Failure mode analysis for a hazardous waste clean-up manipulator

Ian D. Walker & Joseph R. Cavallaro

Department of Electrical & Computer Engineering, Rice University, Houston, TX 77251, USA

This paper describes the application of Fault Tree Analysis to the design phase of a robot manipulator for hazardous waste retrieval. The robot is to be deployed in single-shell under-ground storage tanks at the US Department of Energy (DOE) site in Hanford, Washington. These tanks contain a variety of highly radioactive waste types, necessitating extremely safe and reliable manipulator operation. Based on preliminary design drawings of this long-reach manipulator, fault trees were constructed for several critical failure scenarios. Analysis of the trees revealed a number of ways to improve the safety and reliability of the manipulator design. This paper presents a summary of the fault tree analysis, with a discussion of the applicability of qualitative and quantitative fault tree methods to hazardous waste robotics. © 1996 Elsevier Science Limited

1 INTRODUCTION

Robotic arms will be deployed to assist in remediation of the waste tanks at the Hanford, Washington, Nuclear Site. At Hanford, about 140,000 m³ of high-level radioactive waste is being stored in 149 single-shell tanks (SSTs).²² This waste has been produced at Hanford since the 1940's as a by-product of processing spent nuclear fuel for uranium and plutonium recovery.

The US Department of Energy has established a program to remove, treat, and dispose of the wastes stored in these underground tanks. The goal is to 'develop and field retrieval systems with the capability to remove wastes from these tanks and transfer those wastes for further downstream processing'.²²

Removal of the waste represents a significant technological challenge. The SST wastes exist mainly in two solid forms: sludge (from thick soup to thick peanut butter type consistency) and saltcake (a hard abrasive substance similar to weak concrete in consistency).²² In addition, the situation is made more complex by the addition of in-tank hardware.¹ This hardware is comprised of both monitoring hardware and pumps (extending from vertical risers into the roof of the tanks) and also items (such as lead bricks, measuring tapes, rocks, etc.) which have been dropped into the tanks over the years. The waste volumes range from 5% to 95% of the tank's capacities.

One possible technology for removing the tank wastes is low-pressure, high-volume hydraulic sluicing. However, there are concerns that this approach may compromise the integrity of the tank linings, and that also it may not be able to effectively recover all waste types.²² This concern has led to the consideration of manipulator arm-based retrieval systems, which could manoeuvre high-pressure, low volume water jets into the tank for more efficient retrieval.

Clearly, safety and reliability will be critical factors in the deployment of such robot arm-based systems. This paper details the use of fault trees in the safety analysis of the Hanford Generation 1 Waste Retrieval Manipulator (WRM). The possibility of failure of the arm's hardware or software must be analyzed in detail. The prototype (Generation 1) WRM is being used to develop failure scenarios that can be used to specify parameters for vendors of commercial successor hardware and software.

2 ARM-BASED TANK WASTE RETRIEVAL PLAN

Several different options for remediation of the tanks have been considered.⁸ The most promising approach, that of a long reach manipulator, is being adopted. This strategy involves a mechanical arm mounted above the tank that reaches down into the tank to position retrieval tools.⁸ This approach appears to

offer the best chance of retrieving the waste while avoiding the many (partially known) obstacles¹ in the tanks due to its adaptability and extensibility. The concept is illustrated in Fig. 1.

Several options, including a relatively small Light Duty Utility Arm^{3,6,14} are currently being developed. All involve the manipulator being lowered through an access port in the roof via a vertical positioning mast. The manipulator must have the dexterity and reach to unfold and position an end effector (retrieval tool) throughout the interior of the tank, while avoiding vertical risers and other tank hardware obstacles. The manipulator must also be removable through the mast structure, even in the event of a failure.

In this paper, we focus on the design of the Hanford Generation 1 Waste Retrieval Manipulator (WRM).³³ This manipulator is designed for first deployment and testing in tank 241-C-106 at Hanford. An initial design study for the manipulator was given in Ref. 33. The study includes conceptual arm design figures, and outlines a possible control architecture. The control architecture is quite simple with a number of processing units connected in series. These units are, in turn, an Input Device Controller (operator interface), the Supervisory Computer (planning functions), a Subsystem Controller, individual Joint Controllers, to the low-level Actuator Drivers and Analog to Digital Converters, etc. This architecture

did not initially offer any fault tolerance, since a failure anywhere in the control system will propagate directly through to the robot.

