A formal method for cost and accuracy trade-off analysis in software assessment measures

Ulrik Franke, Pontus Johnson, Robert Lagerström, Johan Ullberg, David Höök, Mathias Ekstedt, Johan König
Industrial Information and Control Systems
Royal Institute of Technology
100 44 Stockholm, Sweden
{ulrikf, pj101, robertl, johanu, davidh, mek101, johank}@ics.kth.se

Technical solution paper

Abstract—Creating accurate models of information systems is an important but challenging task. It is generally well understood that such modeling encompasses general scientific issues, but the monetary aspects of the modeling of software systems are not equally well acknowledged. The present paper describes a method using Bayesian networks for optimizing modeling strategies, perceived as a trade-off between these two aspects. Using GeNi, a graphical tool with the proper Bayesian algorithms implemented, decision support can thus be provided to the modeling process. Specifically, an informed trade-off can be made, based on the modeler’s prior knowledge of the predictive power of certain models, combined with his projection of their costs. It is argued that this method might enhance modeling of large and complex software systems in two principal ways: Firstly, by enforcing rigor and making hidden assumptions explicit. Secondly, by enforcing cost awareness even in the early phases of modeling. The method should be used primarily when the choice of modeling can have great economic repercussions.

Index Terms—Software measurement, Bayesian networks, value of information, cost-benefit analysis.

I. INTRODUCTION

Managing information systems today is a complex business. To achieve effective and efficient system development and management, it is essential to be able to assess the current status of system properties such as availability, performance, security, and modifiability, as well as predict their values in different future scenarios. Estimation of these properties, however, is a great challenge.

First, a great number of factors influence system properties. Second, the factors are intertwined in a complex manner. The researcher or practitioner who sets out to model these interdependencies thus inevitably faces a discomforting amount of modeling choices, all of which to some extent influence the ability of the final assessment framework to provide accurate decision support for management decisions. Furthermore, all modeling choices represent a cost in terms of collecting the information needed for actually using the model. This cost, whether expressed in money, effort, or time, must be kept under control, lest the entire modeling effort be misguided.

This paper addresses a selection of modeling uncertainties and problems, encountered when system property assessment and prediction frameworks are devised. As opposed to other publications addressing similar issues, such as [1], [2], [3], our approach adopts a Bayesian formalism for expressing these uncertainties. The application of Bayesian networks in a graphical environment for decision support enables us to create the most efficient model, given the available information on uncertainties and the cost of data collection. Even if the data available before modeling is scarce, the proposed model forces the modeler to make her implicit assumptions explicit, and her decisions more transparent, even to herself. Furthermore, cost-efficiency is taken into consideration in the early phases.

The rest of this paper is structured as follows. In section II we outline some general modeling problems encountered when creating assessment frameworks. Furthermore the running example of the paper – complexity as a measure of cost of change – is introduced. In section III, Bayesian networks are introduced, paving the ground for the subsequent formalization of the problems identified, given in section IV. Section V describes how the theory of value of information and diagnosis in Bayesian networks can be used to find an optimal modeling strategy when creating analysis frameworks for decision support. Finally, section VI concludes the paper.

II. SIX GENERAL PROBLEMS OF a priori ASSESSMENTS

Many system properties – availability, performance, security, and modifiability, to name a few – share the elusive feature that while they are easy to define a posteriori, i.e. after system implementation, such definitions give precious little guidance on how to ensure them a priori, i.e. before system implementation. For example, measuring the cost of change of a system a posteriori is mere book-keeping. But assessing it beforehand is a formidable task. Such assessment must be carried out by measuring variables available prior to the modification. The literature, of course, provides a wealth of different methods to make such cost of change assessments. But the modeler cannot afford to employ them all. She requires accurate and cost-efficient decision support. The same holds true for the assessment of other system properties, even though we shall use cost of change as a running example for the remainder of the present paper.

Figure 1 illustrates a generic measurement process, as exemplified by our running cost of change example. Six key problems, numbered in the figure, need to be addressed by the modeler:
1) The choice of an a priori measurement quantity is the problem of finding a measure (complexity) that correlates accurately with the sought a posteriori quantity (cost of change).

2) Definitional uncertainty must be handled. Most concepts (complexity) can be interpreted in many different ways (Halstead complexity measures, Cyclomatic Complexity, Lines of code, etc.).

