Incorporating Process Mining into Human Reliability Analysis

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus, prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op woensdag 31 augustus 2011 om 16.00 uur

door

Dana Lee Kelly

goingen te Spartanburg, Verenigde Staten van Amerika
Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. A.C. Brombacher
en
prof.dr. D.M. Karydas

Copromotor:
dr.ir. J.L. Rouvroye

Copyright © 2011 by Dana L. Kelly

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the copyright owner.

A catalogue record is available from the Eindhoven University of Technology Library.

ISBN: 978-90-386-2503-4

Printed by: University Printing Office, Eindhoven

Cover Design by: Paul Verspaget
Acknowledgements

I would first like to thank Dr. Dimitrios Karydas for encouraging me to undertake this work, and for introducing me to Prof. Aarnout Brombacher at TU/e. Thanks to Prof. Brombacher for introducing me to the notion of process mining and its potential applications in technological risk assessment. And thanks to the government of The Netherlands for making my participation in this program possible.

I would like to thank all of the members of my committee for their thoughtful comments on early drafts of this dissertation. Their comments have improved the content substantially.

Dr. Jan Rouvroye helped me navigate the institutional waters at TU/e. His tenacity and attention to detail are greatly appreciated.

I would like to thank the Idaho National Laboratory for the support it has provided during this work. Special thanks are due to Bruce Hallbert and Marty Sattison in this regard, who have insisted that this research take priority over other work that I have been involved with.

I also would like to thank the many colleagues who have provided encouragement and help along the way. Andreas Bye, Michael Hildebrandt, and Helena Broberg were instrumental in providing data from the Halden simulator and the follow-up benchmark experiment at the U.S. plant. Ali Mosleh, Song Shen, and Johanna Oxstrand have worked with me on the development of the new hybrid HRA method described in Ch. 5. Seba Jean-Baptiste helped with the Java programming needed to convert the simulator log files; without Seba’s help, this work would not have been possible. At TU/e, Anne Rozinat and Christian Guenther spent hours discussing the details of process mining with me, and answering my many questions.

Thanks are due to Erasmia Lois at the U.S. Nuclear Regulatory Commission for allowing me to have access to the data from the Halden and U.S. plant simulators.

Lastly, very special thanks to my good friend Curtis Smith for nagging me incessantly about undertaking and finishing this work.
Incorporating Process Mining into Human Reliability Analysis

Summary

It is well established that the human contribution to the risk of operation of complex technological systems is significant, with typical estimates lying in the range of 60-85%. Human errors have been a contributor to many significant catastrophic technological accidents. Examples are 1) the termination of safety injection during the Three Mile Island accident, leading to extensive damage to the reactor core; 2) the introduction of water into the methyl isocyanate storage tank at the Union Carbide facility in Bhopal, India, which led to a large uncontrolled release and thousands of offsite fatalities; 3) the series of deliberate violations, leading to an explosion, combustion of the graphite moderator, and uncontrolled release of radioactivity at the Chernobyl nuclear plant in Ukraine (Reason, 1990). Therefore, in order to adequately characterize and quantify the risk of complex technological systems, the human contribution must be included in the risk assessment.

Human reliability analysis, a component of an integrated probabilistic risk assessment, is the means by which the human contribution to risk is assessed, both qualitatively and quantitatively. Human reliability analysis as a discipline has as its goals the identification, modeling, and quantification of human failure events in the context of an accident scenario. There are literally dozens of human reliability analysis methods to choose from, good practices have been developed for human reliability analysis, many of the methods have been evaluated against these good practices, and new methods are still being developed in the U.S. and other countries around the world. However, many difficulties remain. A principal difficulty, and one that hampers use of human reliability analysis results in risk-informed decision-making, is the large variability associated with the analysis results, from one method to another, and between analysts for a given method.

An important part of any comprehensive human reliability analysis is a task analysis. Task analysis is the name given to a range of techniques that can be used to examine the ways in which humans undertake particular tasks. Some of these techniques focus directly upon task performance, while others consider how specific features of the task, such as the interfaces, operating procedures, and team organization or training, can influence task performance. An important ingredient of the task analysis, however it is performed, is observations from system simulators. These observations are important in order for the analysis team to be able to realistically model procedure implementation, interactions between the crew and the system, and interactions among the crewmembers themselves during low-frequency high-consequence scenarios, for which direct observational data on human performance are lacking. Without such observations, the HRA is likely to deviate significantly from reality.

