Analysis of Built-In Self-Tests for Electronic Control Units

K. Wojtek Przytula[1], David Allen[1], Tsai-Ching Lu[1], Noel Anderson[2], and Jason Wanner[2]

[1] *HRL Laboratories, LLC., Malibu, CA, 90265, USA*

{wojtek,dlallen,tlu}@HRL.com

[2] *Phoenix International – A John Deere Company, Fargo, ND, 58102, USA*

{AndersonNoel,WannerJason}@JohnDeere.com

ABSTRACT

This paper addresses the problem of system design for diagnosability. Specifically, it focuses on design of built-in self-tests (BISTs) for subsystems based on electronic control units (ECUs). The BISTs play a major role in diagnosis of the systems and in particular in determining if the failure is in the ECU or externally in the sensors, detectors, or actuators. The design of BISTs involves a tradeoff between the diagnostic benefit gained by the presence of a BIST versus cost of providing it in the system.

We describe a systematic methodology and software tools for quantitative tradeoff analysis of BISTs. The methodology utilizes graphical probabilistic models (Bayesian networks) to represent the diagnostic properties of the system and structured equation models to perform cost-benefit analysis. The models are developed from the knowledge of the systems (i.e. documentation and/or subject matter experts) and from data. The methodology is suitable for design of BIST for a broad range of systems. We illustrate the use of it on an example of a ECU-based subsystem for control of agricultural machinery.*

1. INTRODUCTION

In many systems used in transportation, communication, aerospace, manufacturing and medicine the number of electronic control units (ECUs) grows as the functionality of the system increases. Moreover, the complexity of the ECUs and the subsystems that they control is also growing.

Efficient and accurate diagnosis of failures in ECU-based subsystems is very difficult. Consequently many component replacements in the systems, in particular ECU replacements, result in no-defect-found (NDF) when the replaced components are later tested. This in turn leads to a very high system warranty cost and reduced customer satisfaction.

There is a need for systematic methodology and software tools for design of built-in self-tests (BISTs) for ECU-based systems. The methodology should support trade-off analysis of quantitative measures of the BIST's diagnostic benefits, i.e. improvements in failure detectability and disambiguation, versus the costs of introducing BISTs into the systems, i.e. nonrecurring and recurring engineering, such as design, validation, software development and hardware extensions.

This paper describes a model-based methodology for quantitative BIST evaluation. The methodology is suitable for broad range of systems, however our focus is on subsystems involving ECUs.

We use graphical probabilistic models to represent the diagnostic properties of the system and for cost-benefit analysis. The diagnostic models are Bayesian networks (BNs), also called belief networks (Pearl 1988). They consist of a qualitative part, namely a directed acyclic graph, and a quantitative part, namely parameters in the form of probabilities. From a mathematical point of view they constitute a joint probability distribution over a set of random variables. The variables represent: components and subsystems of the system and diagnostic observations, e.g. BIST results. The probabilistic diagnostic model allows us to account in our analysis for imperfect observations, i.e. BISTs with limited failure detectability and disambiguation. This results in more accurate diagnosability analysis of them.

The diagnostic models are developed from the knowledge of the systems (i.e. documentation and/or subject matter experts) and from data. The model analysis is performed by processing the diagnostic models using model evaluation algorithms (Przytula et al., 2003). The model creation and evaluation are supported by software tools (Przytula et al., 2006).

The BIST cost-benefit trade-off analysis uses the results of the model evaluation. It is performed with the help of another graphical representation called structured equation models (Lu et al., 2000). The equations represent relations between the BIST costs and benefit parameters. They constitute a convenient form of encoding of the relations.

We have demonstrated the efficacy of our model-based approach to BIST evaluation on a real ECU subsystem, the Header Height Control (HHC) subsystem of the Self-Propelled Foraging Harvester (SPFH). We developed a BN diagnostic model using expert-knowledge and data and then performed diagnosability analysis. The results of the analysis and expert knowledge of the system were used to propose several additional BISTs for the subsystem. Each of the proposed BISTs was added to the diagnostic model and diagnosability analysis of the modified system was performed. The analysis was followed by cost-benefit analysis of each of the BISTs separately. The results indicated that addition to the subsystem of each of the proposed BISTs, except for one, resulted in a net benefit.

Selected aspects of the model-based approach presented in this paper were described in our earlier publication, which discusses its application to development of computerized diagnostic assistants for complex transportations system, (Przytula and Smith, 2004).