3) Measurement devices, ranging from software tools to expert estimates, are necessary and crucial instruments, but introduce further uncertainties.

4) The selection of appropriate scales affects precision and imposes constraints on which statistical operations are permissible to perform on the data.

5) Discretization of measurement variables simplifies measurements and maps them onto the desired scales, but only at the cost of lost accuracy.

6) The overall accuracy of the model must be weighed against the cost of performing the measurement. Out of several models, the most cost-efficient one always ought to be selected.

We now proceed to discuss each problem in slightly greater detail.

A. The choice of an a priori measurement quantity

The basic modeling aim is to ensure the correlation between a priori measurements and a posteriori properties of interest. For any such property, there are numerous candidates for a priori measurements, each with its own set of advantages and disadvantages.

Our running example makes this clearer: the cost of change, i.e. the cost for modifying or amending a software program, is often estimated by the software complexity. Software complexity in itself is just a measure of certain properties of the program code. No complexity measures make any mention of the cost or time spent creating or modifying this code. Nevertheless, it is generally assumed that complex computer programs are difficult to maintain and modify. This is the causal relation that, supposedly, binds together the a priori metric of complexity with the a posteriori system property of cost of change.

The following three complexity measures will be used as examples throughout the remainder of this paper:

1) The Halstead measure is based on the numbers of operators and operands, distinct as well as total, found in the code [4], [5].

2) The number of lines of code (LoC), i.e. the software size, is a crude estimate of complexity. LoC is usually defined excluding comments and blank lines [5].

3) Cyclomatic complexity was introduced by McCabe and is based on graph theory. It measures the number of linearly independent paths through a source code [6].

These concepts all differ in measurement cost and accuracy. In simple cases, there is no trade-off: small programs can be precisely measured using free software off the Internet, such as [7]. In more complex cases, such as the hundreds of systems running in corporate environments, this is not an option. When it comes to these software systems, measurement
using statistical sampling or expert estimations is necessary to avoid prohibitive costs.

B.  Definitional uncertainty

Settling for a theoretical a priori measure is only a first modeling step. Data collection often requires concerted efforts from many investigators, who need a clear common picture of the concepts at hand. Similarly, the literature is often less than clear on the exact nature of the concepts described or used in research. Both of these phenomena contribute to the importance of definitional uncertainty.

As illustrated by the example in Figure 2, definitional uncertainty must be at a minimum before precise measurements can be carried out.

C. Measurement devices

Representational theory of measurement [8, 9] is about mapping an empirical relational system, i.e. observable reality, into a mathematical model, a numerical relational system. As put in [10], measurement is “the correlation with numbers of entities which are not numbers”. This correlation aspect of measurement is one of our foremost interests for the purpose of this paper. To the modeler, the use of a particular measuring device will affect the model’s theoretical performance by its inherent accuracy limits.

One of the most convenient methods for collecting information about certain characteristics of a system is to consult an expert in the field. In this paper, we consider the use of expert estimation to be a kind of measurement, on a par with information as is necessary in a given context.

The use of expert estimation also highlights the cost aspect of measurement. In large and complexly intertwined software systems, measurement programs are not a feasible option. Here, expert estimation provides a superior option, even though it may score lower from a pure correlation perspective.

D. The selection of appropriate scales

That the choice of scales has non-trivial repercussions on scientific models has been known at least since Stevens’ seminal 1946 paper [11]. The fact that different scales permit different statistical operations creates constraints on the permissible structure of assessment frameworks. Following our example, we note that interval scale concepts such as LoC, Halstead and cyclomatic complexity might be transformed onto ordinal scales when discretized. One such possible ordinal scale is \{Low, Medium, High\}. The loss of information is evident, but often unavoidable for instance for expert estimates. Table II lists a few different scales.

\[
\begin{align*}
\text{Scale} & \\
\{\text{Low, Medium, High}\} & \\
\{1, 2, 3, 4, 5\} & \\
\{0, 1, 2, \ldots\} = \mathbb{N} & \\
\mathbb{R} & \\
\ldots &
\end{align*}
\]

TABLE II
SOME MEASUREMENT SCALES.

E. Discretization of measurement variables

A special case of the preceding problem is discretization of properties that are continuous. In the context of measurement, discretization not only approximates a continuous phenomenon, but also enables mapping onto a predetermined scale. In general, of course, such mappings entail a loss of information. The general trade-off thus amounts to achieving the simplifications of discretization while retaining as much information as is necessary in a given context.

