (1996). No changes were made except that a more recent edition of the database was used.

The ESC database contains information on 52 projects and 312 computer software configuration items (Marzo, 1997). It contains contractor identifiers and language, but not information on application type. It does contain inputs for the SEER-SEM model for which it was originally developed. The ESC database was initially stratified by contractor since it was believed that a model can be more accurate when calibrated for a specific developer (Kemerer, 1987). For CHECKPOINT, the ESC database was also stratified by language, contractor, and language.

Calibration rules. The techniques used to calibrate the models were significantly improved over those used in the earlier phases. In the past, small data sets reduced the meaningfulness of the calibration. Indeed, making statistically valid inferences from small data sets of completed software projects is a common limitation of any calibration study. To overcome this limitation, each model was calibrated multiple times by drawing random samples from the data set. The remaining hold-out samples were used for validation. Averages of the validation results became the measure of accuracy. This technique, known as "resampling," is becoming an increasingly popular and acceptable substitute for more conventional statistical techniques (University of Maryland, 1997).

The resampling technique is flexible. For CHECKPOINT, resampling was used on only the small data sets (8-12 data points). Four random samples from the small data sets were used to calibrate and validate the model. For COCOMO II, only data sets of twelve or more data points were used, and resampling was accomplished on all data sets by using 80 percent of the data

points (selected randomly) for calibration and the remaining 20 percent for validation. The process was repeated five times, and the results were averaged. For SAGE, all data sets having four or more points were used with an even more comprehensive resampling procedure. Simulation software, *Crystal Ball*, was used to select two data points for validation and the rest for calibration. Instead of limiting the number of runs to four or five, all possible subsets were run.

Results. Table 5 shows the results of the CHECKPOINT calibration using the ESC database. Unlike the results reported by Mertes (1996), none of the data sets met any of Conte's criteria, even those for a single contractor. This may be due in part to the lack of function point counts in the ESC database; only SLOC are provided for all data points. However, since Mertes' results using CHECKPOINT for SLOC were also very good, it is difficult to account for the differences between the results of Mertes (1996) and Shrum (1997).

TABLE 5
CHECKPOINT CALIBRATION RESULTS (1997)

<u>Data Set</u>	Validation	Pre-Calibration			Post-Calibration		
	<u>Sample Size</u>	<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>	<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>
Ada Language	8	1.21	1.34	0.00	1.70	2.54	0.50
Assembly Language	11	0.83	1.44	0.09	2.05	1.20	0.18
FORTRAN Language	12	0.73	1.12	0.17	0.70	2.31	0.17
JOVIAL Language	7	0.71	1.22	0.00	0.44	0.68	0.43
Contractor B	4**	0.60	0.74	0.13	0.64	0.49	0.25
Contractor J	11	0.69	0.91	0.18	1.33	1.43	0.18
Ada and Contractor R	5**	0.59	0.57	0.05	0.39	0.72	0.45
CMS2 and Contractor M	5**	0.91	1.13	0.00	0.69	0.64	0.10
FORTRAN and Contractor A	7	0.82	0.84	0.00	0.44	0.88	0.29
JOVIAL and Contractor J	6	0.80	1.42	0.00	0.37	0.70	0.33

** Resampling Used For This Set

Table 6 shows the results for COCOMO II, where calibration slightly improved the model's predictive accuracy, but none of the subsets met Conte's criteria. It is possible that better results may be attained when the on-line calibration capability is incorporated into the model.

TABLE 6
COCOMO II CALIBRATION RESULTS (1997)

<u>Data Set</u>	Total	Pre-Calibration			Post-Calibration		
	<u>Sample Size</u>	<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>	<u>MMRE</u>	<u>RRMS</u>	<u>PRED (.25)</u>
Command and Control	12	0.39	0.49	0.30	0.33	0.53	0.40
Signal Processing	19	0.45	0.63	0.33	0.38	0.53	0.40
Ground Support	15	0.71	1.16	0.07	0.66	0.95	0.20
Military Mobile	12	0.79	0.95	0.10	0.68	0.74	0.00

Table 7 shows the results for SAGE on both databases. Although calibration sometimes resulted in improved accuracy, only a few sets met Conte's criteria. This is somewhat surprising for the ESC data sets, where individual contractors are identified by a code letter, and Ctb should be consistent for a company. It may be that even within a single company software programs are developed differently. Also, it is possible that if the simultaneous effort and schedule calibration capability which is now integrated into SAGE was used, the results would be better.