Simulator observations are also a major source of information for some of the newer human reliability analysis methods, and guidelines promulgated by the U.S. nuclear industry emphasize the importance of gathering simulator data. However, the industry guidelines do not provide guidance on how analysts could benefit from the wealth of information that can be obtained from observing simulator exercises to support understanding of crew characteristics and behavior, and
other general plant-specific factors that could influence performance in particular scenarios. A
current use of simulator studies is to inform the development of new human reliability analysis
methods, and this effort is faced with the same lack of guidance on effectively and efficiently
employing the abundance of information produced by these simulator studies.

Put another way, what needs to go into an HRA is well understood. The problem the analysis
community has faced for many years is how. The resources required to analyze the resulting
information are likely a reason for the infrequent use of simulator observations in support of the
human reliability task analysis. The goals of the qualitative analysis are to provide insights about
process improvements that reduce risk, and to produce a model of operator performance for later
quantification, in particular the principal process (and deviations from this process) followed by
operators in responding to a plant upset condition. What is missing are tools to allow analysts to
more efficiently and consistently make use of the sometimes vast amount of information gathered
in the qualitative analysis, particularly during observations of operator responses in plant
simulators.

This research illustrates how select process mining tools, applied to event logs from a facility
simulator, can be used to efficiently develop a model of operator performance in an accident
scenario, including both the nominal process and significant deviations from this process, which
could lead to risk-significant errors of commission. Such errors are known to be important
contributors to risk, but have heretofore been largely absent from risk analyses of complex
technological systems. This represents an advance in human reliability task analysis, which
requires input from simulator observations. The dissertation explores the following four research
questions:

1. What are the requirements for a tool to aid in the analysis of large amounts of simulator
data in support of human reliability analysis?

2. How do current human reliability analysis methods approach the issue of simulator
observations and are these approaches suitable for incorporating simulator observations
into the human reliability task analysis?

3. Are there tools in other domains that are more suitable and which, if adopted (and
adapted to their new domain), could improve the state of the art in human reliability
modeling and task analysis?

4. What are the limits of applicability of these tools from other domains, and what
improvements are needed in order to make them practical for use by an analyst who is
not a specialist in using such tools?

The first question is explored in Ch. 2, which examined the overall human reliability analysis
process. The following characteristics were identified from experience and a literature review as
factors to be considered in human reliability modeling:

- Plant behavior and conditions
- Timing of events and the occurrence of human action cues
Parameter indications used by the operators and changes in those parameters as the scenario proceeds

- Time available and locations necessary to implement the human actions
- Equipment available for use by the operators based on the sequence
- Environmental conditions under which the decision to act must be made and the actual response must be performed
- Degree of training, guidance, and procedure applicability.

The first three and the last of these can be informed by simulator observations. However, simulators can produce very large output files, in a variety of formats. Manually analyzing such output data is very resource intensive, and has in the past limited the use of simulator experiments and observations in support of human reliability analysis. Thus, one requirement for an analysis tool is that it be capable of accepting data in a flexible format, and that it be able to handle large amounts of data. A second requirement is that the analysis cannot be purely statistical, because the human reliability analysis is concerned with the process followed by the operators, and not solely with statistical variables, such as the time at which a certain action is performed.

Ch. 2 also examined the task analysis guidance provided by two representative human reliability analysis methods, THERP and ATHEANA, both of which are considered complete methods, in that they address all three aspects of the analysis: identification, modeling, and quantification. In addition to these two methods, Ch. 2 also examined other approaches for human reliability task analysis. The conclusion of these examinations was that there are no extant tools in the human reliability community of practice that are suitable for analyzing large amounts of simulator data in the context of a human reliability task analysis.

