Real-time Metareasoning with Dynamic Trade-off Evaluation

U. M. Schwuttke
Jet Propulsion Laboratory
California Institute of Technology
MS T1704 4800 Oak Grove Drive
Pasadena, CA 91109
ums@nereid.jpl.nasa.gov

L. Gasser
Computational Organization Design Lab
Institute for Safety & Systems Management
University of Southern California
Los Angeles, CA 90089-0021
gasser@usc.edu

Abstract

This paper describes dynamic trade-off evaluation (DTE), a new technique that has been developed to improve the performance of real-time problem solving systems. The DTE technique is most suitable for automation environments in which the requirement for meeting time constraints is of equal importance to that of providing optimally intelligent solutions. In such environments, the demands of high input data volumes and short response times can rapidly overwhelm traditional AI systems. DTE is based on the recognition that in time-constrained environments, compromises to optimal problem solving (in favor of timeliness) must often be made in the form of trade-offs. Towards this end, DTE combines knowledge-based techniques with decision theory to 1) dynamically modify system behavior and 2) adapt the decision criteria that determine how such modifications are made. The performance of DTE has been evaluated in the context of several types of real-time trade-offs in spacecraft monitoring problems. One such application has demonstrated that DTE can be used to dynamically vary the data that is monitored, making it possible to detect and correctly analyze all anomalous data by examining only a subset of the total input data. In carefully structured experimental evaluations that use real spacecraft data and real decision making, DTE provides the ability to handle a three-fold increase in input data (in real-time) without loss of performance.

Introduction

Most AI systems have addressed self-contained applications in which relatively unlimited time was available for producing solutions. Many real problems (such as monitoring) have high and dynamic data rates; in such applications the processing capability of AI systems may easily be exceeded by the processing requirements of the problem. In situations where these requirements change dynamically, AI systems must be able to rationally adjust their decision-making parameters to track changing problem requirements. To do this, new methods are being sought to provide accurate responses in the presence of time constraints and conflicting objectives. Many such methods recognize the need to make implicit trade-offs and compromises that favor timeliness over optimality. However, in some ways, these approaches are still not ideally suited for complex real-time environments. For example, some approaches require a metareasoning step before each action in the domain-level reasoning process [Russell 1989], where it may be more optimal to selectively invoke metareasoning, particularly when the metareasoning is not time-constrained. Other approaches such as those that use incremental (or "anytime") algorithms [Dean 1990; Horvitz 1989] analyze the expected value of a computation prior to performing it. In severely time-constrained situations, it may be better to vary the solution strategy according to the dynamics of the environment than to rationally terminate or continue a single fixed strategy. Thus, additional methods are needed for complex, highly dynamic applications. While no techniques will handle all situations without overload, newer methods can make it possible to obtain increased performance from limited resources in critical moments.

Toward this end, we introduce Dynamic Trade-off Evaluation (DTE), a new approach to real-time metareasoning that combines decision theory and knowledge-based techniques to automatically determine both *when* trade-offs become necessary and *how* to implement them with minimal impact on solution quality. DTE offers a general methodology for explicitly making a variety of trade-offs. It provides the ability to perform metareasoning only when necessary by dynamically modifying the solution strategy based on both the dynamics of the environment and the changing goals of the monitored system. We have evaluated the performance of DTE in real spacecraft-monitoring problems with real data. In carefully structured experimental evaluations the DTE technique provides the ability to handle a three-fold increase in input data (in real-time) without performance loss. DTE is applicable to a wide variety of run-time trade-offs and can be integrated into a real-time monitoring architecture.

Metareasoning with DTE

The applicability of decision theory and the psychology of judgement to metareasoning was recognized early, with research on heuristic methods for controlling inference [Simon 1955]. However, initial enthusiasm for using decision

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronatutics and Space Administration.

theory as an AI technique dwindled in favor of other approaches that more easily expressed the rich structure of human knowledge [Horvitz 1988]. Recently, there has been renewed interest in decision theory for real-time AI applications.