Many authors have addressed the issue of economic effects of BISTs. For example, (Lu and Wu, 2000) discussed the cost and benefit models for logic and memory BIST. They included in their analysis impact of BIST on design verification and test development time. (Ungar and Ambler, 2001) discussed the cost-benefit of BIST for electronics at the integrated circuit, board, and system level. (Feldman et al., 2008) addressed return of investment into BIST for prognostics and health management.

This paper consists of five sections. Section 2, following this introduction, presents material on development of BN diagnostic models. In addition to general information on BN models and their creation, it contains discussion of the BN model for the HHC subsystem. Section 3 is devoted to diagnosability analysis of the ECU-based systems. It also presents a general introduction to our approach as well as its application to the HHC subsystem. In Section 4 we present the cost-benefit analysis of BIST using structured equations. The results for several different BISTs for HHC are included. The paper results are summarized in Section 5.

2. DEVELOPMENT OF DIAGNOSTIC MODELS

Our approach to BIST evaluation is based on diagnostic models. This section presents the diagnostic modeling approach and its application to the development of a model for a real-world vehicle subsystem.

2.1 Bayesian Network Models

We are using a specific form of graphical probabilistic models referred to as Bayesian networks (BN) or belief networks (Pearl, 1988). These models capture relations between failure modes of components and outcomes of diagnostic observations (e.g. BIST results). They describe how system failures are observed via the diagnostic observations. They do not simulate system operation and are different from physics models. In general, they are simpler and easier to create than physics models, but contain all the necessary information for model based diagnosis.

BNs are directed acyclic graphs, which consist of nodes and directed links. Nodes represent components and subsystems as well as diagnostic observations. There are two or more states associated with each node. The states of the component or subsystem nodes represent failure modes and a correct operation mode. Each node has at least one failure mode state and one OK state. The states of observation nodes encode different outcomes of the observation, i.e. different BIST results including the “pass” or OK result. A link between a component node and a subsystem node indicates that the component failure affects the subsystem. A link from a component node to an observation node indicates that failures of the component affect the outcome of the observation. A failure of a given component in a typical system affects only a small subset of observations.

The BN graphs are annotated with probabilities. In mathematical terms it means that the nodes of the BN are random variables with specified probability distributions. The root nodes in the graph, i.e. the nodes that have only outgoing links, are annotated with prior probabilities. In diagnostic models these nodes typically represent system components and their prior probabilities reflect frequency of

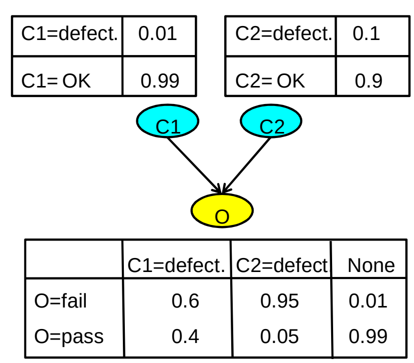

Figure 1. Simple Bayesian network diagnostic model with two component nodes and one observation node. Each node has two states. The conditional probability table assumes the use of a "Noisy-Or" node.

component failures. The nodes that have one or more parents are annotated with conditional probabilities. The probabilities indicate how likely each state of the child node is given the states of the parent node(s). The child nodes in the diagnostic models are typically subsystem and observation nodes. The conditional probabilities are used to express uncertainty of the observation state given the failure states. The diagnostic BN models allow for imperfect observations, e.g. tests with nonzero false positives and negatives.

A complete BN model constitutes a joint probability distribution over the random variables represented by the nodes. Figure 1 shows a simple diagnostic BN with two component nodes and a single observation node. Each node has only two states. The tables next to the nodes contain network parameters, i.e. prior and conditional probability values. The observation node in Figure 1 is represented as a Noisy OR node (Pearl, 1988), which is reflected in its conditional probability table. The table has a separate column for each failure mode of the two component nodes C1 and C2. The conditional probability values in the table indicate how well the observation detects the failures (e.g. defect of C1 is detected by observation O only 60% of time, whereas defect of C2 is detected almost perfectly – 95%) and how many times the observation indicates failure even if there is no C1 or C2 failure present (e.g.1% of false alarms).

A BN diagnostic model for a particular system can be developed from domain knowledge only, data only or a combination of knowledge and data. The domain knowledge can be acquired from system documentation, service manuals or from experts. The data may be in form of detailed repair records, which contain results of the observations and repair actions, or in form of average failure statistics. The most typical model development scenario assumes a combination of sources in the form of knowledge and data.