In our maintainability example, the cost of change can be estimated based on approximate, discrete versions of either LoC, the Halstead measure, or the cyclomatic complexity. A discretized measure might be cheaper and easier to obtain.

Consider as a simple example the discretization of the Halstead measure. We need to map the positive real numbers onto, say, three intervals, the first with a lower bound at zero and the last interval unbounded upwards. Table III provides an example of such a discretization.

\[
\begin{align*}
\text{Sample discretization of } \mathbb{R}^+ & \\
\text{Low} & \leftarrow 0 \leq x \leq 3.3 \\
\text{Medium} & \leftarrow 3.3 < x \leq 14.4 \\
\text{High} & \leftarrow 14.4 < x < \infty \\
\end{align*}
\]

TABLE III
AN EXAMPLE OF A DISCRETIZATION OF THE HALSTEAD MEASURE.

F. Accuracy vs. cost

The trade-off between accuracy and cost is a generalization of the five previous ones. In fact, each of the preceding problems is a problem precisely by virtue of being a trade-off. Aspects of measurement that do not exhibit this feature are trivial: if accuracy can be increased without additional cost,
or if cost can be decreased with no loss of accuracy, there is no doubt as to the reasonable course of action.

Sometimes, this is the case. As mentioned earlier, certain qualities of small programs running on standard operating systems can be precisely measured with free software available on the Internet. In the case of large and numerous software systems running on proprietary operating systems in demanding industrial settings, this is not the case. As the trade-off arises, the modeling problem becomes non-trivial, and the need for a non-arbitrary resolution arises.

It is not obvious, however, that the five preceding problems can be subsumed by the sixth into a single trade-off. To show that this is indeed the case, we will make use of the mathematical formalism of Bayesian networks, which is briefly introduced in the next chapter.

### III. BAYESIAN NETWORKS

Friedman et al. [12] describes a Bayesian network, \( B = (G, P) \), as a representation of a joint probability distribution, where \( G = (V, E) \) is a directed acyclic graph consisting of vertices, \( V \), and edges, \( E \). An acyclic graph is a graph where there are no cycles.

In a Bayesian network, the vertices denote a domain of random variables \( X_1, \ldots, X_n \), also called chance nodes. Each chance node, \( X_i \), may assume a value \( x_i \) from the finite domain \( \text{Val}(X_i) \). The advantage of the graph representation is that it provides a compact way of expressing the dependency relations between the random variables, i.e. which variables are conditionally independent given another variable. Each edge denotes a causal dependency between its nodes.

In order to specify the joint distribution, the respective conditional probabilities that appear in the product form

\[
P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|\text{Pa}(X_i)).
\]

must be defined [12]. The second component \( P \) describes distributions for each possible value \( x_i \) of the chance node \( X_i \), given the values \( \text{pa}(X_i) \) of its causal parents \( \text{Pa}(X_i) \). These conditional probabilities are represented in matrices, here forth called Conditional Probability Matrices (CPMs).

Using a Bayesian network, it is possible to answer questions such as what is the probability of variable \( X \) being in state \( x_1 \) given that \( Y = y_2 \) and \( Z = z_1 \). An example of a Bayesian network and a CPM representing the chance nodes \( X, Y, \) and \( Z \) and how these relate is shown in Figure 3. The CPM next to the network answers the question stated above.

More comprehensive treatments on Bayesian networks can be found in e.g. [13], [14], [15] and [16].

### IV. EXPRESSING THE PROBLEMS IN BAYESIAN NETWORKS

Returning to the problems described in section II, we can now attempt to put them in Bayesian form. This formalism enables us to quantify the uncertainties involved, thus allowing us to treat them mathematically and make modeling trade-offs in a non-arbitrary manner.

![Fig. 3. A Bayesian network and the conditional probability matrix for the chance node X given Y and Z.](image)

![Fig. 4. A Bayesian network for various assessments of LOC, with sample CPMs for measurement and ordinal expert estimation. Similar CPMs hold between consultant measurement and discretization.](image)

To address this issue, we shall again make use of our running cost of change example. Consider the Lines of Code definition of complexity. The simple Bayesian network of Figure 4 relates the LoC notion to various measurements that can be performed to assess it. The arrows reflect causalities: each assessment depends on LoC itself, but the true value can, we assume, never be fully accessible at a reasonable cost. The dependencies – specifically, the uncertainties – between the nodes are defined by the CPMs associated with each of them. Specifically, we have the following nodes:

1) LoC. This node reflects the underlying actual value, deemed inaccessible (at reasonable cost). By introducing structures similar to that of Figure 4 for Halstead and cyclomatic complexity, the problem of choosing an a
priori measurement quality, discussed in section II.A is modeled.

