*Int. J. Human–Computer Studies* (1996) **44,** 171–185

# The relationship between errors and size in knowledge-based systems

DANIEL E. O'LEARY

*3660 Trousdale Parkway, University of Southern California, Los Angeles, CA 90089-1421, USA. email: oleary@rcf.usc.edu*

Previous researchers in knowledge-based systems verification have concentrated on developing various approaches and computational tools to find errors in knowledge bases. However, unlike software engineering for traditional systems, there has been little investigation of the relationship between errors and system size. In addition, there has been little analysis of the relationship between the occurrence of different types of errors. Thus, this paper investigates the empirical relationships between knowledge-based system size and number of errors, and between the number and existence of different kinds of errors.

It is found that, in general, system size is statistically significantly correlated with two of those error types, and with total errors. Further it is found that the size of "smaller" systems is not correlated with total number of errors, but the size of "larger" systems is correlated with total number of errors. As a result, this evidence indicates that it can be important to use a modular approach in the development of such systems. In addition, it is found that the number of different types of errors have a statistically significant correlations with each other. Further, the existence of different errors types are statistically related. As a result, errors signal the existence of other errors. 

## 1. Introduction

Previous software engineering research (e.g. Akiyama, 1971) has found a positive relationship between computer program size and the number of errors.† However, there has been virtually no such analysis of errors in knowledge-based systems. Instead, knowledge-based systems researchers have focused primarily on the development of different approaches and tools to find errors or anomalies in knowledge-based systems. Further, little attention has been directed to determining the existence of a relationship between the number of different types of errors, or the relationship between the existence of different kinds of errors in systems, in general. The purpose of this paper is to mitigate those limitations.

### 1.1. RELATIONSHIPS BETWEEN SYSTEM SIZE AND ERRORS

There are a number of reasons to anticipate that there is a relationship between knowledge-based systems size and the number of errors. First, previous software engineering research into the number of errors in other types of computer programs, has found that the size of the program is related to the number of errors in the

† This paper employs the term "errors" throughout since in the situation investigated in this paper they were errors, and since this terminology use is consistent with much of the verification and validation literature. However, there are a variety of other terms that are used, depending on which literature is addressed. An alternative set of terminology deriving from the reliability literature would be to employ the terms "faults" and "manifestations of faults." Some of the software engineering literature uses the terms "defects." Some of the knowledge-based systems literature uses the term "anomalies."



1071-5819/96/020171 + 15$12.00/0 

program (Akiyama, 1971; Conte, Dunsmore & Shen, 1986). In general, the larger the program the greater the number of errors.

Second, the cognitive capacity of humans is limited (e.g. Hogartin, 1987). As a result, we would expect that limited capacity could manifest itself in errors and biases in the development of computer programs. Under that theory, in general, the larger the required effort, the more errors and biases that would be exhibited. Thus, the larger the system, the greater the number of errors.

Third, from a probability perspective (Freund, 1971) and a reliability theory perspective (Conte *et al.,* 1986), it is ''reasonable'' to assume that the size of a knowledge-based system is related to the frequency of occurrence of errors. For example, the probability of introducing an error, whether from typing or carelessness or some other underlying process, is likely to increase as the system size increases. As a result, the larger the system the more likely there are more errors in it.

### 1.2. IMPORTANCE OF A RELATIONSHIP BETWEEN SIZE AND ERRORS

The determination of the existence of a relationship between errors and size in knowledge based systems would be important for a number of reasons. If the number of errors are related to system size, then that indicates that should be taken into account as part of the testing process. If ''smaller'' systems are unrelated to the number of errors, but ''larger'' systems are related to the number of errors, then that would suggest that modularization could be used to mitigate the number of errors. Further, it would also suggest that other software engineering approaches should be integrated in the process of developing knowledge-based systems. Finally, there is some concern (e.g. Conte *et al.,* 1986: p. 335) that previous studies are no longer valid in today's programming environments. This is a particular concern for knowledge-based systems, whose advocates suggest many advantages for, e.g. rule-based approaches, when compared to other kinds of representations.

### 1.3. RELATIONSHIPS BETWEEN ERROR TYPES

In addition, there also is reason to anticipate that if there are errors of one type then there are errors of another type. Cognitive limitations of humans are likely to generate multiple kinds of errors. The same underlying processes (typing errors, carelessness, information overload, etc.) may cause more than one type of error. Thus, one type of error may signal that there are other similar errors or errors of a related type.