The model development process begins by enumerating the components and their failure modes. The level of granularity of the component is determined by diagnostic requirements and repair practice. It is typically assumed that the component is a line replaceable unit (LRU). Given the list of components we determine the pertinent observations. The observations may include symptoms, BIST results, inspections and manual tests. We are interested in BIST evaluation for design and will include in our models only BIST-based observations. The BN model with all the forms of observations included in it can be used to support diagnosis in service operations. In order to be able to evaluate potential additional BISTs, we create multiple models of the system: with and without the nodes representing the new BIST under consideration. In practice, BISTs are represented as algorithms executed on-board, which produce a diagnostic trouble code (DTC) in the system archive. Some BISTs may indicate multiple failure modes, which are recorded as DTC with failure mode indicator (FMI).

The parameters of the model can be estimated by experts or acquired from data. The prior probabilities of component failures can be approximated using frequency of failure statistics, which are typically available. The data for conditional probabilities are often not available initially. Thus, the conditional probabilities have to be estimated by expert and later updated as the data become available.

Figure 2. Self-Propelled Forage Harvester.

The model development process can be improved in terms of efficiency and accuracy by use of software tools. The main tools include editors for model entry, learning tools for model creation or updating from data and model evaluation tools. We will discuss a model evaluation tool in detail in Section 3. The remaining tools are available from many sources, both commercial and academic. HRL has its own toolset including a tabular editor Gnosis (Przytula et al., 2006) and model evaluation tool (Przytula et al., 2003). We also use a graphical editor called GeNIe, which comes with a reasoning engine, called SMILE, which also contains learning algorithms. Both tools are available free of charge from the University of Pittsburgh (http://dsl.sis.pitt.edu/).

2.2 Example Domain: Header Height Controller

In this paper we evaluate our methodology and tools on an electronic control unit (ECU) from a self-propelled forage harvester (SPFH). The specific ECU presented in this paper is known as the *Header Height Controller (HHC)*. It is responsible for automatically adjusting the height and angle of the header in order to follow the contour of the ground while harvesting. An SPFH with a corn header is shown in Figure 2.

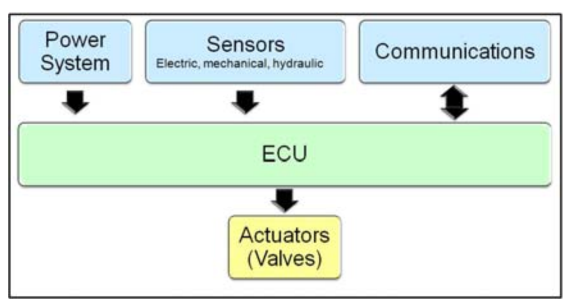

Figure 3. High-level block diagram of header height controller electronic control unit (ECU).

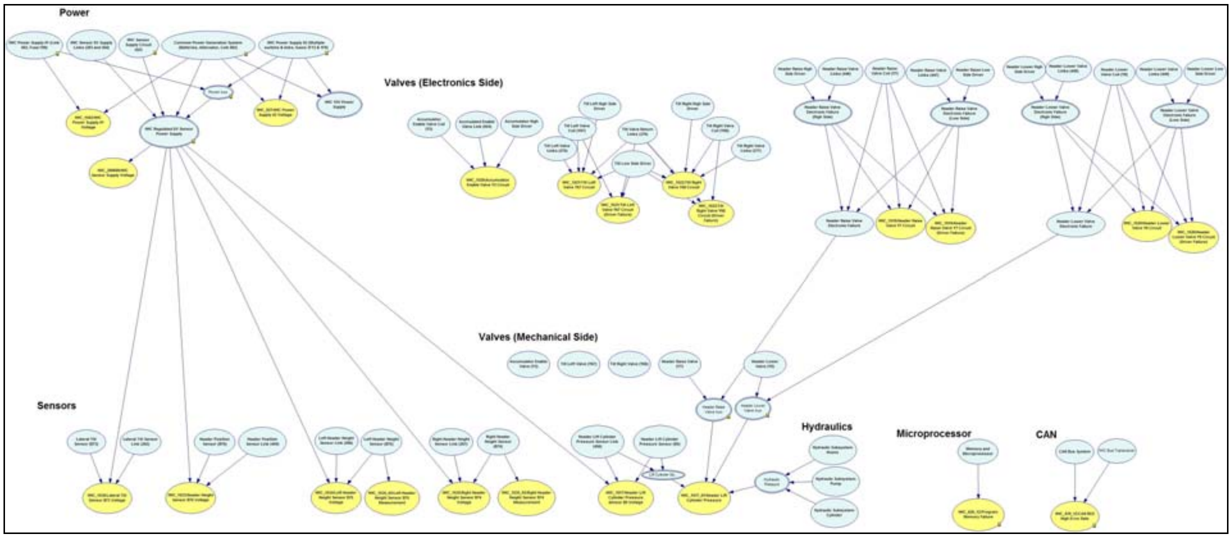

Figure 4. Structure of a diagnostic model of the ECU.