2) **Measurement of LoC.** This reflects a measurement performed by the modeler herself.

3) **Consultant measurement of LoC.** This reflects a measurement performed by someone who is not the modeler. Along with the previous node, this is where the problem of definitional uncertainty, discussed in section II.B, occurs.

4) **Ordinal Expert Estimation of LoC.** This quantity reflects the use of expert estimates, as discussed in section II.C.

5) **Discretization Expert Estimation of LoC.** This quantity reflects the discretization of measurement variables, as discussed in section II.E.

The selection of appropriate scales, as discussed in section II.D, is a general problem reflected within the entire Bayesian structure by the contrast between the available modeling nodes. The representation of the trade-off problem discussed in section II.F will be the topic of section V.

As illustrated by the sample CPMs of Figure 4, the Bayesian formalism enables a quantification of the uncertainty associated with measurement. At the stage of theoretical modeling, this provides an opportunity for the researcher to sort out her beliefs and make them consistent. Thus, modeling choices can be made rationally, with a consistency check on the beliefs on the relative merits of various modeling and measurement approaches. E.g. a direct measurement of Lines of Code should, reasonably, have higher probability of being accurate than an expert estimation of LoC, albeit at a higher cost.

As should be evident from the samples of Figure 4, the proposed CPMs for the various measurement procedures are quite similar. The difference is modeled primarily as a quantitative one – the level of noise diverging from a pure diagonal matrix – rather than a qualitative one. The main benefit of this is that the five disparate problems listed in section II.F all are modeled in a commensurable way. This facilitates trade-offs and prioritizations between them.

Now, in our running example, the prime concern is to create a feasible model for predicting overall cost of change. Estimation of cost of change by measuring LoC, Halstead or cyclomatic complexity will require similar networks but with different CPMs. Then, the CPM for cost of change contains information about which of the three measures that best estimates the cost. However, there are two separate aspects that this CPM must handle: (i) since the different measures have different scales, the CPM needs to contain a compensating scaling factor, and (ii) since the different measures have different predictive abilities, the CPM needs to reflect a prediction factor.

Now, (i) is tedious, but not very interesting from a theoretical point of view. In the following, we will disregard it and focus on (ii), which is where the real job is being done. For the purpose of our example, as illustrated in Figure 5, (ii) is reflected by a weighted distribution, where the LoC, Halstead, and cyclomatic complexities (normalized by (i)) have been assigned the relative weights 1, 5, and 7, respectively. These figures do not reflect a research contribution, but have been chosen somewhat arbitrarily for the purpose of the example.

Yet another possibility is to assess the complexity directly – with no further details of how the concept should be interpreted – by an expert estimate. Here, the crucial problem is to define exactly what the expert is referring to when estimating the complexity. For instance, a large number of LoCs might result in an estimate of high complexity from the expert, independent of other measures. This could result in an inaccurate complexity estimate, since LoC in some respects relevant to cost of change is probably the least accurate of our three modeling alternatives. The CPM representing expert estimation of complexity in the present model combines a mapping similar to that of ordinal expert estimation with an unweighted distribution of the three modeling alternatives. The CPM associated with the expert complexity estimation thus defines the probabilistic dependence of an expert’s answers on the actual complexities – LoC, cyclomatic and Halstead – were they accessible.

Having thus outlined a Bayesian representation of our sample modeling problem, we now proceed to discuss how this formalism can be used to make modeling decisions.

V. **Estimating the Quality of the Models**

Once a Bayesian representation of the modeling choices has been created, this formalism can be employed to guide the modeling. Basically, a trade-off is sought between (i) the value of a certain model (in our example; how well a model predicts cost of change), and (ii) the cost or difficulty in using that model (in our example; the cost of collecting different sorts of data). Precisely this trade-off can be systematically tackled using the theory of diagnosis in Bayesian networks [14]. Following a brief introduction of this concept, we give an example of a modeling trade-off using our running example. Actual calculations are performed in the software GeNIe [17], a graphical environment for building decision models created by the Decision Systems Laboratory at the University of Pittsburgh. [18] is a detailed description of the implementation of the diagnostic functionality of GeNIe.