A variety of techniques exist in multi-attribute utility theory for evaluating competing objectives. However, only three variants of these have been commonly applied to real-world situations [von Winterfeldt 1986]: the simple, multi-attribute rating technique [Edwards 1977], difference value measurement [Dyer 1979], and subjectively expected utility (SEU) measurement [Keeney 1976]. Of these three techniques, the Edwards technique is the simplest computationally, because it uses additive (rather than multiplicative) utility and aggregation models, it relies on direct rating and ratio estimation (rather than probability methods) for determining utilities and weighting factors, and it involves only the calculation of a simple dot-product for each alternative under evaluation. Moreover, for many practical applications, the results of the simpler technique are theoretically and behaviorally comparable with the other methods [von Winterfeldt 1986].

Utility analysis methods studied in multi-attribute utility theory have generally been applied to one-time decision making in situations such as selecting real estate sites [Edwards 1982] or evaluating coastal development proposals [Gardiner 1974]. The implicit assumption under these methods is that the criteria and objectives which formed the basis of the evaluation process will remain valid after a decision has been made; once selected, a real estate site or a coastal development proposal should remain appropriate for a suitably long period. In this type of decision, it is appropriate that the decision criteria, weighting factors, and final evaluation be static. However, a single decision under static criteria will not appropriately reflect changing circumstances in the environment of a continuous real-time problem solving system, where trade-offs must be made dynamically and continually.

In contrast, DTE enables dynamic evaluation of real-time trade-offs and maintains the advantages of simplicity, robustness, and flexibility associated with static methods. In DTE, utility analysis is used to rank alternatives in a preference space. Domain knowledge provides decision rules that are used at run time to 1) dynamically reweight decision criteria and 2) dynamically select among alternatives in a preference space (based on situational attributes and operational modes). Here we describe the DTE procedure in general. In the next section, we illustrate the specifics of DTE by showing how we have applied it in a case of trade-off analysis in spacecraft monitoring. The DTE procedure consists of six steps, some of which are dynamic parallels of steps in a static utility analysis procedure [Edwards 1977]. The first three of these steps and part of the fourth must be completed during the design phase of the system. The other steps are automated, real-time control activities. The DTE procedure includes:

1. *Definition of the trade-off instantiation mechanism.* This step involves specifying when DTE is required and designing the mechanism that invokes the trade-off evaluation when appropriate.

2. *Definition of application-specific alternatives and criteria that determine their values.* The alternative actions to be considered in trade-off evaluation are specified, along with criteria that will be used to evaluate the alternatives. As part of this process, the system designers and domain experts also specify domain knowledge that defines the various ways of implementing each alternative.

3. *Separate evaluation of each alternative.* This is done in conjunction with the previous step, and involves reliance on subjective judgements in cases where no basis for objective evaluation exists. Each alternative is ranked with respect to each of the evaluation criteria, i.e. on a scale of 0 to 100, and suitable consistency checks are applied to the evaluation.

4. *Definition of weights and modes.* Relative weights are assigned to each of the criteria, along with ranges within which the weights can vary. Domain knowledge is specified to determine the circumstances under which the weights will be varied. In addition, multiple modes may be specified, where each mode is governed by a different set of weights. At run-time, both the variation of the weights and the choice of a mode are automatically determined, as needed.

5. *Aggregation.* The weights from the previous step are used to determine the aggregate value of each of the alternatives, using an additive aggregation model.

6. *Selection.* The alternative with the maximum utility is selected and implemented. When the evaluation indicates that two or more alternatives are equally good, domain knowledge is used to select one alternative over the others, or, if the alternatives are not mutually exclusive, to perhaps select several of them.

For a specific illustration of how to apply the DTE procedure, we address a representativeness vs. timeliness trade-off that occurs in managing data for the NASA Galileo mission's Solid State Imaging subsystem. We have also studied a second trade-off of problem solving strategy: focus (on a specific problem solving task) versus responsiveness (to other unforeseen but possibly more important tasks) in Voyager mission system-level analysis problems. The details of this second trade-off are addressed elsewhere [Schwuttke 1991].