As shown in Figure 3, the ECU communicates with or receives inputs from the power system, external communications, and multiple types of sensors, including electric, mechanical, and hydraulic. It in turn controls multiple actuators affecting such functions as the header's height and tilt angle. The actuators are solenoid valves commonly used in hydraulic power systems.

In order to build a diagnostic model of the BIST for this system we began by identifying sources of domain knowledge such as: user and technical manuals, electronic schematics, hydraulic schematics, IO charts, diagnostic trouble code (DTC) lists, repair records, experts from the service department, and system designers and engineers.

Upon gaining an understanding of the domain we began identifying system components, their failure modes, and relevant observations. We grouped these into seven groups: power system, electric portion of valves and hydraulic portion of valves, sensors, hydraulic support system, microprocessor, and communications. During our modeling we identified 47 components (each with one or more failure mode) and 22 observations relevant to modeling the BISTs within the HHC. The structural relationships between these items are depicted in Figure 4, where components are shown as blue ovals and observations are shown as yellow ovals. The links between nodes in the figure depict the probabilistic (in)dependence assumptions of our model.

Our model also contains parameters in the form of prior likelihood of failures for components and sensitivities associated with the links. Component failures were determined by examining the manufacturing specifications and repair records in the service database. The sensitivities associated from links are derived from the observability of a test; its false positive and false negative rates.

Once we built a model of the system we analyzed the model to determine the extent to which the automated BIST observations could detect and disambiguate the different failures of the system (this will be further discussed in the next section). We then identified additional potential tests which could be included in the system, for an additional cost, to improve diagnosability. They were identified by looking both at where diagnosability was lacking in the current system as well as by looking at domain knowledge and what other observations would be feasible to include.

We identified three BIST additions. The first was a new BIST, which used existing sensors. Implementation would involve writing a new software routine. We labeled it Timeout DTC. The second addition improved existing BISTs by extending the set of failure mode indicators, they are called FMI DTC. The third was the most extensive and included a new sensor and new software. We call it Sensor DTC. We built and analyzed five different models: HHC (system with the original set of BISTs), "HHC+Timeout DTCs", "HHC+FMI DTCs", "HHC+Sensor DTC", and "HHC+All" (includes all the above additions).

3. MODEL BASED DIAGNOSABILITY ANALYSIS

3.1 Introduction to Diagnosability Analysis

System diagnosability analysis provides information about quality of diagnosis for a given system with defined diagnostic observations. In a conventional approach to diagnosability, system observations are assumed to be perfect, i.e. they always detect observed failure modes and never produce false alarms. Under these assumptions, only the following three cases are possible:

- component failure is perfectly detected and attributed to the root cause, e.g. there is a separate observation available for the failure mode or multiple observations uniquely point to it
- component failure is undetectable, i.e. no observation is available for the failure mode
- component failure is ambiguous, i.e. the observations point to more than one potential failure

For perfect observations, the system diagnosability results are reduced to listing the percent of all component failures uniquely diagnosable, percent diagnosable with ambiguity group of two failures, three failures, etc.

Our probabilistic approach allows for imperfect observations. The diagnostic results are expressed as probability of failure, i.e. they are real values from the interval [0,1]. The system diagnosability analysis is much more complicated, but the results are more informative. We perform the analysis using BN diagnostic models with the help of model evaluation software tools.

BN models effectively represent probabilistic relations between system components and observations as a joint probability distribution. Given the distribution we can compute the posterior probability of any variable given observations of any other variable.

Diagnosability computations are implemented in two phases: phase I – forward propagation from failures to observations and phase II – backward propagation or diagnosis of failure from observations. In phase I we assume that a particular failure mode is present and all the other components are OK. Given the states of the components, we compute posterior probability of outcomes for all observations. These probabilities represent how likely the outcomes are given the failure. In phase II we assume in turn that the outcomes of observations follow the distributions obtained in phase I and compute the diagnosis i.e. the likelihood of component failures given the observation outcomes. Phases I and II are repeated for all the failure modes – one at a time. Also, each execution of phase II involves multiple setting of observation states and diagnosis. The states of observation are sampled from the distribution obtained for them in phase I. For details on the algorithm see (Przytula et al., 2006).