The kinematics of the arm in Ref. 33 are designed for the special features of the task. The first joint motion is vertically down into the tank (the arm descends in a telescoping fashion). The second joint motion is a rotation about the vertical axis (to rotate the arm in the tank to point in all directions around the tank). The following joints are all in the same plane (perpendicular to the ground) which rotates with motion of the second joint. This allows the arm to 'unfold' in the tank.

The authors conducted an investigation, based on a fault tree analysis of the design in Ref. 33, to analyze the failure modes and reliability of the design. The following sections describe the results of this study. A full description of the work, which was performed as part of the DOE Robotics Technology Development Program (University Research Program) at Sandia National Laboratories¹² can be found in the report.³¹

3 FAULT TREE ANALYSIS

Fault Tree Analysis is a deductive method in which failure paths are identified using a fault tree drawing or graphical representation of the flow of fault events. Essentially, the sequence of events which could lead to a given critical manipulator failure scenario are identified and logically combined into a tree structure. Fault Tree Analysis is a well-known analysis technique often used in industry for computer control systems and large industrial plants. For a more detailed description of the fault tree technique, the reader is encouraged to consult Refs 2, 25, 35.

Fault Tree methods have been applied to robot manipulators only quite recently, and still quite sparingly.^{4,11,18,26,30} There has also been recent work in fault tree analysis for robots designed for nuclear environments.^{10,15,21} For a review of fault tolerance and reliability approaches in robotics, see Refs 7, 28.

A number of fault trees were constructed³¹ to aid in the reliability analysis of the Hanford Manipulator. The failure scenario of a collision with a vertical riser within the tank was examined in detail. In addition the failure scenario of the inability to remove the manipulator from the underground storage tank was extensively studied.

In the following, we present a set of simple example fault trees for the second of these critical failure scenarios of the WRM. The scenario presented is the failure to remove the arm from the tank. This is the top event in the overall failure tree. This event in turn can be caused by a number of factors, for example, collision of one or more links of the arm with the

Fig. 1. Tank waste retrieval manipulator concept.

deployment mast during arm retrieval. Much of our later discussion will centre on this scenario. Failure events which could lead to the collision and their interrelations are detailed in the trees.

A number of assumptions were made in producing the trees. The position of any risers are presumed to be reasonably accurately known. The operator is assumed to have access to a monitor with camera feed from inside the tank. The actuators are assumed to be hydraulic, driven by hydroelectric valves. The kinematic design of the robot is assumed to be that displayed in drawings SK-2-300048 (Waste Retrieval Manipulator with End Effector) and SK-2-300049 (Waste Retrieval Manipulator Deployment) in Ref. 33. This means that the arm/mast collision scenario is caused primarily by motion of one joint ('elbow' or 'wrist'), by rotation about a horizontal axis (perpendicular to the mast and support tower direction). For this reason, the trees consider one joint of the arm. Subtrees at the lower level for the other joints are similar, but are not considered critical for the failure scenario considered here. Revision of any of the above assumptions would of course produce variations in some of the sub-trees.

The computers and controller hardware assumed for the manipulator in the fault trees here is also taken from figure 5.2 (Manipulator Retrieval System Control Hardware) in Ref. 33, with the addition of a watchdog processor with independent access to the sensor outputs and high level planner. Also, a secondary network has been added between each of the major processors. Based on our fault tree analysis, recommended additions to the core of the Manipulator Retrieval System Control Hardware (primarily the watchdog processor and a backup network) are shown in Fig. 2.

An initial analysis revealed almost no fault tolerance capability in the system, as discussed earlier. Fig. 3 shows a fault tree for the initial system design shown above in Fig. 1. The structure of the fault tree contains major subtrees for the mechanical components and for the control system computers and sensors. The initial design has no provision for fault tolerance. In the fault tree, this can be observed by the fact that the tree contains only OR-gates. Therefore, a single basic failure event, for example in the sensor analog to digital converter (ADC), will propagate up through all levels of the fault tree and result in the top event failure to remove manipulator arm from the underground storage tank. In Fig. 3, there are 24 such single event cut sets that can lead to the top event failure.