The Solid State Imaging (SSI) subsystem on newer missions has a much faster image frame rate than the technology used on previously. Readout rates can be as fast as one image every two seconds (compared to one every 96 seconds from Voyager). Forty-eight channels of camera status data are associated with each image. This includes dynamic data pertaining to exposure time, filter position, gain state, readout mode, and data compression mode, etc. There are also non-dynamic data channels that indicate general instrument status, voltages and currents. While non-dynamic channels could be managed with an extension of existing data management techniques [Washington 1989], the dynamic data does not follow trends and re-

quires a new approach. This is because the "correctness" of a data value is independent of the correctness of previous values: a value that was correct at one moment can, without changing, become incorrect at the next moment, depending on subsystem goals. For example, exposure times vary with spacecraft goals. When goals change, new exposure times may be required; if data related to these parameters does not change, the goals will not be achieved. As a result, intelligent management of more complex data requires the application of knowledge-based techniques that reflect the dynamic goals of the monitored system. The large amount of data, the occasional dependence on heuristics, and the complexity of tasks make this an ideal problem for demonstrating the benefits of DTE.

The basic real-time mission operations task involves comparison of telemetry to predicted data values or accepted limit ranges. The predictions reflect expected performance based on known command sequences and the limit ranges reflect the general operating parameters of the instrument. The task involves two AI components: intelligent data management and anomaly analysis, as shown in Figure 1; the latter capability has been addressed in the MARVEL system [Schwuttke 1991b] and will not be addressed here. The (competing) data management goals in this application include adjusting input data volumes to meet processing capabilities, maximizing the content of the input information, maintaining alertness to unusual events in the input data, and focusing on particularly relevant tasks.

The first step of DTE involves defining an instantiation mechanism. For this application, a software module that analyzes the size of the input backlog invokes trade-off evaluation.

In the second step, the four possible data management alternatives (provided by an imaging subsystem specialist) include: 1.1) eliminating channels not in a basic monitoring set, 1.2) eliminating channels not in a heuristically defined minimal monitoring set, 2.1) reducing sampling rate on heuristically defined subset of channels, and 2.2) reducing sampling rate on the entire channel set. The four alternatives are evaluated on criteria that define representativeness, or information content. For data reduction, these include: (A) non-dynamic behavior, (B) irrelevance to an existing problem area, and (C) non-negative impact on monitoring integrity. A data channel must be non-dynamic to be eliminated; frequent value changes indicate a high level of activity that must be monitored to maintain representativeness. Irrelevance to existing problem areas is also important in deciding which channels to remove from the monitored set. Finally, only channels that do not compromise situational monitoring integrity can be eliminated without impacting representativeness.

The second step also requires the specification of rules that show how to implement the alternatives. The channel elimination alternatives and the second sampling rate alternative are influenced most heavily by a decision tree that defines which channel subsets may be deleted from the monitored set and when they may be deleted. In contrast, the heuristically-defined sampling rate alternative is entirely governed by the situation in which it is applied. In a normal operating mode, the sampling rate can be reduced on all channels that are not part of the critical subset. In an anomaly detection mode, sampling can only be reduced on channels that are irrelevant to anomaly detection.

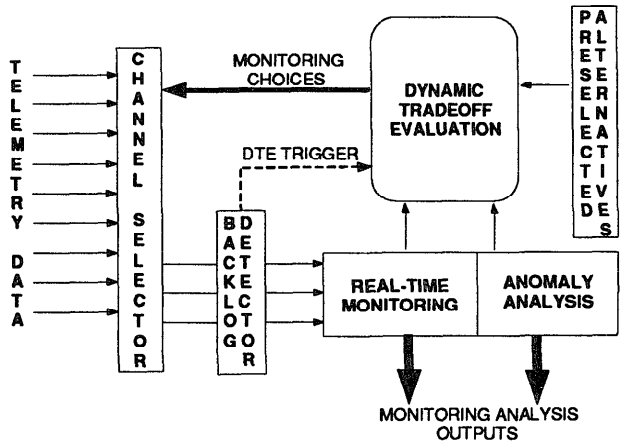

Figure 1. A DTE architecture for intelligent data management in a monitoring system.