The diagnosability result for a given system is a square matrix. It has a row and a column for each of the failure modes. The values of the matrix are probabilities of failures. Each row corresponds to an "actual" failure, whereas columns are the "diagnosed" or "predicted" failures. The values of the matrix on the diagonal are the diagnosed probabilities of actual failure, i.e. probability of detecting a failure. A system with perfectly detectable failures has all 1's on the diagonal of its matrix. The matrix values off the diagonal represent incorrect diagnoses. A system with no ambiguity in diagnosis has all 0's off the diagonal. Thus, a perfectly diagnosable system has the matrix with all 1's on the diagonal and 0's off the diagonal.

The diagnosability results can be visualized in 2D and 3D. It is convenient to sort rows of the matrix, so that the probabilities appear on the diagonal arranged from the largest to the smallest. In the 3D representation the values of probabilities are depicted over the grid of the failure modes. This representation gives a good overview of the diagnostic properties of the system, see Figure 5. The 2D representation, Figure 6, shows the diagonal values of the matrix (in blue) and the largest incorrect diagnosis (in red).

In our evaluation of a specific BIST we produce diagnosability results for BN models with and without the additional BIST. The changes in diagnosability results for the BN model representing the system with BIST relative to the system without it are used to compute the diagnostic benefit of the BIST. The details of this computation are presented in Section 4.

3.2 Diagnosability Analysis for Header Height Controller

In this section we present the diagnosability results for the HHC BISTs. Note that the analysis is limited to BIST only and does not include manual tests and inspections, which are available to service technicians. We present a 3D results in Figure 5 and a 2D projection of them in Figure 6.

In Figure 5 the axis along the lower portion of the screen contains the predicted failures and the axis along the right portion contains the true failures. Therefore, values along the diagonal show correct diagnosis and off diagonal results are incorrect diagnosis (values are sorted from highest detection to lowest). Flat areas show ambiguous failures (such as those with limited observability).

Figure 6 is a projection of Figure 5 down to 2D, where the blue values represent those along the diagonal (correct diagnosis) and the red values represent the largest off-diagonal value. Therefore, the desirable situation is for a large blue value and a small red value (this means good detectability and low ambiguity). When they are close together (e.g. near the middle of the figure) it means there is ambiguity and when the red values are larger than

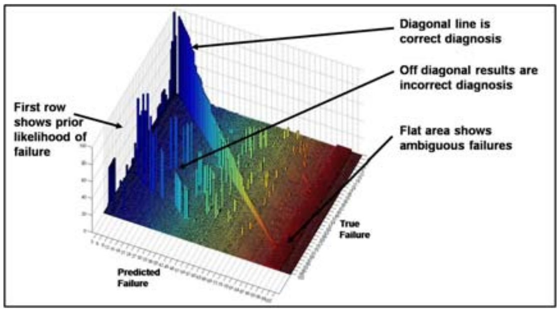

Figure 5. 3D visualization of model analysis showing the detectability and ambiguity for the model.

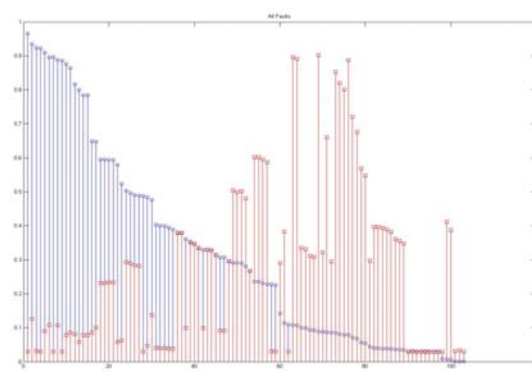

Figure 6. 2D visualization of model analysis showing the correct failure diagnosis with BIST (blue) compared with the highest incorrect failure (red).