4 FINDINGS/RECOMMENDATIONS

This fault tree analysis identified a number of critical failure paths for the WRM. Suggested additions to the

hardware configuration were made in Fig. 2 and in the fault trees that follow. In particular, it was considered important to consider the use of redundant components where possible. Also, an additional backup computer which acts as a checker or 'watchdog' would improve the reliability of the system. The set of trees shown reflect suggested modifications to the design in Ref. 33 by the authors in Ref. 31. Figure 4 presents an overview of the suggested fault tolerance modifications. They contain additional levels of fault tolerance (represented by the AND-gates in the trees). Notice that the trees shown here are not complete, i.e., the 'leaves' of the tree must be expanded further in a more detailed analysis to explore in more depth the causes of the failures in the leaf circles. Further subtrees giving more detail at the lower level are presented in Ref. 31. However, the purpose of the trees presented here is to demonstrate how, for a given robot design and operating assumptions, a fault tree analysis can be usefully employed to effectively identify failure modes, identify means to make the system more reliable, and support assertions of overall system safety. With the proposed enhancements, the number of single event cut sets has been reduced from 24 to 5. The remaining five single event cut sets are due to failures in peripheral systems, such as the tower support structure, which were beyond the scope of the study. Through the use of hardware redundancy and software protection systems, there are now 4 two-event cut sets and 14 five event cut sets in the revised fault tree. All other cut sets are of higher order.

The main recommendations for modifications to the robot design based on the fault tree analysis include the following. Each enhancement is presented and related to the appropriate fault trees in Figs 4-11.

4.1 Redundant sensors

In the high-level fault tree in Fig. 4, two sensors are shown. This allows feedback information to emanate from a surviving sensor if there is a sensor failure. The use of a redundant sensor provides important back-up information to the robot joint controller in the event of a failure. The use of redundant sensors may prevent the robot from drastically straying off course in the event of a sensor failure.

4.2 Sensor fault detection

Sensor fault detection and reconfiguration algorithms²⁹ running at the joint controller level (Fig. 4) can exploit the benefits of redundant sensors. The function of the fault detection algorithms is to detect failure signatures from the sensors. The sensor

Fig. 2. Modified manipulator retrieval system control hardware.

reconfiguration algorithms' purpose is to isolate faulty sensors and feed only information from functional sensors into the controller (switching from suspect sensors where necessary to prevent a collision). Part of this function is indicated in the 'Sensor Tree' subtree between the individual sensors and the 'Joint Controller' subtree in Fig. 4.

4.3 Kinematic redundancy

In order to provide a level of fault tolerance with respect to manipulator arm mechanism failures, it is recommended to provide for kinematic reconfiguration of the arm following a joint failure. The joint is assumed to be locked by the brakes after failure. In

Fig. 3. Initial fault tree for waste retrieval manipulator concept.

Fig. 6, it is seen that kinematic reconfiguration can help to compensate for actuator failure, for example. Reconfiguration (in this case to avoid collision with the mast) could be performed by using kinematic

redundancy in the system. In such a scheme, the initial manipulator posture is chosen so that the robot path plan can be modified to utilize the remaining functional joints to closely implement the original path

Fig. 4. Fault tree after fault tolerance modifications.

plan in the event of a joint failure. For examples of kinematic redundancy in fault tolerant manipulator design and operation, see Ref. 16.

4.4 Emergency stop

An Emergency Stop (E-stop or red-switch) activated by the operator provides a method to manually stop

the motion of the robot arm. For the effect in the fault trees, see Fig. 7. The addition of an E-stop mechanism provides an important method to prevent collisions of the robot arm with obstacles. Two analogous additional features to be discussed below are recommended for the enhancement of the robot controller. These include a software reflex control module and a hardware/software automatic brake shutdown system.

Fig. 5. Top level fault tree-failure to remove arm from tank.

4.5 Reflex control module

This small segment of code is called immediately before the control command is sent to the actuators at each iteration of the control cycle. Its function is to determine if an erroneous command has somehow been generated by the overall control system, and to

perform an Emergency Shut Down (ESD) if a collision is imminent. In Fig. 7, the placement of the reflex control module is at the four input AND gate that protects from operator or controller errors. For details on successful implementations of such code, see Ref. 34. A related reflex control system can be implemented through the use of a multi-sensor protective 'skin' and associated control system.¹⁷

Fig. 6. Robot mechanism failure: unable to comply with task.