However, in the event of extreme backlogs, sampling on all channels may be reduced. In such situations it is important to note that if the minimal subset is not preserved, some loss of representativeness may result; domain knowledge must be used to make timeliness vs. representativeness trade-off in these cases. Occasionally channels must be added irrespective of timeliness. This is because in anomaly detection mode, increased representativeness takes instant precedence, and channels pertinent to that anomaly must be added. When the system returns to a normal operating mode, the channels relevant to a previously resolved anomaly may be candidates for removal from the monitoring set if timeliness must be improved.

The third step calls for subjectively ranking each alternative in the context of each criterion at design time, as shown in Figure 2. The ranking, obtained with the help of the subsystem expert, is on a scale of 0 to 100, with 100 having the maximum value, then checked for consistency. This step also involves assigning relative weights to the criteria. Initial weights and variance ranges for these weights are defined for adjusting the weights during the

	ALTERNATIVE NUMBER			
ATTRIBUTE	1.1	1.2	2.1	2.2
A	75	90	30	40
B	20	30	90	50
C	100	75	40	25

Figure 2. Alternative values for the SSI example.

reasoning process. Weight variations are initiated when the system detects that its performance is degrading, and are implemented using rules that provide updates based on situational parameters.

Two sets of weights are defined for this application, as shown in Figure 3. The first set applies in the normal operating mode and the second applies in an anomaly detection mode. In the normal operating mode, the irrelevance of a channel to an existing problem area is given no weight, (no problems exist in this mode). However, in anomaly analysis mode, this attribute receives the most weight.

In the fourth step, the single-attribute rankings and weights are aggregated into an overall evaluation of alternatives which, with the application-specific domain knowledge, enables the selection of the best alternative for the given circumstances. This step differs significantly from the comparable static step for two reasons. First, circumstances dictate varying weights, which in turn dictate varying aggregations. Secondly, circumstances may require varying the knowledge that is applied from situation to situation. Examples of the varying aggregations that are obtained for both operating modes are shown in the tables of Figure 4. These tables show that the data management actions that are most compatible with maintaining maximum representativeness are determined by external circumstances. The rankings of the alternatives with regard to representativeness is summarized in Figure 5, with the value of 1 being the highest ranking.

The final step involves the selection of an alternative and is based on dynamic evaluation of the representativeness vs. timeliness trade-off. In order to make this trade-off, the four alternatives must also be evaluated with regard to timeliness. The timeliness impact of an alternative is directly proportional to the percentage reduction (or increase) in the number of monitored channels that results from implementing that alternative. However, this percentage must be calculated immediately prior to making the trade-off, based on the channels in the current monitoring set, because the number of monitored channels is a dynamic quantity determined by the events leading up to the current circumstances. The following example shows the dynamic and adaptive nature of this evaluation.

Assume that the monitoring system has just been activated. Initially, 49 data channels are in the monitored set. After some time, the system detects a growing input backlog, and responds by deciding that some channels must be removed from the monitored set. No anomalies

A	NON-DYNAMIC BEHAVIOR	N.O.M. (.45+/- 0.2) A.D.M. (.15 +/-0.1)
B	IRRELEVANCE TO AN EXISTING PROBLEM AREA	N.O.M. (0.0) A.D.M. (.60)
C	NON-NEGATIVE IMPACT ON MONITORING INTEGRITY	N.O.M. (.15 +/- 0.2) A.D.M. (.25 +/- 0.1)

(NOM- Normal Operation Mode / ADM- Anomaly Detection Mode)