In examining the third research question, Ch. 3 provided an overview of business process mining tools, along with some selected industrial applications of these tools, and concluded that these tools have potential for application in support of human reliability task analysis specifically, and simulator data analysis more generally. To begin answering the fourth research question, Ch. 3 examined some of the process mining tools and techniques in the context of analyzing simulator data. The most promising tool appeared to be the fuzzy mining algorithm developed by Christian Guenther as part of his PhD research at TU/e. Because simulator log files are typically very large, traditional process mining approaches can be expected to produce an overly complex “spaghetti model” that would be quite opaque to analysis. The fuzzy model abstracts away irrelevant details, leaving the salient aspects of interest for the task analysis.

Ch. 4 began exploring how the tools of process mining might be applied to simulator data, beginning with a relatively small set of logs collected at the Halden Reactor Project simulator in Norway. A number of difficulties were encountered during conversion of the data files to the format required by the process mining software. These problems became even more severe in the application of Ch. 6, which involved much larger event logs. Despite the problems with file conversion, Ch. 4 concluded that certain process mining tools, especially the fuzzy miner, had the potential to be of use in support of human reliability task analysis, because they could clearly highlight differences in the underlying process governing each crew’s performance.
Ch. 6 continued the examination of the fourth research question by exploring the application of process mining tools to a much larger set of simulator data collected at a U.S. plant. File conversion was found to be a particularly severe problem, worse than for the Halden data analyzed in Ch. 4, and considerable time had to be spent in writing a file conversion routine. Following file conversion, considerable up-front manual filtering of the simulator action logs to remove low-level actions was still necessary to reduce the complexity of the mined models. Such filtering has the potential to introduce errors into the resulting mined models, and so the analyst who does the filtering must have detailed knowledge of facility procedures and operations, or have access to someone who does, to ensure that such errors are not introduced.

Applying the fuzzy miner to the filtered logs provided some especially useful insights for human reliability task analysis, and particularly for construction of the crew response trees being considered for use in the new hybrid human reliability analysis method described in Ch. 5. This method is not being developed as part of the research described herein, although the author is part of the team that is developing the method.

The scientific contributions of this research are as follows.

- This research illustrates how process mining, applied to event logs from a facility simulator, can be used to efficiently develop a model of operator performance in an accident scenario, including both the nominal process and significant deviations from this process, which could lead to risk-significant errors of commission. Such errors are known to be important contributors to risk, but have heretofore been largely absent from risk analyses of complex technological systems. This represents an advance in human reliability task analysis, which requires input from simulator observations.

- This research illustrates how process mining can aid in the construction of crew response trees, which are a proposed framework for task analysis and quantification in a new hybrid human reliability analysis method being developed by the U.S. Nuclear Regulatory Commission, in collaboration with a team (of which the author is a member) comprising researchers from Sandia National Laboratories, Idaho National Laboratory, the University of Maryland, the Electric Power Research Institute, and the Paul Scherer Institute.

- This research illustrates the potential for process mining to improve data reduction and analysis for future simulator experiments at the Halden Reactor Project and elsewhere. It also illustrates some of the limitations in current process mining tools, which will need to be overcome in order for these techniques to be able to be applied broadly by analysts in the field who are not process mining specialists.

Several potential future contributions of process mining to human reliability analysis and facility analysis more generally have been identified in this research, although these are not explored in detail in this dissertation. These are 1) the potential to employ process mining in post-processing of data produced by dynamic simulation models being developed by the risk analysis research community, 2) expanding the process model by incorporating process variables such as pressure and temperature, which are often collected at very short intervals (e.g., 1 msec), 3) use of the underlying Petri net model produced by process mining to simulate operator performance in an
accident scenario, 4) use of process mining to post-process data produced by dynamic PRA simulation tools, and 5) use of process mining to identify process deviations in a nuclear reprocessing facility, where such deviations could be indicative of an attempt to divert special nuclear material from the facility.
# Table of Contents