4.6 Hardware limiters

The addition of Hardware Limiter switches for the actuator driver circuitry is recommended. This can be seen in Fig. 8. This feature protects against failure in the drive electronics and provides a simple low-level electrical method to prevent joint movement beyond a particular range.

4.7 Model-based preview

Model-based preview and update software should be used to help plan complex tasks in the underground storage tanks. In Fig. 8, it can be seen how model based preview can help to prevent operator error. Through the use of cameras in the tank it is possible to update the model of the workspace to include previously unknown obstacles. This model generation and updating allows the operator to 'preview' a desired motion before sending the command to the robot.

4.8 Heartbeat monitor

Heartbeat monitoring for all key processors in the manipulator control and monitoring path is recom-

mended. The heartbeat monitor periodically checks the health of each processor, and signals a fault if one is found in one of the processors. The effects of this feature can be seen in Figs 8, 9, and 11. The use of a heartbeat monitor helps to guarantee that each of the processors in the system is meeting the real-time update rate schedule. If a processor is not responding due to an overloaded work schedule or a system crash, then this problem will be detected before data corruption spreads through the overall control system.

4.9 Communication network redundancy

Dual buses/cabling for communications between all key processors in the manipulator control and monitoring path is recommended as shown in Fig. 2. The processor configuration in the initial design contains several processors on a single network. The addition of a second network controller in each processor and a secondary network can provide additional capacity and alternate data routing capability. This allows backup if part of the communications network goes down, as seen in Figs 9 and 11.

Fig. 7. Lower manipulator contact with deployment mast.

4.10 Fail-safe brakes

Fail-safe brakes with dual redundant power hardware units as shown in Fig. 10 should be added to the robot system. The brakes require power-on to release, and therefore a power failure will activate all brakes. Brakes are activated by a manual emergency stop, or by automatic ESD signals from the robot controller, watchdog computer, or reflex control module.

4.11 Watchdog processor

A Watchdog processor, which is shown in Fig. 2, will provide additional monitoring capability independent of the robot joint motion controller. The function of the watchdog is a backup to the fault detection and plan verification routines (Figs 4 and 11). The watchdog processor is fed the sensor outputs and plan, and has redundant connections to the command computer. The watchdog will issue an ESD if a fault leading to a potential collision is detected. The watchdog processor may be implemented with different software routines to provide a level of diversity for software fault tolerance.

4.12 Peripheral system design

Several unresolved issues remain, including: the arm kinematics, the camera and vision system, the hose/pulley system used for waste removal and support, and the single power line supplying the site.

5 RESULTS/DISCUSSION

The contribution of the work outlined in this paper has been to identify and highlight areas in the initial manipulator design which were prone to failure. The fault tree reliability analysis provided a systematic method to highlight the advantages of adding redundant and safety systems. Also, the analysis provides project managers with justification for enhancements to the design. The fault tolerance benefits of these enhancements are expected to more than justify their added cost when total mission success is considered.

In summary, a study of the fault trees reveals that, in order for the event (Failure to Remove Arm from Tank) to occur, at least four other events must occur (four-fault tolerance). Note that the organization of the trees show the safety structure of the manipulator in a logical manner. The trees can also be used to determine weak paths in the manipulator design, to reveal areas where key improvements must be made. Again notice, for example, that without fault detection, a sensor fault propagates directly through the trees.

Another interesting feature of the trees is that several of the AND gates (i.e., fault tolerance options) are provided by technology currently being developed at universities (see Figs 6 and 7). These universities are part of a team coordinated at Sandia National

Fig. 8. Operator/controller command in error.

Laboratories^{12,16,17,20,27,34} to investigate robotics issues (particularly robot safety) in support of the DOE's hazardous waste robotics program. The trees provide support for the usefulness of the algorithms being developed at the universities to the program.