Figure 3. Criteria and weights for the SSI example.

have been detected as yet, and no modifications to the starting weights have been suggested by the knowledge base. As a result, the system uses the aggregate values in the first line of Figure 3 as representativeness values for the four alternatives. Corresponding timeliness values are obtained by calculating the net percentage reduction in input data that would be achieved with each alternative. Actual timeliness values are plotted against the aggregate representativeness value as shown in Figure 6 (left). Both representativeness and timeliness are rated on a scale of 0-100; 1 unit on the representativeness scale is equivalent to 1 unit on the timeliness scale. The indifference curves shown in the figure are created by this constant trade-off of units; alternatives lying on the same indifference curve have equivalent value, and alternatives lying nearest to the upper right of the graph are perceived as best. For this application, the alternatives in order of preference are 1.1, 1.2, 2.2 and 2.1. (This is not the same order of preference as in Figure 4, which was based on representativeness alone.) As a result of this analysis, alternative 1.1 is implemented. Our system is now monitoring 17 of the 49 data channels, and is achieving adequate throughput.

Later, an anomaly appears on one of the channels, which requires three additional channels to be added for anomaly analysis. The anomaly is solved, and at some later time, an anomaly appears on a second channel; this anomaly requires the addition of 12 more channels. We are now actively monitoring 32 channels, and are building a backlog. This causes the backlog detection module to initiate metareasoning. Figure 6 (right) shows the re-evaluation at this point. Now, however, the selection of an alternative is not as obvious as in the previous cycle: alternatives 2.1 and 2.2 are very close to lying on the same indifference curve. However, heuristics indicate that in the current mode, representativeness is the more important consideration, and alternative 2.1 must be selected. Eventually, this anomaly is resolved, and we return to the normal operation mode.

Experimental Evaluation of DTE

We have successfully tested DTE in the dynamic evaluation of the representativeness vs. timeliness trade-off in the Galileo SSI subsystem. This section describes the evaluation procedures and presents the results obtained.

To test DTE, we assembled test files containing simulated data that contains actual anomalies (supplied by JPL's imaging-subsystem expert) in various anomaly densities (percentages of anomalous data in the file). The densities were chosen to provide evaluation of the DTE methods across the complete range of possible anomaly densities, in order to understand for which applications and situations DTE would be most successful. The test files were used to compare three different input management approaches. These included random data elimination, incremental filtering, and intelligent data management using dynamic trade-off evaluation. The first two methods provide a way to compare DTE to conventional approaches

Attribute	Weight*	Weight**	Weight***	ALTERNATIVE NUMBER 1.1	1.2	2.1	2.2
A	*0.45*	*0.65*	*0.25*	75	90	30	40
B	*0*	*0*	*0*	20	30	90	50
C	*0. 55*	*0.35*	*0.75*	100	75	40	25
Aggregate Value (using weight*)*				88.57	81.75	35.5	31.75
*Aggregate Value** (using weight**)*				83.75	84.75	33.5	34.75
*Aggregate Value*** (using weight***)*				93.75	78.75	37.5	28.75

Figure 4. Aggregate alternative values for varying weights in single anomaly mode.

MODE \ ALTERNATIVE	Elimination of chan. not in basic subset	Elimination of chan. not in critical subset	Sampling reduction on heuristic subset	Sampling reduction on entire subset
N.O.M. with no modification	1	2	3	4
N.O.M. with backlog modification	2	1	4	3
N.O.M. with monitoring modification	1	2	3	4
A.D.M. with no modification	3	2	1	4
A.D.M. with backlog modification	3	2	1	4
A.D.M. withmonitoring modification	2	3	1	4

Figure 5. Rankings of alternative values with respect to representativeness.

actually used in practice. Random data elimination may appear to be an unusual choice. However, it most closely parallels the methods that less-experienced human analysts use in data-overload situations: when their data backlog becomes too large, they skip over the data in the backlog, and focus on the newly arriving data. Incremental filtering involves less loss of information. Data is filtered according to $f = n/b$ (when $n<b$), where f is the fraction of samples passed through the filter, n is the total number of channel types, and b is the number of samples in the backlog. A backlog accrues according to the ratio of the incoming data rate to real-time processing rate (i.e., with a backlog accrual of x, data is arriving x times faster than it can be processed). Each of the three methods were evaluated with respect to performance under increasing backlog accrual. Finally, we used two criteria of performance: percentage of anomalies successfully detected and percentage of related data processed that is actually needed for correctly diagnosing anomalies.