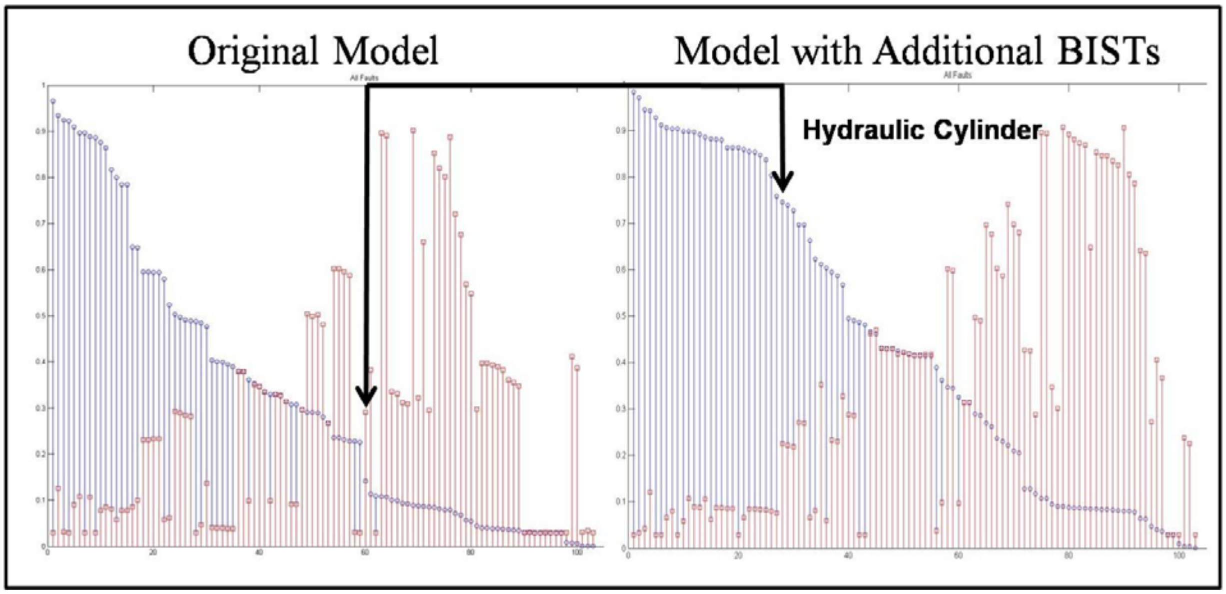

Figure 7. Analysis of the original model (left) and with all the additional BISTs (right). The marked failure is misdiagnosed in the original model, however with the additional BIST the correct diagnosis is made with almost no ambiguity in the new model.

the blue it means there is a misdiagnosis. If both are low, it implies that it is not only ambiguous, but that there is low detectability (e.g. near the right of the figure).

These values only represent the diagnosability based on the built-in self-tests; by adding in manual tests, symptoms, and inspections (e.g. examining the wires or hoses or plugging in a volt meter) we could further improve the detectability and disambiguation to help a service technician, however for this work we will continue to focus on built-in self-test observations.

In Figure 7 we show the analysis for both the original model and the model with all the new BISTs. You can see that in the original model there are 24 failures with a detection rate of over 0.5; in the new model this increases to 39 failures. Furthermore, the failure depicted for the hydraulic cylinder in both shows a failure which was misdiagnosed in the original model (a valve was blamed instead of the cylinder) and then in the new model it is correctly diagnosed with almost no ambiguity.

4. COST-BENEFIT ANALYSIS

The main goal of the cost-benefit analysis is evaluating the economic effect brought about by inclusion of additional BISTs. In general, inclusion of a new BIST results in the following costs: (1) *nonrecurring software cost* for BIST algorithm development, BIST parameter definition, coding, testing and verification, (2) *nonrecurring hardware cost* for the development of specification and the design of new hardware, (3) *recurring costs* for part manufacturing, assembly, and testing. On the other hand, it is expected that the new BISTs bring the benefit of reduced cost of warranty repairs (or system lifetime maintenance), as well as other benefits, such as shorter time in testing/verification, shorter time to market, design and test reuse, and customer satisfaction.

4.1 Graphical Cost-Benefit Analysis Model

Our approach to cost-benefit analysis is based on structural equation models. The model takes the form of a directed graph describing dependency relations among variables of the problem domain (Lu et al., 2000). In our cost-benefit model nodes represent variables, which can be input parameters or can be associated with a deterministic (or probabilistic) equation. The arcs represent dependency relations among variables. For any given node, its value is directly dependent on its parent nodes.

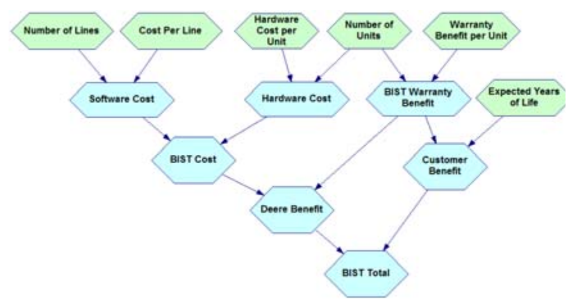

Figure 8. A Graphic Model for BIST Cost-Benefit Analysis.