Acknowledgements .................................................................................................................... 4  
Incorporating Process Mining into Human Reliability Analysis ............................................. 6  
Summary .................................................................................................................................... 6  
Acronyms and Abbreviations .................................................................................................. 15  
1 Introduction ........................................................................................................................... 18  
  1.1 Research questions ......................................................................................................... 20  
  1.2 Contribution of research .............................................................................................. 21  
2 Overall framework for human reliability analysis ............................................................. 24  
  Research question ................................................................................................................ 24  
  2.1 Major elements in HRA ................................................................................................. 25  
  Research question ................................................................................................................ 27  
  2.2 Illustrative HRA methods .............................................................................................. 27  
    2.2.1 Technique for Human Error Rate Prediction (THERP) .......................................... 28  
    2.2.2 A Technique for Human Event Analysis (ATHEANA) ......................................... 31  
  2.3 Role of task analysis in HRA ......................................................................................... 33  
  2.3.1 Representative examples of HRA Task Analysis ....................................................... 33  
    2.3.1.1 THERP ................................................................................................................. 34  
    2.3.1.2 ATHEANA .......................................................................................................... 35  
  2.4 Need to incorporate data from simulator studies into HRA task analysis ..................... 42  
  2.5 Approaches for linking simulator data to HRA task analysis ........................................ 43  
  2.6 Summary of where current methods fall short in supporting HRA task analysis.......... 44  
3 Overview of process mining tools and applications ............................................................. 46  
  Research question ................................................................................................................ 46  
  3.1 Process mining overview .............................................................................................. 46  
  3.2 Review of process mining applications ......................................................................... 48  
    3.2.1 Process mining applied to wafer testing in ASML .................................................. 48
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2.2 Application of process mining in healthcare</td>
<td>53</td>
</tr>
<tr>
<td>Research question</td>
<td>56</td>
</tr>
<tr>
<td>3.3 Mining a Petri net process model from simulator event logs</td>
<td>56</td>
</tr>
<tr>
<td>3.4 Extensions of process mining to flexible processes and noisy event logs</td>
<td>59</td>
</tr>
<tr>
<td>3.4.1 Heuristic nets</td>
<td>60</td>
</tr>
<tr>
<td>3.4.2 Fuzzy models</td>
<td>62</td>
</tr>
<tr>
<td>3.4.3 Trace clustering</td>
<td>66</td>
</tr>
<tr>
<td>3.5 Summary</td>
<td>68</td>
</tr>
<tr>
<td>4 Application of process mining to international HRA empirical study</td>
<td>70</td>
</tr>
<tr>
<td>Research question</td>
<td>70</td>
</tr>
<tr>
<td>4.1 Obtaining Halden simulator data for case study</td>
<td>70</td>
</tr>
<tr>
<td>4.2 Overview of international HRA empirical study</td>
<td>73</td>
</tr>
<tr>
<td>4.2.1 Summary of crew performance for base case SGTR scenario</td>
<td>79</td>
</tr>
<tr>
<td>4.3 Application of process mining to Halden simulator data</td>
<td>80</td>
</tr>
<tr>
<td>4.3.1 Application of clustering algorithms to Halden data</td>
<td>85</td>
</tr>
<tr>
<td>4.3.2 Insights for HRA task analysis provided by process mining</td>
<td>86</td>
</tr>
<tr>
<td>4.4 Conclusions from application of process mining to HRA empirical study simulator data</td>
<td>90</td>
</tr>
<tr>
<td>5 Overview of new hybrid HRA method</td>
<td>92</td>
</tr>
<tr>
<td>5.1 Underlying human performance model</td>
<td>92</td>
</tr>
<tr>
<td>5.2 Proposed analysis approach</td>
<td>94</td>
</tr>
<tr>
<td>5.2.1 Example</td>
<td>97</td>
</tr>
<tr>
<td>5.3 Relationship of human performance model to HRA task analysis</td>
<td>100</td>
</tr>
<tr>
<td>Research question</td>
<td>101</td>
</tr>
<tr>
<td>6 Extending the international HRA empirical study</td>
<td>102</td>
</tr>
<tr>
<td>6.1 Project research questions and goals of the follow-on benchmarking exercise</td>
<td>102</td>
</tr>
<tr>
<td>6.1.1 Study methodology</td>
<td>103</td>
</tr>
</tbody>
</table>
C.2 Fuzzy model animation .............................................................................................. 192

Appendix D  Overview of Nuclear Reactor Systems ............................................................ 196