The experimental results are summarized in Figure 7. The first of these graphs summarizes the average percentage of anomalies successfully detected under increasing input data volumes and increasing anomaly density. The second graph summarizes the high-low spread of anomaly-relevant information that is processed under increasing input data volumes and increasing anomaly density. The first of these evaluation criteria is important with respect to automated monitoring, and the second is relevant to automated anomaly analysis. Anomaly detection using DTE is highly successful for data rates as high as 2.5 times the real-time monitoring capability, particularly at anomaly densities of 10% or less. In these operating ranges, DTE outperforms both random data elimination and incremental filtering, detecting over 90% of all anomalies. The success of anomaly detection with random data elimination and incremental filtering, on the other hand, drops below 50% at backlog accrual rates as low as 1.5.

Processing of anomaly-relevant information involves passing channels relevant to the analysis of a detected anomaly from the data management module to the monitoring and analysis module. If anomaly-relevant channels are filtered by the data management module, some of the information needed for analysis will be lost. Intelligent data management with DTE is most successful at low (5% or lower) anomaly densities with backlog accrual rates that exceed real-time processing capabilities by as much as 2.5. Within these operational parameters, processing of anomaly-relevant information is as high as 95% for backlog accrual rates equal to twice the processing capability, 80% for backlog accrual rates equal to 2.5 times the processing capability, and 70% for backlog accrual rates equal to three times the processing capability. At these backlog

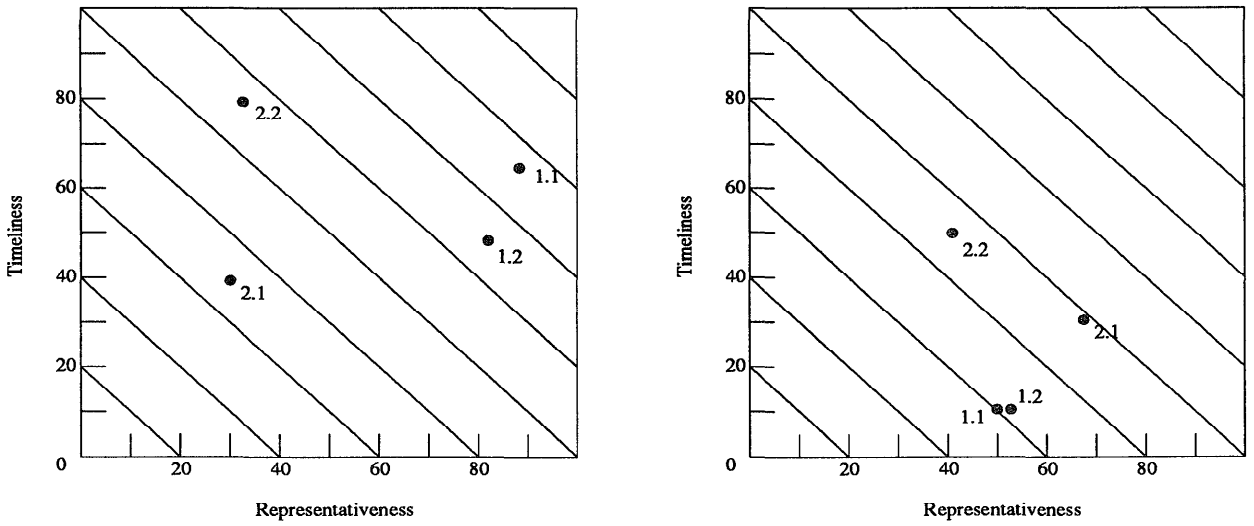

Figure 6. Timeliness vs. Represntativeness Values for Galileo SSI input data management scenarios.

accrual rates, the other two methods provide no more than 70%, 50%, and 50% of anomaly relevant data, respectively.