A. **Value of information**

Let a diagnostic probability network (DPN) be defined as a Bayesian network where at least one random variable \( H \) is a **hypothesis variable** (in our case the cost of change) and at least one random variable \( T \) is a **test variable** (in our case the variables of the model that we potentially can choose for measurement, e.g. one of the LoC assessment methods).

Let \( \mathcal{H} \) denote the set of all hypothesis variables, and \( \mathcal{T} \) the set of all test variables. Furthermore, each test \( T \in \mathcal{T} \) has a cost function \( \text{Cost}(T) : \mathcal{T} \to \mathbb{R} \). If a test is free, the associated cost is set to zero. Also, each hypothesis \( H \) has an associated value function, \( V(P(H)) : [0, 1] \to \mathbb{R} \), defined on the probability of the hypothesis.

Given a DPN, we have the expected value \( EV \) of performing a test \( T \in \mathcal{T} \):

\[
EV(T) = \sum_{T'} \text{Cost}(T') \cdot V(P(T'H))
\]

where the sum is over all combinations of \( T \) and \( H \) such that \( P(T'H) \neq 0 \).

...
Weighted distribution:
LoC: 1
Halstead: 5
Cyclomatic: 7

Unweighted distribution:
LoC: 1
Halstead: 5
Cyclomatic: 7

Fig. 5. A Bayesian network reflecting various modeling options for cost of change assessment.

\[ EV(T) = \sum_{t \in T} V(P(H | t)) \cdot P(t) \]

To make an informed decision, we also need to account for the expected outcome of not performing the test \( T \). We therefore introduce the expected benefit \( EB \):

\[ EB(T) = EV(T) - V(P(H)) = \sum_{t \in T} V(P(H|t)) \cdot P(t) - V(P(H)) \]

Still, however, no connection has been made to the cost of the test. This is remedied by the test strength \( TS \):

\[ TS(H, T) = \frac{EB(T)}{V(P(H))} - K \cdot \text{Cost}(T) \]

Here we have introduced the coefficient \( K \), reflecting the relative importance of the expected benefit versus the cost of the test, i.e. how much the cost of the test is to weigh relative to the benefit it provides. Furthermore, \( K \) is necessary for combining the differently scaled variables of cost and expected benefit. \( K \) is an expression of the modeler’s priorities, and will depend heavily on the context. For example, cost (including time and effort) might be less important in a one shot academic case study than in an industrial model to be reused hundreds of times.

The definition of the value function still remains. To optimize the test selection with respect to multiple hypotheses [18] introduces a function based on the marginal (rather than the joint) probability between hypotheses called Marginal Strength 1 (\( MS1 \)):

\[ MS1(P(F)) = \left( \sum_{f \in F} (f - 0.5)^2 - n_F \right) \cdot \frac{1}{n_F} \]

This function reflects the diagnostic value of information with respect to several potential target states \( f \), such as different outcomes of tests. \( n_F \) is the total number of target states. This test selection function is convex with a minimum at \( 1 - n_f \) and maxima at 0 and 1. The value function that we are looking for now becomes the sum of the marginal strength for all target states:

\[ V(P(F)) = \sum_{f \in F} MS1(P(f)) \]

B. Measurement costs

The diagnosis theory described above does not only take the value of information obtained from each test into account: It also considers the cost for obtaining it. In order to make use of the method, it is thus necessary to assign a cost to each potential test. The costs assigned here do not have any specific unit such as dollars. However, they do represent numerical and comparable measures of the effort needed to obtain data using the various procedures.

This trade-off is made based on the modeler’s prior knowledge of the predictive power of certain models, combined
The relative costs of each example measurement procedure.