If one error type signals another error type then that can be used to guide testing efforts. Such signals can take two basic approaches. First, the *existence* of one error might signal the existence of another. Second, the *number* of errors of one type could signal the number of errors of another type.

### 1.4. THIS PAPER

This paper proceeds as follows. Section 2 provides a brief background on the verification and validation of knowledge-based systems and discussion of error types. Section 3 summarizes the data that is investigated and the methodology used in this paper. Section 4 presents the findings. Section 5 discusses the findings. Section 6

briefly summarizes the paper, its contributions and discusses some extensions of interest for further research.

## 2. Previous research

The previous research relates to both knowledge-based systems and to the relationship between size and errors in computer programs.

### 2.1. SIZE AND ERRORS IN COMPUTER PROGRAMS

There is a substantial literature associated with the relationship between errors and various program size metrics (e.g. Conte *et al.,* 1986). Aikyama (1971) was among the first researchers in this area. Aikyama investigated the relationship between number of errors and size, as measured by lines of code, count of decisions, sub-routine calls, and the sum of counts of decisions and sub-routine calls. When the number of errors was regressed individually against each of those four independent variables, he found slopes on regression coefficients of 0.018, 0.2, 0.27 and 0.12. He also found constants in the regression equation of 4.86, −1.4, 6.9 and −0.88, respectively. The correlation between each of these independent variables and the number of errors were, respectively, 0.83, 0.89, 0.91, 0.92.

### 2.2. VERIFICATION OF KNOWLEDGE-BASED SYSTEMS

There is a growing body of literature on research in the verification of knowledge-based systems. Gupta (1991) provides a collection of about 50 papers in verification and validation of knowledge-based systems. O'Keefe and O'Leary (1992) provide a summary of the verification and validation literature in knowledge-based systems, including over 100 references for verification and validation.

An analysis of those and other sources (e.g. Brown, Nielson & Phillips, 1993) yields the conclusion that previous research in verification of knowledge-based systems has been primarily aimed at the development of systems designed to perform verification and the theory necessary to develop systems to perform that verification. Little, if any, research has been done on the descriptive analysis of the occurrence of errors.

### 2.3. SYSTEMS THAT PERFORM VERIFICATION OF KNOWLEDGE-BASED SYSTEMS

There is a history of developing new systems to perform verification of knowledge-based systems. Probably the first work on knowledge-base verification is summarized in Davis and Lenat (1982). TEIRESAIS was the first attempt to automate the process of debugging a knowledge base. TEIRESAIS verified MYCIN rule bases, by examining new rules as they were added to the knowledge base.

Suwa, Scott and Shortliffe (1982) built a program for verifying the completeness

and consistency of a rule base. Their system was built within the context of the ONCOCIN system, a rule-based clinical oncology system. ONCOCIN, unlike TEIRESAIS, was meant to be used as the system was being developed.

Nguten, Perkins, Lafferty and Pecora (1987) extended ONCOCIN. Their system, CHECK, took a more global view of the knowledge base. They integrated new tests into the verification process. In particular, they developed tests for errors such as unreachable conclusions and dead-end IF conditions, dead-end goals, unnecessary IF conditions, and unused attributes/constructs.

Stachowitz and Combs (1987) and Chang, Combs and Stachowitz (1990) discussed Lockheed's verification system, EVA. The goal of the developers of EVA was to build a generic set of tools that could be used to validate any knowledge-based system. To facilitate the interaction between the developer and EVA, Stachowitz and Combs (1987) and Chang *et al.* (1990) designed and implemented a unifying architecture and defined a common metaknowledge base for specifying requirements, constraints and models for domain knowledge.

Jafar (1989) developed another verification tool, Validator, designed to verify rule-based systems. Validator can be used to verify systems written in M.1 (Teknowledge, 1985), one of the first expert system shells.

Preece (1990) developed Cover with the intent of verifying a medical system. Cover has since been generalized for use in other domains. Cover performs a number of knowledge-base checks, such as redundant rules, useless rules, unreachable rules, missing values, missing rules, illegal values, and others.

### 2.4. ERROR TYPES: SOURCES AND MANIFESTATIONS OF ERRORS

The development of these systems has focused on four sources of errors and anomalies (e.g. O'Keefe & O'Leary, 1992; O'Leary, 1987): redundancy, completeness, correctness and consistency. Those sources are based on the relationships between rules and components of rules. As in much of the previous literature, the focus here is on rule-based systems. However, the scope can be expanded to include other forms of knowledge representation.