TABLE 7
SAGE CALIBRATION RESULTS (1997)

Data Set	Total Sample Size	Pre-Calibration			Post-Calibration		
		MMRE	RRMS	PRED (.25)	MMRE	RRMS	PRED (.25)
SMC – Avionics	9	0.45	0.54	0.21	0.39	0.52	0.24
Command and Control	10	0.23*	0.23*	0.70	0.29	0.30	0.45
Signal Processing	16	0.39	0.43	0.44	0.50	0.54	0.20
Unmanned Space	7	0.66	0.69	0.14	0.59	0.88	0.30
Ground Support	14	0.32	0.44	0.43	0.32	0.44	0.43
Military Mobile	10	0.37	0.47	0.29	0.41	0.52	0.36
Missile	4	0.66	0.89	0.00	0.67	0.44	0.24
ESC – Contractor A	17	0.48	0.57	0.17	0.41	0.40	0.31
Contractor J	17	0.37	0.47	0.33	0.47	0.57	0.14
Contractor R	6	0.32	0.36	0.32	0.21*	0.23*	0.54

* Met Conte's Criteria

CONCLUSIONS

Calibration does not always improve a model's predictive accuracy. Most of the calibrated models evaluated in this project failed to meet Conte' criteria. The one exception was the calibration of CHECKPOINT to the SMC database (Mertes, 1996), where almost all of the calibrated data sets met Conte's criteria, both for function point and SLOC applications. Unfortunately, this result could not be replicated on the ESC database (Shrum, 1997) using a superior validation technique. Overall, none of the models was shown to be more accurate than within 25 percent of actual cost or effort, one half the time.

This does not mean the Decalogue project was a failure. Much was learned about the models, their strengths and weaknesses, and the challenges in calibrating them to DOD databases. One major insight of the project is that the use of a holdout sample is essential for meaningful model calibration. Without a holdout sample, the predictive accuracy of the model is probably overstated. Since all new projects are outside of the historical database(s), validation is much more meaningful than the more common practice of analyzing within-database performance. The calibrations performed in 1997 also developed and applied resampling as a superior technique to use in validating small samples. It is better than just using one subset of data for a holdout, and can be done easily with modern software, such as *Excel* and *Crystal Ball*. Hopefully, the findings

of the Decalogue project will inspire additional effort in the area of model calibration, and more promising results will be obtained.

REFERENCES

- Albrecht, A.J. and J.E. Gaffney. November, 1983. "Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation." *IEEE Transactions on Software Engineering*. Volume SE-9.
- Andolfi. M. August, 1996. "A Multi-criteria Methodology for the Evaluation of Software Costs Estimation Models and Tools." *CSELT Technical Reports 24*: 643-659.
- Bernheisel, Wayne A. 1997. "Calibration and Validation of the COCOMO II.1997.0 Cost/Schedule Estimating Model to the Space and Missile Systems Center Database." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.
- Boehm, Barry W. 1981. *Software Engineering Economics*. Englewood Cliffs, NJ, Prentice-Hall.
- Conte, S.D., H.E. Dunsmore, and V. Y. Shen V.Y. 1986. *Software Engineering Metrics and Models*. Menlo Park, CA: Benjamin-Cummings.
- Daly, Bryan A. 1991. "A Comparison of Software Schedule Estimators." Unpublished masters thesis. Dayton, OH: Air Force Institute of Technology.
- Ferens, Daniel V., and David S. Christensen. Fall 1997. "Software Cost Model Calibration, An Air Force Case Study." *Journal of Cost Analysis*.
- Galonsky, James C. 1995. "Calibration of the PRICE-S Model." Unpublished masters thesis. Dayton, OH: Air Force Institute of Technology.
- Harbert, Chris E., et al. September 1992. *Database Management System Calibration Tool (DBMS) Version 1.4 User's Guide*. Denver, CO, Martin-Marietta.
- IITRI. 1989. "Test Case Study: Estimating the Cost of Ada Software Development." Lanham, MD, IIT Research Institute.
- Jensen, Randall W. 1996. "A New Perspective in Software Schedule and Estimation." Brigham City, UT, Software Engineering, Inc.
- Kemerer, Chris F. May 1997. "An Empirical Validation of Software Cost Estimation Models", *Communications of the ACM*, pp. 416-429.