The issues uncovered above, and the recommendations made in Ref. 31, were based on a purely

qualitative fault tree analysis. One problem with quantitative reliability analyses for robot manipulators, and particularly for those deployed in nuclear environments, is that there is a lack of well-trusted failure and reliability data in robotics. Currently, there are a number of efforts which promise to provide better reliability data for robots in

Fig. 9. Incorrect joint position/velocity plan from robot controller.

nuclear environments^{5,9,13,24} which will be extremely useful. In the absence of well-trusted data for some sub-systems of the robot, there exist several techniques for handling uncertain data in fault

trees.^{19,23,32} It is expected that the use of these techniques will assist in reliability analyses of future design evolutions for the Hanford arm, and also other manipulators.

Fig. 10. Automatic brake shutdown failure. **Fig. 11.** Watchdog ESD fails.

6 CONCLUSIONS

The use of fault trees to analyze (in the initial design phase) the reliability of a proposed tank waste retrieval manipulator revealed a number of important areas in the design which could be markedly improved. Recommendations included the addition of redundancy and safety checks in both hardware and software. These recommendations were based on a qualitative analysis of fault trees constructed for the preliminary manipulator design. More in-depth quantitative analyses are made difficult by the lack of well-trusted reliability data for large manipulators in hazardous environments, although there are currently several research efforts aimed at providing better data. Our ongoing work focuses on developing new techniques for quantitative fault tree analyses of robots with uncertain reliability data.³²

ACKNOWLEDGEMENT

This work was supported in part by DOE Sandia National Laboratory Contracts DE-AC04-94AL8500, 18-4379 and AL-3017, in part by the National Science Foundation under grants DDM-9202639 and IRI-9526363, and in part by NASA under grant NAG-9-740.

REFERENCES

- Ballou, R. A., Identification of single-shell tank in-tank hardware obstructions to retrieval at Hanford tank farms. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 580–583, February 1995.
- Barlow, R. E. & Lambert, H. E., Introduction to fault tree analysis. In *Reliability and Fault Tree Analysis: Theoretical and Applied Aspects of System Reliability and Safety Assessment*, (eds R. E. Barlow, J. B. Fussell & N. D. Singpurwalla) SIAM, Philadelphia, pp. 7–35, 1975.
- Cartaret, B. A., Light duty utility arm system applications for tank waste remediation. In *Proc. Am. Nucl. Soc. Top. Meeting on Robotics and Remote Systems*, Monterey CA, pp. 554–561, February 1995.
- Cavallaro, J. R. & Walker, I. D., Rice university robot arm. *The RiskWorks Rev.*, **1** (1995) 1.
- Coenen, S. & Decreton, M., The radiation resistance of force sensors used on telerobotic nuclear equipment. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 159–164, February 1995.
- Cunliffe, G., Light duty utility arm for underground storage tank inspection and characterization. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 562–568, February 1995.
- Dhillon, B. S., *Robot Reliability and Safety*, Springer-Verlag, New York, NY, 1991.
- Gibbons, P. W. & McDaniel, L. B., First generation long-reach manipulator for retrieval of waste from Hanford single-shell tanks. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 575–579, February 1995.
- Giraud, A. & Robiolle, M., Fault tolerant embedded computers and power electronics for nuclear robotics. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 143–150, February 1995.
- Gordon, J. W. & Curry, J. J., Reliability analysis of the RRV-1 robot. In *Proc. 33rd ANS Conference on Remote Systems Technology*, San Francisco, CA, pp. 210–214, February 1984.
- Guthrie, V. H. & Whittle, D. K., RAM analysis software for optimization of servomanipulator designs. *DOE Small Business Innovative Research (SBIR) Program Report JBFA-101-89*, JBF Associates, Inc., Knoxville, TN, Performed for Oak Ridge National Laboratory, 1989.
- Harrigan, R. W., Research on robotics by principal investigators of the robotics technology development program. Sandia Report, SAND94-0844, Albuquerque, NM, 1995.
- Holmes-Siedle, A., Christensen, P., Lauridsen, K. & Watts, S. J., The management of radiation-induced faults in teleoperators used in nuclear plants. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 169–176, February 1995.
- Kiebel, G. R., Control system design for robotic underground storage tank inspection systems. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 569–574, February 1995.
- Lauridsen, K., Christensen, P. & Kongsø, H. E., Assessment of the reliability of robotic systems for use in radiation environments. *Reliab. Engng System Safety*, **53** (1996) 1–2.
- Lewis, C. & Maciejewski, A. A., Dexterity optimization of kinematically redundant manipulators in the presence of joint failures. *Int. J. Computers & Elect. Engng*, **20** (1994) 273–288.
- Lumelsky, V. J. & Cheung, E., Real-time collision avoidance in teleoperated whole-sensitive robot arm manipulators. *IEEE Trans. Systems, Man, & Cybernetics*, **23** (1993) 194–203.
- NASA, Computer Based Control System Non-compliance Report for Computer Independent Hazard Control System, Report, NASA Goddard Flight Centre, Greenbelt, MD, 1991.
- Page, L. B. & Perry, J. E., Standard deviation as an alternative to fuzziness in fault tree models. *IEEE Trans. Reliab.*, **43** (1994) 402–407.
- Paredis, C., Au, A. & Khosla, P., Kinematic design of fault tolerant manipulators. *Int. J. Computers & Elect. Engng*, **20** (1994) 211–220.
- Schlieff, D., Fault tree analysis for the schilling arm. *Technical Report 951020*, Laboratory for Intelligent Automated Systems, Colorado School of Mines, 1995.
- Shen, E. J., Retrieval of underground storage tank wastes: the Hanford challenge. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 549–553, February 1995.
- Tanaka, H., Fan, L. T., Lai, F. S. & Toguchi, K., Fault-tree analysis by fuzzy probability. *IEEE Trans. Reliab.*, **32** (1983) 453–457.
- Tulenko, J. S., Ekdahl, D., Liu, H., Phillips, K., Jones, S., Cable, T. & Harvey, H., Development of a radiation resistant ANDROS robot for operation in severe environments. In *Proc. Am. Nucl. Soc. Topical Meeting on Robotics and Remote Systems*, Monterey, CA, pp. 165–168, February 1995.