Redundancy errors, e.g. occur when the same rule occurs more than a once, or the same attribute or conclusion appears more than once in the same rule. Completeness errors occur when rules, rule attributes, conclusions, etc., are missing. One way to establish completeness is to require the developer to establish the appropriate constructs and then test to ensure that all the established constructs would be manifested as completeness errors. Correctness errors occur when the knowledge is represented in a manner that is incongruent with the particular type of knowledge representation. For example, correctness errors are manifested, e.g. when illegal values are used for variables. Consistency errors occur when the same attribute, conclusion, etc., is assigned multiple names. Consistency errors may also manifest themselves as unused constructs.

This paper presents an analysis of the manifestation of specific errors, the way they appear, rather than the generic cause or manner in which they are likely to be discovered. Three types of errors are analysed: number of unused constructs (a manifestation of completeness and consistency errors), number of redundancy

errors (a manifestation of redundancy errors), and number of illegal values (a manifestation of correctness errors).

## 3. Variables, data and data analysis

This section summarizes the variables analysed in order to determine the relationship between system size and number of errors, and the relationship between the different types of errors. In addition, this section summarizes the generation of the data and the analysis of that data.

Jafar (1989) captured data on errors in computer programs developed using M.1. The errors in the systems were found by analysing the sample systems using Validator (Jafar, 1989). The use of Validator to analyse the sample systems mitigated the potential introduction of investigator bias into the investigation. The issue of the impact of size on errors is analysed using that data.

### 3.1. SYSTEM SIZE METRIC

Kilobytes (kbytes) was used as the measure of system size. Although the number of rules are often the basis of measuring the size of a rule base, that measure can be misleading. Errors can occur in rules, or facts. Further, the number of rules can be a function of the ability and experience of the system designer to write rules. Measures such as kbytes provide an alternative that does not have those same limitations.

### 3.2. ERRORS INVESTIGATED

The data in this experiment captures three basic types of errors that were found in the analysis of rule-based systems developed in M.1 (Teknowledge, 1985): unused constructs, redundancies and illegal values. The unused construct errors include those where they are rules, facts and legal values that are defined, but cannot be used or reached. Redundancy errors include multiple occurrences of the same rule (e.g. two rules are concerned with obtaining values for the identical expression) and multiple methods of obtaining the information (e.g. the same expressions appear in rules and facts). Finally, the number of errors where illegal values were used in rules also was captured. These errors include those situations where, for example, used values and legal values were not the same or the value used was not defined as a legal value.

### 3.3. DATA

The data consists of 49 systems analysed by Jafar (1989), who presented data from two samples of systems. The systems in each sample spanned a broad range of applications, including medical systems (e.g. diagnoses of retinal disease), assistant (e.g. aid students using DOS and UNIX), environment analysis, and many others.

The systems derive from a wide range of environments including commercial and student systems.

The data from the two samples was pooled in order to generate sample sizes large enough to make statistical inferences. Prior to pooling, a *t*-test was made with the hypothesis that the two samples have the same mean (Freund, 1971: p. 119). The errors in each sample were scaled by the system size for compatibility. Thus, for each system there was a measure of total number of errors/kbytes. The mean for the two sets of systems were, 0.223 and 0.190, respectively. A *t*-test found that we can reject the hypothesis that the means are not the same. There was not sufficient evidence that there was a difference. Thus, the two samples were combined.

The data investigated in this paper was limited to those systems larger than 10 kbytes. This was done in order to ensure that the systems investigated were non trivial. In addition, two other observations were eliminated as being ''far-outliers'' (Velleman & Hoaglin, 1981). The data used in the analysis are summarized in Table 1.

### 3.4. DATA ANALYSIS

The analysis of the test of the relationship between size and number of errors, and the relationship between the number of different kinds of errors, used regression analysis, Pearson correlation coefficient analysis, computer intensive statistical analysis of the correlation coefficient and a chi-squared test.

A regression was done with size as the independent variable and the number of errors as the dependent variable. A *t*-test was used to measure the significance of the regression coefficient of the slope. In addition, the constant in the regression was investigated.

The correlation coefficient often is used to compare vectors of numbers to determine whether or not the vectors are independent. Correlation measures the linearity of the relationship between two variables. If the correlation coefficient is zero then that means that the two variables have no linear predictive ability for each other. If the one variable can be expressed exactly as a linear function of the other variable, then the two are either directly (correlation = 1) or inversely (correlation = $-1$) linearly related.