Discussion

Three criteria for intelligent data management systems have been identified [Washington 1989].

* The system should be responsive to changing resource requirements. For example, the amount of data sampled should vary with the computational load placed on the system.
* The system should be responsive to important and unusual events in the input data, even when it is "busy".
* The system should be able to focus its attention on parameters that are particularly relevant to the current reasoning task.

The Galileo SSI application has shown that DTE provides an effective way to achieve each of these criteria, not only for "thresholdable" data addressed by Washington, but also for goal-driven data that does not occur in his application. DTE enables both anomaly detection and anomaly diagnosis for low anomaly densities and moderate backlog accrual rates. Actual anomaly densities for this application average less than 3%, which is well within the acceptable operational parameters of the method.

Figure 7 show performance degradation in the DTE method beginning at backlog accrual rates that exceed real-time processing capability by a factor greater than three. Furthermore, when the backlog accrual rates exceed processing rates by a factor of four or more, the DTE method begins to converge with the other two methods. This performance degradation is governed by the domain knowledge. The minimal monitoring set for complete anomaly detection (as defined by the domain expert) consists of one third of the entire channel set. When these channels are not monitored, some loss of monitoring integrity will occur, as is demonstrated by degradation of the DTE method at backlog accrual rates of 3.5 or more. Subsequent testing shows that with an imaginary domain, in which the minimal set can be defined as a significantly smaller subset of the total channel subset, the effective increase in data reduction that can be achieved is on the same order as the decrease in size of the minimal set. A long-term solution to this problem in the context of a specific domain involves designing telemetry (or other input data) definition to later enable maximum data reduction. The more hierarchically the monitored data can be structured, the more the monitored data set can be reduced.

The observed performance degradation at known data rates enables the system to predict its own failure and provide warnings of reduced monitoring integrity. In a distributed environment, a module that predicts its own failure to meet real-time constraints could actually request additional processing resources from the environment.

Conclusions

Dynamic Trade-off Evaluation has been shown to be an effective technique that offers significant benefit to real-time AI systems. DTE incorporates a mix of knowledge-based and utility-theoretic techniques and is particularly valuable in real-time monitoring situations of moderate anomaly densities, varying data rates, and dynamic decision criteria. In experimental evaluations, DTE signifi-

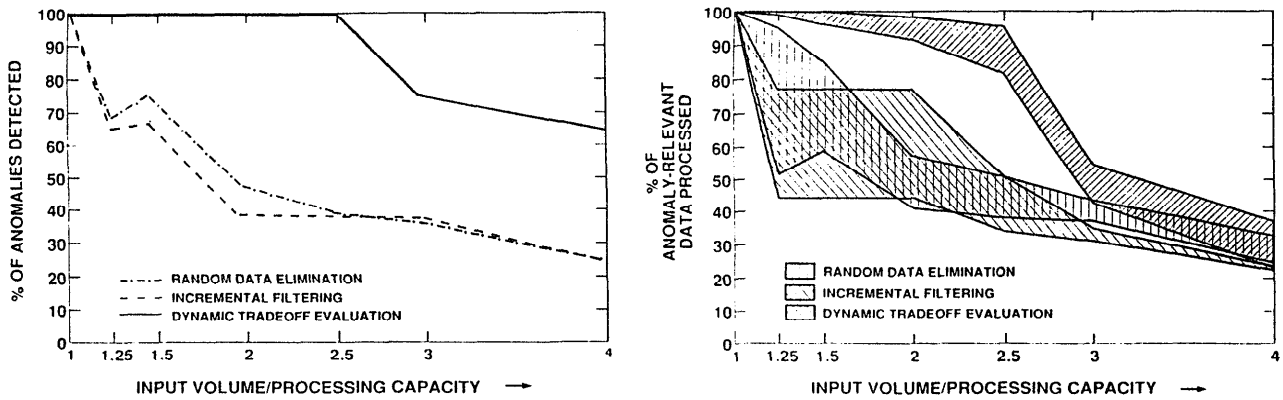

Figure 7. Results of experimental analysis for anomaly detection and anomaly analysis at 3% anomaly density.