25. Vesley, W. E., Goldberg, F. F., Roberts, N. H. & Haasi, D. F., Fault Tree Handbook, NUREG 0492, Systems and Reliability Research Office of Nuclear Regulatory Research, US Nuclear Regulatory Commission, Washington D.C. 20555, 1981.
26. Visinsky, M. L., Walker, I. D. & Cavallaro, J. R., Robotic fault tolerance: algorithms and architectures. In *Robotics and Remote Systems for Hazardous Environments*, volume 1, Prentice Hall Publishing Co., Englewood Cliffs, NJ, 1993 pp. 53-73
27. Visinsky, M. L., Cavallaro, J. R. & Walker, I. D., Expert system framework for fault detection and fault tolerance in robotics. *Int. J. Computers & Elect. Engng*, **20** (1994) 421-436.
28. Visinsky, M. L., Cavallaro, J. R. & Walker, I. D., Robotic fault detection and fault tolerance: a survey. *Reliab. Engng System Safety*, **46** (1994) 139-158.
29. Visinsky, M. L., Cavallaro, J. R. & Walker, I. D., A dynamic fault tolerance framework for remote robots. *IEEE Trans. Robotics & Automation*, **11** (1995) 477-491.
30. Walker, D. A., Creating reliability in the design process: an evaluation approach, design tool, and example application. Master's thesis, Department of Mechanical Engineering, University of Tennessee, Knoxville, TN, 1990.
31. Walker, I. D. & Cavallaro, J. R., Failure mode analyses of the Hanford manipulator. *Technical Report #9402*, Department of Electrical and Computer Engineering, Rice University, 1994.
32. Walker, I. D. & Cavallaro, J. R., The use of fault trees for the design of robots for hazardous environments. In *Proc. IEEE Ann. Reliab. & Maintainability Symp.*, Las Vegas, NV, pp. 229-235, 1996.
33. Wallace, D. A., Project W-340 Manipulator Retrieval System for Tank 241-C-106. *Technical Report WHC-SD-W340-ES-001*, Westinghouse Hanford Company, Richland, WA, 1993.
34. Wikman, T., Branicky, M. & Newman, W., Reflex control for robot system preservation, reliability, and autonomy. *Int. J. Computers & Elect. Engng*, **20** (1994) 391-407.
35. Young, J., Using the fault tree analysis technique. In *Reliability and Fault Tree Analysis: Theoretical and Applied Aspects of System Reliability and Safety Assessment*, (eds R.E. Barlow, J.B. Fussell, N.D. Singpurwalla), SIAM, Philadelphia, PA, 1975, pp. 827-848.