A $z$-coefficient test was used to measure the statistical significance of the hypothesis that the correlation coefficient is not equal to some comparison value, say zero (Freund, 1971; p. 381). If the correlation is large enough then the correlation is said to be statistically (significantly) different from that comparison value. Similarly, the $z$-coefficient can be used to measure whether the correlation is significantly different from any other value that varies from 0 to approaching an absolute value of one.

A computer intensive approach was used to provide an alternative non-parametric measure of the statistical significance of the correlation coefficients. Computer intensive approaches (e.g. Noreen, 1989) use the power of the computer to generate estimates of the statistical significance of test statistics. In this study, each sample pair was used to generate a distribution of 100 correlation coefficients. Those distributions were used to establish an alternative measure of statistical significance for each pair.

TABLE 1
*Verification errors from 51 systems*

| | System size (kbytes) | No. of unused constructs | No. of redundancy errors | No. of illegal values | Total errors |
|---|---|---|---|---|---|
| | 82 | 12 | 35 | 1 | 48 |
| | 100 | 1 | 0 | 0 | 1 |
| | 126 | 30 | 13 | 0 | 43 |
| | 60 | 0 | 0 | 0 | 0 |
| | 50 | 1 | 0 | 14 | 15 |
| | 120 | 4 | 0 | 1 | 5 |
| | 81 | 17 | 1 | 8 | 26 |
| | 54 | 15 | 4 | 0 | 19 |
| | 55 | 5 | 3 | 0 | 8 |
| | 15 | 2 | 1 | 1 | 4 |
| | 41 | 17 | 0 | 5 | 22 |
| | 54 | 4 | 0 | 0 | 4 |
| | 27 | 3 | 0 | 1 | 4 |
| | 28 | 1 | 1 | 2 | 4 |
| | 37 | 1 | 0 | 0 | 1 |
| | 28 | 0 | 0 | 0 | 0 |
| | 32 | 0 | 0 | 1 | 1 |
| | 18 | 0 | 1 | 2 | 3 |
| | 19 | 0 | 0 | 0 | 0 |
| | 17 | 0 | 2 | 0 | 2 |
| | 53 | 0 | 0 | 8 | 8 |
| | 21 | 0 | 0 | 0 | 0 |
| | 42 | 9 | 0 | 3 | 12 |
| | 20 | 0 | 0 | 0 | 0 |
| | 19 | 0 | 0 | 1 | 1 |
| | 40 | 0 | 0 | 0 | 0 |
| | 13 | 1 | 0 | 0 | 1 |
| | 20 | 6 | 3 | 1 | 10 |
| | 28 | 1 | 12 | 3 | 16 |
| | 27 | 3 | 0 | 2 | 5 |
| | 30 | 1 | 0 | 0 | 1 |
| | 52 | 0 | 0 | 0 | 0 |
| | 102 | 6 | 1 | 0 | 7 |
| | 14 | 0 | 0 | 0 | 0 |
| | 19 | 0 | 0 | 0 | 0 |
| | 54 | 0 | 0 | 0 | 0 |
| | 26 | 0 | 0 | 0 | 0 |
| | 35 | 1 | 0 | 3 | 4 |
| | 62 | 14 | 0 | 0 | 14 |
| | 19 | 0 | 0 | 0 | 0 |
| | 46 | 7 | 0 | 5 | 12 |
| | 59 | 0 | 0 | 0 | 0 |
| | 23 | 2 | 0 | 2 | 4 |
| | 38 | 9 | 0 | 1 | 10 |
| | 23 | 4 | 1 | 6 | 11 |
| | 16 | 0 | 0 | 0 | 0 |
| | 15 | 0 | 0 | 0 | 0 |
| | 16 | 0 | 1 | 0 | 1 |
| | 15 | 10 | 1 | 3 | 14 |
| Mean | 40.63 | 3.82 | 1.63 | 1.51 | 6.95 |
| S.D. | 27.87 | 6.16 | 5.50 | 2.73 | 10.33 |

Source: Jafar (1989).

The analysis of whether the error types were independent from each other used a chi-squared test. The independence of the error types from each other can be tested using a two-by-two table and chi-squared test (Dixon & Massey, 1969). The tables allow us to investigate the hypothesis that an error type is independent of the other error types. The test gives a chi-squared with one degree of freedom.