cantly outperforms other commonly-used approaches to manage real-time monitoring data trade-offs in increasing-backlog situations. Moreover, DTE is a generic technique that can be effectively applied in many kinds of trade-off analysis for real-time systems. We have designed a generic architecture for DTE applications, treated elsewhere [Schwuttke 1991], and have taken initial steps to implement DTE as an operational part of the MARVEL intelligent monitoring system [Schwuttke 1992; Schwuttke 1992] in use at JPL.

Acknowledgments

The authors wish to acknowledge mission expertise from William Cunningham, software contributions from Alan Quan, and helpful discussions with John Rohr, Kenneth Goldberg, Bruce Abramson, and Alice Parker.

References

Dean, T. 1990. Decision Theoretic Control of Inference for Time-Critical Applications, Technical Report CS-90-44, Department of Computer Science, Brown University.

Dyer, J. S. and Sarin, R A. 1979. Measurable Multi-attribute Value Functions. *Operations Research*, 22: 810-822.

Edwards, W. 1977. How to Use Multi-attribute Utility Measurement for Social Decision Making, *IEEE Transactions on Systems, Man and Cybernetics*, SMC-7, 326-40.

Edwards, W. and Newman, J. R. 1982. *Multi-attribute Evaluation*. California: Sage.

Gardiner, P. 1974. The Application of Decision Technology to Multiple Objective Public Policy Decision Making. Unpublished Ph.D. dissertation, University of Southern California.

Horvitz, E. J.; Breese, J. S.; and Henrion, M. 1988. Decision Theory in Expert Systems and Artificial Intelligence. *International Journal of Approximate Reasoning*, Special Issue on Uncertain Reasoning.

Horvitz, E. J.; Cooper, G. F.; and Heckerman, D.E. 1989. Reflection and Action Under Scarce Resources: Theoretical Principles and Empirical Study. Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, 1121-1127.

Keeney, R. and Sicherman, A. 1976. An Interactive Computer Program for Assessing and Using Multi-attribute Utility Functions. *Behavioral Science*, 21: 173-182.

Russell, S. and Wefald, E. 1989. On Optimal Game-Tree Search Using Rational Metareasoning, Proc. of Eleventh Int. Joint Conf. on Artificial Intelligence, 334-340.

Schwuttke, U. M. 1991. Intelligent Real-time Monitoring of Complex Systems. Ph.D. dissertation, Dept. of Electrical Engineering, University of Southern California.

Schwuttke, U. M.; Quan, A. G.; and Gasser, L. 1991b. Improved Real-time Performance in Automated Monitoring Using Multiple Cooperating Expert Systems. Proc. of 4th International Conference on Industrial and Engineering Applications of AI and Expert Systems, 221-228.

Schwuttke, U. M.; Quan, A. G.; Angelino R.; et al. 1992. MARVEL: A Distributed Real-time Monitoring and Analysis Application. In *Innovative Applications of Artificial Intelligence 4*, eds. P. Klahr and C. Scott. Boston: MIT Press. Forthcoming.

Simon, H. 1955. A Behavioral Model of Rational Choice. *Quarterly Journal of Economics*, 69.

von Winterfeldt, D. and Edwards, W. 1986. *Decision Analysis and Behavioral Research*. Cambridge: University.

Washington, R. and Hayes-Roth, B. 1989. Input Data Management in Real-time AI Systems. Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, 250-255.