<table>
<thead>
<tr>
<th>Complexity measure</th>
<th>Test</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. LoC</td>
<td>Measurement (through external part)</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Consultant measurement</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Discretization through expert estimation</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Ordinal expert estimation</td>
<td>1</td>
</tr>
<tr>
<td>2. Cyclomatic complexity</td>
<td>Measurement (through external part)</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Consultant measurement</td>
<td>160</td>
</tr>
<tr>
<td></td>
<td>Discretization through expert estimation</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>Ordinal expert estimation</td>
<td>60</td>
</tr>
<tr>
<td>3. Halstead measure</td>
<td>Measurement (through external part)</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>Consultant measurement</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>Discretization through expert estimation</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Ordinal expert estimation</td>
<td>30</td>
</tr>
<tr>
<td>4. All of the above</td>
<td>Expert estimation of complexity</td>
<td>45</td>
</tr>
</tbody>
</table>

Table IV

The relative costs of each example measurement procedure.

with his projection of their costs. The elicitation of this prior knowledge is beyond the scope of the present paper. It should be noted, however, that there exists abundant literature on the general correlation of different a posteriori measures with a priori metrics. A thorough review is found in [19].

Table IV lists each observable node along with its attributed cost. Roughly, the costs are set by the principle "you get what you pay for", i.e. the higher the expected accuracy, the higher the observation cost. We have set the Halstead measure as the most reliable measure of complexity and LoC as the least reliable.

C. Diagnosis in GeNIe

Figure 6 illustrates GeNIe’s test ranking user interface. The value function defined above is implemented in GeNIe’s multiple cause module. The modeling alternatives available are listed to the right, ranked by their diagnostic value as given by the value function. Again, the concrete alternatives are familiar from our running maintainability example The entropy/cost ratio set above the list corresponds to the coefficient $K$ defined above.

As is evident from Figure 6, the suggested course of action in our running example is to model Halstead complexity, with a measurement as its test. The interpretation is that given our judgment of the correlation between cost of change and various complexity measures, and given our measurement costs, this is the optimal modeling alternative.

The example detailed above should provide ample proof of concept that the diagnosis algorithm enables us to make an informed modeling decision in order to obtain most "bang for the bucks". In the example case, the result sought was information on which complexity model provides the most information about the cost of change. Clearly, the method could be generalized to other settings.

Now, even though it is clearly difficult to obtain the information necessary for applying the method proposed, there are two strong arguments for its validity and applicability:

1) The modeling decision must, regardless of method, be made based on the information available to the modeler. The present method forces the modeler to make this knowledge explicit, by putting numbers on it. While the merit of the numbers as such may be questioned, their mere existence forces the modeler to be consistent in her judgments, and put controversial beliefs to scrutiny. Indeed, the present method encourages and assists such critical reviewing, by providing tools that enable the modeler to see precisely which of her assumptions are critical.

2) The present method forces the modeler to think about costs. While costs are easy to disregard in the early stages of modeling, they are always crucial to any practitioner who has a limited budget. Therefore, theoretical modeling guidelines that require cost consideration at the earliest possible stage should have a better chance of producing models that are viable in the industry.

Together, these two considerations provide a good argument in favor of the method proposed.

VI. Conclusions

The present paper has scrutinized a number of general problems related to measurements and assessments of large software systems. It has been argued that these problems are not in general given sufficient thought when making decisions on how to model software systems.

The running example used in the paper should provide ample proof of concept regarding the method proposed. However, due to its somewhat laborious nature, care should be taken when deciding how and when to apply it.

An example of a setting where the application of the proposed method should prove useful is the construction of standards. The ISO-9126 Software engineering – Product quality standard [20], for instance, has a profound impact on the industry by virtue of being an ISO standard. But it is far from clear that it has been put together considering the trade-off between measurement precision and measurement cost. The ISO-9126 defines the measure of "Modification complexity" as $T = \text{Sum}(A/B)/N$, where $A =$ Work time spent to change, $B =$ Size of software change, and $N =$ Number of changes. The modification complexity is thus measured in units of time. For interpretation guidance, the standard offers the explanation that "The shorter is the better or the required number of changes were excessive".

While this explanation does hint at a sense of cost – excessive changes are costly – it does not appear that the authors of the standard have considered the second order cost of modeling, but merely the first order cost of bad modifiability. Our purpose is not to call this particular metric into question. We do note, however, that all metrics defined in standards, whether by ISO or by individual companies for internal use, have a tremendous impact on the operations and costs of
anyone using the standard. Because of the close link between modeling decisions and costs, not only measurement precision, but also measurement cost, should be taken into consideration when constructing standards for measurement. The effort spent doing so is a sound investment. The method proposed in this paper is a structured approach to this problem.

REFERENCES


Fig. 6. A ranking of tests in a diagnostic probability network, using GeNIe.