## 4. Findings

The results indicate that there is a statistical relationship between the size of knowledge-based systems and the total number of errors, and between size and the number of particular types of error in the system. In addition, there is a positive and statistically significant relationship between the number of redundancy errors and the number of unused constructs. The number of each of the error types was positively related to the total number of errors. Finally, the existence of the three types of errors is related to the existence of other errors.

### 4.1. SIZE AND NUMBER OF ERRORS

The correlation between the number of errors and size are summarized in Table 2. The correlation was found to be statistically significantly different from zero, when system size was compared to ''number of unused constructs'', ''number of redundancy errors'' and ''total number of errors''. There was not a statistically significant relationship between of the correlation between size and ''number of illegal values''.

The results of the significance test using computer intensive methods (also in

TABLE 2
*Correlation coefficients between size and verification errors*†

| (A) System size (kbytes) | (B) Number of unused constructs | (C) Number of redundancy errors | (D) Number of illegal values | (E) Total number of errors |
|---|---|---|---|---|
| (A) System size | 0.542 | 0.321 | 0.067 | 0.512 |
| | (0.0001) | (0.025) | (0.952) | (0.0001) |
| | (0.00) | (0.05) | (0.66) | (0.01) |
| (B) Unused constructs | | 0.399 | 0.152 | 0.848 |
| | | (0.004) | (0.300) | (0.0001) |
| | | (0.05) | (0.25) | (0.00) |
| (C) Redundancy errors | | | 0.034 | 0.761 |
| | | | (0.978) | (0.0001) |
| | | | (0.87) | (0.00) |
| (D) Illegal values | | | | 0.336 |
| | | | | (0.002) |
| | | | | (0.00) |

† Comparative value of correlation is zero. Source of test: Freund (1971; p. 381).

‡ Number in top parenthesis is the statistical significance level based on assumption of normality and using $z$-values as the basis of analysis. Number in the bottom parenthesis derives from relative position in a distribution of 100 correlation coefficients generated using computer intensive statistical methods (Noreen, 1989).

TABLE 3
*Statistical significance of the relationship between size and total errors*

| Value of comparison (rho†) (comparative correlation) | $z$-value‡ | Significance level (approximate) |
|---|---|---|
| 0.00 | 3.83 | 0.0001 |
| 0.10 | 3.15 | 0.0002 |
| 0.20 | 2.46 | 0.007 |
| 0.25 | 2.10 | 0.035 |
| 0.3 | 1.73 | 0.096 |

Source: Freund (1971: p. 381).

† rho is the comparative value used in the computation of the $z$ value.

‡ $z = (1/2) * \text{Sqrt}(n-3) * \ln[((1+r)/(1-r) * (1-\text{rho})/(1+\text{rho})]$, where $z$ is from a standard normal distribution, $n$ is the total sample size, $r = 0.5115$ is the sample correlation and rho is the comparative value of the population correlation.

Table 2) were very similar to those generated under the assumption of normality. All variables that were significant at the 0.05 level or better, using the normal distribution-based test, were also significant at the 0.05 level or better using the computer intensive approach.

A detailed comparison of the statistical significance of the correlation between system size and total errors, when compared to values other than zero is summarized in Table 3. It was found that the correlation was statistically significantly different from comparative values of 0.1, 0.2, 0.25 and 0.3.

A regression equation was also developed, with size as the independent variable and number of errors as the dependent variable. The results for the slope are summarized in Table 4. The constant was found to be $-0.0048$, not statistically

TABLE 4
*Panel A: Slopes in regression equation where size is treated as the independent variable*

| Observations | Range of size | Slope |
|---|---|---|
| 1–49 | 13–126 | 0.1891 |
| 1–24 | 13–28 | 0.1813 |
| 25–49 | 30–126 | 0.1930 |

*Panel B: Significance of slopes in regression equation where size is treated as the independent variable*

| Observations | $t$ value | Slope not equal to 0 significance |
|---|---|---|
| 1–49 | 2.886 | 0.001 |
| 1–24 | 0.899 | 0.40 |
| 25–49 | 2.095 | 0.05 |

TABLE 5

*Panel A: Correlations between size and number of errors*

| Observations | Range of size | Correlation |
|---|---|---|
| 1–24 | 13–28 | 0.2521 |
| 25–49 | 30–126 | 0.4205 |

*Panel B: Relationship between correlation coefficients*

| *Correlation*: *rho*† | $z$ value‡ | Significance |
|---|---|---|
| 0.2521: 0.0 | 1.180 | 0.240 |
| 0.4205: 0.0 | 2.102 | 0.035 |

† Rho is the comparative value used in the computation of the $z$ value.