- Kile, Raymond L. 18 February 1991. *REVIC Software Cost Estimating User's Manual, Version 9.0*.
- Kressin, Robert K. 1995. "Calibration of the Software Life Cycle Model (SLIM) to the Space and Missile Systems Center Software Database (SWDB)." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.
- Marzo, David B. 1997. "Calibration and Validation of the SAGE Cost/Schedule Estimating System to United States Air Force Databases." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.
- MCR. 1995. *Space and Missiles Center Software Data Base User's Manual: Version 2.1*. Oxnard, CA, Management Consulting and Research.
- Mertes, Karen R. 1996. "Calibration of the CHECKPOINT Model to the Space and Missile Systems Center (SMC) Software Database (SWDB)." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.
- Mosemann, Lloyd K., II. June 1996. *Guidelines for Successful Acquisition and Management of Software Intensive Systems, Volume 1*. Hill AFB, UT, Ogden Air Logistics Center.
- Ourada, Gerald L. 1991. "Software Cost Estimating Models: A Calibration, Evaluation, and Comparison." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.
- Ourada, Gerald L. and Daniel V. Ferens. 1992. "Software Cost Estimating Models: A Calibration, Evaluation, and Comparison." *Cost Estimating and Analysis: Balancing Technology and Declining Budgets*. New York, Springer-Verlag, 1992, pp. 83-101.
- The PRICE Software Model User's Manual*. 1993. Moorestown, NJ, PRICE Systems.
- Rathmann, Kolin D. 1995. "Calibration of the System Evaluation and Estimation of Resources Software Estimation Model (SEER-SEM) for the Space and Missile Systems Center." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.
- Ratliff, Robert W., et al. 1993. *SASET 3.0 User's Guide*, Denver, CO, Martin-Marietta.
- SEER-SEM User's Manual*. September 1994. Los Angeles, CA, Galorath Associates.
- Shrum, Thomas C. 1997. "Calibration and Validation of the CHECKPOINT Model to the Air Force Electronic Systems Center Software Database." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.

SLIM 3.2 For Windows User's Manual, Second Edition. 1993. McLean, VA, QSM.

SoftCost-OO User Guide and Reference Guide, Version 3.1. July 1994. Englewood, CO, Resource Calculations, Inc.

Southwell, Steven V. 1996. "Calibration of the SoftCost-R Software Cost Model to the Space and Missile Systems Center (SMC) Software Database (SWDB)." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.

Software Productivity Research (SPR). 1996. *CHECKPOINT For Windows User's Guide, Version 2.3.1.* Burlington, MA, Software Productivity Research.

University of Maryland. 1997. "The Resampling Project." College Park, MD.

University of Southern California. 1997. *COCOMO 2.0 Model Definition Manual.* Los Angeles, CA, University of Southern California, 1997.

Van Genuchten, Michiel and Hans Koolen. 1991. "On the Use of Software Cost Models." *Information and Management 21*: 37-44.

Vegas, Carl D. 1995. "Calibration of the Software Architecture Sizing and Estimation Tool (SASET)." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.

Weber, Betty G. 1995. "A Calibration of the REVIC Software Cost Estimating Model." Unpublished masters thesis. Dayton, OH, Air Force Institute of Technology.

BIOGRAPHIES

Daniel V. Ferens

Daniel V. Ferens is an Engineering Analyst at Air Force Research Laboratory at Wright-Patterson AFB in Dayton, Ohio. He is also an Adjunct Associate Professor of Software Systems Management at the Air Force Institute of Technology (AFIT), Graduate School of Logistics and Acquisition Management, at Wright-Patterson AFB in Dayton, Ohio, where he teaches courses in software estimation and software management in general. He was the Advisor for all ten thesis described in this paper. He is an active member of the Society of Cost Estimating and Analysis and a lifetime member of the International Society of Parametric Analysts. Mr. Ferens has a Master's Degree in Electrical Engineering from Rensselaer Polytechnic Institute, and a Master's Degree in Business from the University of Northern Colorado.

David S. Christensen

David S. Christensen is an Associate Professor of Accounting at West Florida University. He was the reader for all ten theses described in this paper. He received his Ph.D. in accounting from the University of Nebraska-Lincoln in 1987, and has published extensively in the area of defense cost management. He is a member of the American Accounting Association, the Institute of Management Accountants, and the Society of Cost Estimating and Analysis. Presently, he serves as an Associate Editor to the *Journal of Cost Analysis*.