Figure 8 shows a simplified graphical model for BIST cost-benefit analysis. Green nodes represent input parameters, whereas blue nodes are associated with equations. The equation computes the value of the node using the values from its parent nodes. We can obtain from the graph the direct dependency between nodes by considering a node and its parents (e.g., "BIST Warranty Benefit" directly depends on "Number of Units" and "Warranty Benefit per Unit"), the indirect dependency relation by inspecting the path between nodes (e.g., "BIST Cost" indirectly depends on the "Number of Units")

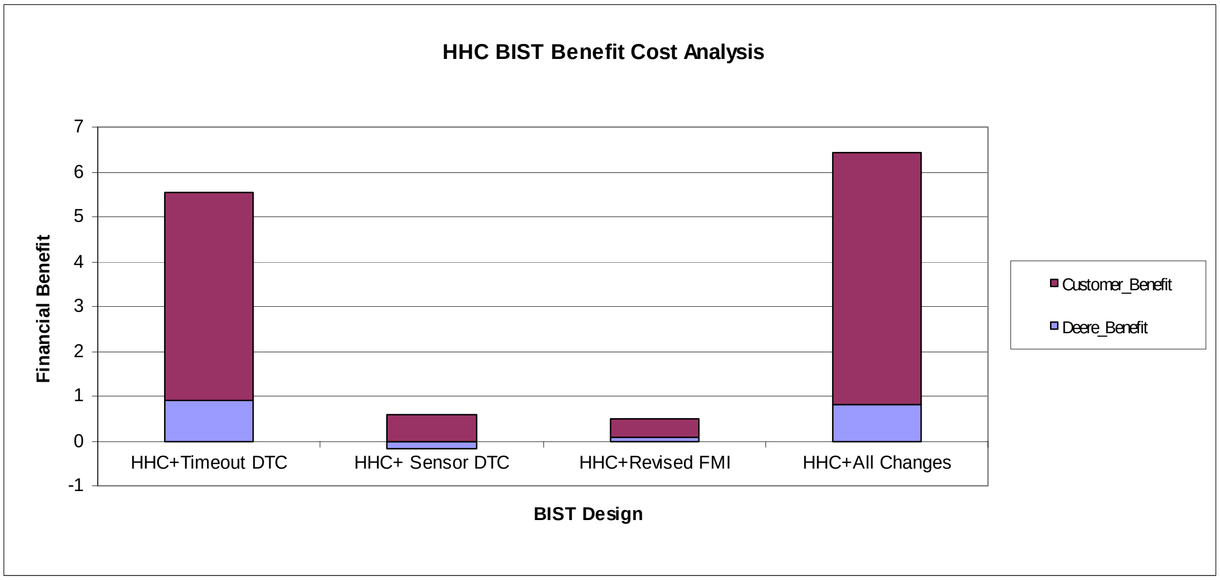

Figure 9. Cost Benefit Result for HHC BIST Addition.

as well as as common cause (e.g., "Number of Units" is the common cause of "Hardware Cost" and "BIST Warranty Benefit"). In other words the graphical model enables us to qualitatively examine our cost-benefit analysis model before defining any numerical values.

The specific equations for the model in Figure 8 are listed in Table 1.

Endogenous Variables	Equation
Software_Cost	CostPerLine*Number_of_Lines
Hardware_Cost	Number_of_Units*Hardware_Cost_per_Unit
BIST_Warranty_Benefit	Number_of_Units*Warranty_Benefit_per_Unit
BIST_Cost	Hardware_Cost+Software_Cost
Customer_Benefit	BIST_Warranty_Benefit*Expected_Years_of_Life
HRL_Benefit	BIST_Warranty_Benefit-BIST_Cost
BIST_Total	HRL_Benefit+Customer_Benefit

Table 1. Equations for BIST Cost-Benefit Analysis Model in Figure 9.

4.2 Expected Warranty Benefit

The graphical cost-benefit model in Figure 8 uses as input the Warranty Benefit Per Unit, which is expressed in dollars. This value is derived for a given BIST from the diagnosability matrix for the original system and the matrix for the system with the BIST added to it, see Section 3. In order to compute the dollar value of the benefit from the diagnosability benefit expressed in the matrices we need repair costs for all the components of the system.

We assume that the repair costs include the recurring cost, such as labor and part cost, as well as nonrecurring cost for testing/repair equipment. We make also several simplifying assumptions pertaining to the repair costs and procedures:

1. Cost/Benefit calculation is limited to the faults captured in the diagnostic model for the system, and only use BIST observations.
2. Repair is guaranteed to fix a fault, if the repaired component is correctly diagnosed.
3. Repair of misdiagnosed fault, is followed by a make-up sessions of diagnosis and repair, which continue until the actual fault is found and repaired.