‡ $z = (1/2) * \text{Sqrt}\,(n-3) * \ln[((1+r)/(1-r) * (1-\text{rho})/(1+\text{rho}))]$, where $z$ is from a standard normal distribution, $n$ is the total sample size, $r$ is the sample correlation and rho is the comparative value of the population correlation. Source: Freund (1971: p. 381).

different than 0. The slope of the regression coefficient on the independent variable was 0.1891. The slope was significantly different than 0 at the 0.001 level.

### 4.2. PARTITIONING THE SAMPLE BY SIZE

The sample was put into two different samples of almost an equal number of elements, in order to study the potential impact of modularization. First, the data was ranked by size. Second, the median was found and used as the basis of developing two sub-samples. There were 24 elements in the first sample, ranging in size from 13 to 28 kbytes. There were 25 elements in the second sample, ranging in size from 30 to 126 kbytes.

The results are summarized in Table 5, panels A and B. The correlation coefficient between size and number of errors for the two samples were 0.251 and 0.4205, respectively. The correlation from the first group was not statistically significantly from zero. However, the correlation for the second group was statistically significant different than zero at the 0.035 level.

The results on the slopes of the regression coefficients are summarized in Table 4. The slope was 0.1813 for the first group, and 0.1930 for the second group. The regression coefficient for the first group was not statistically significantly different than 0. The regression coefficient for the second group was statistically significantly different than 0 at the 0.05 level.

### 4.3. RELATIONSHIP BETWEEN NUMBER OF DIFFERENT ERRORS AND TOTAL ERRORS

Table 2 also contains the correlations between the number of different error types. It was found that the number of redundancy errors and number of unused constructs were positively related and statistically significant at the 0.03 level or better. In addition, the number of each of the error types were positively and statistically significantly related to the total number of errors.

TABLE 6
*Two way classification–independence*

| | Unused construct error | No unused construct errors |
|---|---|---|
| Illegal value or redundancy error | 22 | 6 |
| No other illegal value or redundancy errors | 6 | 15 |
| Chi-squared = 10.29 (0.001) | | |
| | **Illegal values error** | **No illegal values errors** |
| Unused construct or redundancy error | 19 | 12 |
| No other unused constructs or redundancy errors | 3 | 15 |
| Chi-squared = 7.45 (0.01) | | |
| | **Redundancy error** | **No redundancy errors** |
| Illegal value or unused construct | 13 | 19 |
| No other illegal value or unused constructs errors | 2 | 15 |
| Chi-squared = 3.10 (0.085) | | |

### 4.4. RELATIONSHIP BETWEEN EXISTENCE OF DIFFERENT ERROR TYPES

A chi-squared approach was used to examine the relationship between the existence of one type of error and the existence of other types of errors. Two-by-two tables were formulated for each error type as compared to the two other error types, and tested using a chi-squared test. Those three tables are summarized in Table 6.

The chi-squared of 10.29 (unused constructs vs. other error types) is significant at the 0.001 level, the chi-squared of 7.45 (illegal values vs. other error types) is significant at the 0.01 level, and the chi-squared of 3.10 (redundancy vs. other error types) is significance at the 0.085 level. We reject the hypothesis of independence of the different error types from each other at the 0.001, 0.01 and 0.085 levels.

## 5. Discussion

The results in the previous section are consistent with expectations. Size is positively related to the number of errors, and size is related specifically to the number of unused construct errors and the number of redundancy errors. Further, ''larger'' systems are statistically significantly related to size, whereas ''smaller'' systems are not. In addition, the number of different types of errors are related to each other. Finally, the existence of errors of one type are related to the existence of other errors of other types.

### 5.1. SIZE AND ERRORS

The results of the regression equation are persuasive. The slope is very similar to those genreated in previous studies (0.1891 as compared to Akiyama's 0.018, 0.20, 0.27 and 0.12). Further, the constant was −0.00481, not statistically different than 0.

This paper also provides evidence that there is a very strong positive association

between the number of errors and size in knowledge-based systems. The correlation of 0.512 was significant at the 0.0001 level, under the assumptions of normality and 0.01 level with no such normality assumption. Further, as noted in Table 3, the correlation between size and total number of errors is statistically different than any (0, 0.1, 0.2, 0.25) at the 0.035 level or better.

However, the correlations were not as large as other studies such as that by Akiyama (1971). There are a number of possible reasons for that finding. First, this paper used only errors in rule-based systems. Systems that employ only a rule-based structure may not be as linearly sensitive to size as other systems. Second, the errors investigated in this study occurred in the context of an expert system shell, a higher level language that many previous studies. Thus, the results here suggest that the correlation, between size and number of errors, may not be as large in programs built using higher level languages. Third, each of the programs was developed by a single individual. Some differences may be due to the particular programmer. It is likely that different programmers have different cognitive limitations and this impacts the errors caused by the size of the system. If programs are developed by a team of programmers then that is likely to smooth many individual differences. Fourth, the methodology by which the system is being developed may influence the number of errors. There was no company sponsored or promulgated methodology that the developers were forced to use. If some developers used more structured methods, then it is likely that there were fewer errors in those systems. This would provide another source of individual differences. Fifth, the nature of the knowledge-based system being developed may influence the relationship. Some domains or problems being modeled within those domains, may be more structured and easily modularized, thus fostering the development of systems consisting of decomposable, but interrelated smaller problems. Since each of the problems would be small, there may be fewer errors in the system as a whole.

### 5.2. PARTITIONING THE SAMPLE BY SIZE

When the data was partitioned into two sub-samples, by relative size, it was found that the "smaller" systems (less than 30 kbytes) were not statistically significantly related to size (Tables 4 and 5). However, it was also found that the "larger" systems (greater than or equal to 30 kbytes) were statistically significantly related to size. The results of partitioning the sample into two sub-samples suggests that it can be critical to use modular approaches, even in rule-based systems.

### 5.3. RELATIONSHIP BETWEEN THE NUMBERS OF DIFFERENT ERROR TYPES AND THE EXISTENCE OF THE DIFFERENT ERROR TYPES

There is little in the previous literature than links the existence of number of different error types. The analysis did find one pair of the number of different error types was statistically significant (unused constructs and redundancy errors). These errors are similar, in that each results in an "excess" of concepts, so it is probably not surprising that the two were highly correlated.

In addition, the number of all error different types were statistically related to the number of total errors. Correlational analysis indicates that the number of each type of error was positively related to the total number of errors.

The chi-squared analysis suggests that the existence of one error, appears to signal the existence of other errors. In each case a statistical relationship occurred between the existence of each error type and the other errors.

Although a large majority of the error types occurred in conjunction with other error types, some error types occurred alone in some systems. This suggests that other factors may be influencing those relationships between error types, similar to those discussed above.

## 6. Summary, contributions and extensions

The purpose of this section is to briefly summarize the paper, review some of the contributions and discuss some extensions to the current paper.

### 6.1. SUMMARY

This paper provides empirical analysis of the relationship between system size and (a) the number of unused construct errors, (b) the number of redundancy errors, and (c) the total number errors. In addition, the paper provides an empirical analysis of the relationship between the number of unused construct errors, number of redundancy errors and number of illegal values, and the total number of errors.

The analysis provides evidence that indicates that knowledge-based system size is related to the number of errors, and to particular error types. In addition, the results indicate that the number and the existence of errors of one type are related to errors of other types, and can be useful to guide the testing effort.

Further, it appears that the number of errors in ''smaller'' systems are not as sensitive to system size, whereas, the number of errors in ''larger'' systems are statistically significantly related to size. This last finding suggests that it is critical to modularize rule-based systems.

Finally, these statistical results were found in spite of the fact that there was such a diverse group of systems. Normally, such diversity would bias away from finding such strong statistical results. The fact that the statistical results were so strong, inspite of that sample diversity, makes the findings that much more robust, and applicable to a wide range of settings.

### 6.2. CONTRIBUTIONS

This is the first paper to provide statistical analysis that indicates that the number of different types of errors are related. Previous research in software engineering has concentrated primarily on the study of the relationship between program size and the number of errors. The research in verification and validation has focused primarily on the development of algorithms to find these errors, and not on the descriptive analysis of these errors. This paper finds that if we have a number of one type of error, in general, we are likely to have a number of other types of errors.

This also is the first paper to provide a statistical analysis of the relationship between knowledge-based system size and number of errors. As a result, the amount of verification and validation analysis of a system for errors should be a function of the size of the system. Further, approaches such as modularization might be used to reduce the number of errors.