The diagnosability matrices obtained in Section 3 represent the quality of diagnosis in terms of probabilities. The Warranty Benefit in dollars is actually an expected value of the financial benefit. The detailed discussion of the computations and the issues involved in it will be presented in an upcoming publication.

4.3 Analysis Results for Header Height Control System

In Section 3, we proposed three additional sets of BISTs for inclusion in the system. Therefore, we can look at each of these additions independently, or look at all of them together. The diagnosability analysis results and costs for repairs, estimated from historical repair data by our Deere subject matter experts, are taken as input for computing the expected warranty cost for each unit. The expected warranty benefit is then computed as the difference between the warranty cost for the new BIST design and the one for the existing BIST design. We then take this value as the input value for "Warranty Benefit per Unit" in the cost-benefit analysis model (Figure 8). We also elicit the values for "Number of Lines," "Hardware Cost per Unit," "Number of Units," and "Expected Years of Life" from our Deere experts for the four versions. We then perform the inference on the graphical cost-benefit model to derive the target values for "Deere Benefit, "Customer Benefit," and "BIST Total" (see Figure 9). It should be noted that it is possible for the benefit to be negative, rather than positive, as the costs can outweigh the benefits. For example, in the case of the BIST involving new sensor, i.e. HHC +

Sensor DTC, the manufacturer had a loss during the warranty year, because the cost of additional hardware and software was greater than warranty repair benefit. However, for the entire lifetime of the vehicle the BIST does provide the customer with a financial benefit.

5. CONCLUSIONS

Complex systems based on ECUs are difficult to diagnose. As a result, technicians often replace the ECUs to avoid time-consuming troubleshooting. This leads to a very high level of no-defect found and high warranty costs.

Our methodology for troubleshooting ECU subsystems begins with a review of DTCs produced by BISTs. The DTCs, sometimes augmented with symptoms of failure, determine which manual tests need to be performed to find the root cause of failure. Manual tests are expensive, so it is hard to expect that technician would perform more than two or three tests.

All this suggests that proper design of BISTs is critical for life-time cost of system health management. The BIST design has to include a careful cost-benefit analysis.

We have presented a methodology for accurate, quantitative BIST cost-benefit analysis. The methodology uses two forms of graphical models: a BN diagnostic model and a structured equation model. The development and verification of these models is well supported by software tools. The tools make it possible to apply the methodology to complex real-life systems with high efficiency and reliability.

We have illustrated the methodology and the tools on an example of a real-world ECU-based system. The models were created using data and expert knowledge and we applied the algorithms and computed numerical values of BIST cost-benefit.

In the future we intend to expand the analysis of BISTs beyond the tradeoff between cost of BIST and diagnosability benefits. In particular we intend to include additional BIST evaluation criteria such as energy efficiency.

ACKNOWLEDGMENT

We would like to thank John Deere for supporting this research. In particular, we are indebted Finbarr Collins who was kind enough to select and make available to us the repair frequency and cost statistics.

REFERENCES

K. Feldman, P. Sandborn, and T. Jazouli (2008). The Analysis of Return on Investment for PHM Applied to Electronic Systems, *in Proceedings of IEEE International Conference on Prognostics and Health Management (PHM)*, Denver, CO.

J.M. Lu and C.W. Wu (2000). Cost and Benefit Model for Logic and Memory Test, *in Proceedings of Design, Automation and Test in Europe*, Paris, France.

T.C. Lu, M.J. Druzdzel, and T.Y. Leong (2000). Causal Mechanism-based Model Construction, *in Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)*, San Francisco, CA.

J. Pearl (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., San Mateo, California.

K.W. Przytula, D. Dash, and D. Thompson (2003). Evaluation of Bayesian Networks under Diagnostics, *in Proceedings of IEEE Aerospace Conference*, Big Sky, MO.

K.W. Przytula and S. Smith (2004). Diagnostic Assistant Based on Graphical Probabilistic Models, *in 2004 SAE World Congress*, Detroit, MI.

K.W. Przytula, G.B. Isdale, and T.C. Lu (2006). Collaborative Development of Large Bayesian Networks, *in Proceedings of IEEE AUTOTESTCON*, Anaheim, CA.

L.Y. Ungar and T. Ambler (2001). Economics of Built-in Self-Test, *IEEE Design and Test*, 15(5), pp. 70-79.