In addition, the findings in this paper suggest that if one type of error is found then there is a higher probability that are errors of other types. This can be critical to software quality, since it indicates that if some error types are found additional effort is likely to be necessary to find other types of correlated errors. Finding errors of one type serves to suggest that there are additional errors of other types.

### 6.3. EXTENSIONS

This paper has provided a statistical analysis of the relationship between system size and the number of errors. There are several ways to extend this research.

First, this paper used data that measured size using kbytes. There are a number of other ways to measure size including, number of rules (or frames, etc.), number of paths, number of conditions, etc. Future research could investigate the relationship between those size measures and errors.

Second, this paper used data regarding three types of errors. This analysis could be extended to other types of errors. Alternatively, rather than focusing on manifested errors, the analysis could focus on sources of errors.

Third, this paper used data based on a single shell (M.1), future research could examine data derived from the use of other shells or languages. The use of different shells or languages could result in different types of errors and different types of relationships between the errors and the system size.

Fourth, the availability of data on knowledge-based systems is currently limited. Analyses of this type could be expanded to include a broader base of systems if there was a database of knowledge-based systems available for analysis. A central clearing house of knowledge-based systems could also facilitate other types of analyses such as a comparison of the ability of different types of verification tools to find errors.

Fifth, this paper is a first step, the scope of this paper was the relationship between size and errors. However other variables that influence the number of errors could be studied. For example, the use of structured methods could be examined to determine the impact on errors in systems. Alternatively, characteristics of problem types could be investigated to determine whether or not they influenced the number of errors.

The author wishes to acknowledge the comments of Lance Miller and Robert Plant on an earlier version of this paper. The author also wishes to acknowledge the extensive comments of the referees on earlier versions of this paper.

## References

Akiyama, F. (1971). An example of software system debugging. *Proceedings of the IFIP Congress '71,* Ljubljana, Yugoslavia. Montvale, NJ: American Federation of Information Processing Societies.

Brown, C., Nielson, N. & Phillips, M. (1993). Evaluating expert systems in financial domain. *International Journal of Intelligent Systems in Accounting, Finance and Management,* **2,** 81–100.

Chang, C., Combs, J. & Stachowitz, R. (1990). A report on the expert systems validation associate (EVA). *Expert Systems With Applications,* **1,** 217–230.

CONTE, S., DUNSMORE, H. & SHEN, V. (1986). *Software Engineering Metrics and Models.* New York, NY: Benjamin/Cummings Publishing.
DAVIS, R. & LENAT, D. (1982). *Knowledge-based Systems in Artificial Intelligence.* New York, NY: McGraw-Hill.
DIXON, W. & MASSEY, F. (1969). *Introduction to Statistical Analyiss.* New York, NY: McGraw-Hill.
FREUND, J. (1971). *Mathematical Statistics.* Englewood Cliffs, NJ: Prentice-Hall.
GUPTA, U. (1991). *Validating and Verifying Knowledge-based Systems.* Washington, DC IEEE Press.
HOGARTH, R. (1987). *Judgment and Choice.* Chichester: John Wiley.
JAFAR, M. (1989). *A tool for interactive verification and validation of rule-based expert systems.* Ph.D. Thesis, University of Arizona, AZ, USA.
NGUYEN, T., PERKINS, W., LAFFERTY, T. & PECORA, D. (1987). Knowledge-base verification. *AI Magazine,* **8,** 65–79.
NOREEN, E. (1989). *Testing Hypotheses Using Computer Intensive Methods.* New York, NY: John Wiley.
O'KEEFE, R. & O'LEARY, D. (1992). Expert System verification and validation. *Artificial Intelligence Review,* **16,** 25–60.
O'LEARY, D. (1987). Validation of Expert Systems. *Decision Sciences,* **18,** 468–486.
PREECE, A. (1990). Towards a methodology for evaluating expert systems. *Expert Systems,* **7,** 215–223.
STACHOWITZ, R. & COMBS, J. (1987). Validation of knowledge-based systems. *Proceedings of the Twentieth Annual Hawaii Conferenece on Systems Sciences,* pp. 686–695. Hawaii.
SUWA, M., SCOTT, A. & SHORTLIFFE, E. (1982). Completeness, and consistency in rule-based systems. *AI Magazine,* **3,** 16–21.
TEKNOWLEDGE (1989). *M.1 Reference Manual.* Palo Alto, CA.
VELLEMAN, P. & HOAGLIN, D. (1981). *Applications, Basics and Computing of Exploratory Data Analysis.* Belmont, CA: Duxbury Press.