D.1 Primary system........................................................................................................... 196

D.2 Secondary system....................................................................................................... 196

D.3 Support and emergency systems ................................................................................ 198

Appendix E  Overview of Emergency Procedures ............................................................. 200

E.1 Operator role in control room emergency operation ................................................... 200

E.2 Framework for emergency operations ......................................................................... 201

Curriculum Vitae ................................................................................................................... 206
<table>
<thead>
<tr>
<th>Acronyms and Abbreviations</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC: Alternating current</td>
</tr>
<tr>
<td>ACRS: Advisory Committee on Reactor Safeguards</td>
</tr>
<tr>
<td>ADS: Advanced Dynamic Simulation</td>
</tr>
<tr>
<td>AFW: Auxiliary feedwater</td>
</tr>
<tr>
<td>ARP: Alarm response procedure</td>
</tr>
<tr>
<td>ATHEANA: A Technique for Human Event Analysis</td>
</tr>
<tr>
<td>BIT: Boron injection tank</td>
</tr>
<tr>
<td>BPM: Business process mining</td>
</tr>
<tr>
<td>CD: Core damage</td>
</tr>
<tr>
<td>CRT: Crew response tree</td>
</tr>
<tr>
<td>CSF: Critical safety function</td>
</tr>
<tr>
<td>DC: Direct current</td>
</tr>
<tr>
<td>EFC: Error-forcing context</td>
</tr>
<tr>
<td>EOP: Emergency operating procedure</td>
</tr>
<tr>
<td>ERG: Emergency Response Guidelines</td>
</tr>
<tr>
<td>HAMMLAB: Halden.human-Machine Laboratory</td>
</tr>
<tr>
<td>HEP: Human error probability</td>
</tr>
<tr>
<td>HFE: Human failure event</td>
</tr>
<tr>
<td>HRA: Human reliability analysis</td>
</tr>
<tr>
<td>ISLOCA: Intersystem loss of coolant accident</td>
</tr>
<tr>
<td>LOFW: Loss of feedwater</td>
</tr>
<tr>
<td>MFW: Main feedwater</td>
</tr>
<tr>
<td>MSIV: Main steam isolation valve</td>
</tr>
<tr>
<td>MSR: Moisture separator re heater</td>
</tr>
<tr>
<td>mxml: Mining extensible markup language</td>
</tr>
<tr>
<td>NR: Narrow range</td>
</tr>
<tr>
<td>NRC: Nuclear Regulatory Commission</td>
</tr>
</tbody>
</table>
OECD: Organization for Economic Cooperation and Development
PAIS: Process-aware information system
PORV: Power-operated relief valve
PRA: Probabilistic risk analysis
PSF: Performance shaping factor
RCP: Reactor coolant pump
RCS: Reactor coolant system
RHR: Residual heat removal
SAMG: Severe Accident Management Guidelines
SBO: Station blackout
SG: Steam generator
SI: Safety injection
SRM: Staff Requirements Memorandum
TALENT: Task Analysis Linked Evaluation Technique
TDAFW: Turbine-driven auxiliary feedwater
THERP: Technique for Human Error Rate Prediction
1 Introduction

It is well established that the human contribution to the risk of operation of complex technological systems is significant, with typical estimates lying in the range of 60-85% (Reason, 1990). “Risk” for this dissertation is taken to have the meaning of a triplet of scenario, likelihood, and consequences, as articulated by (Kaplan & Garrick, 1981). Besides equipment failure, the human operator of a complex system can contribute to each component of this risk triplet, through acts of both omission and commission. In particular, complex commission errors have been a contributor to many significant catastrophic technological accidents. Examples are 1) the termination of safety injection during the Three Mile Island accident, leading to extensive damage to the reactor core; 2) the introduction of water into the methyl isocyanate storage tank at the Union Carbide facility in Bhopal, India, which led to a large uncontrolled release and thousands of offsite fatalities; 3) the series of deliberate violations, leading to an explosion, combustion of the graphite moderator, and uncontrolled release of radioactivity at the Chernobyl nuclear plant in Ukraine (Reason, 1990). Therefore, in order to adequately characterize and quantify the risk of complex technological systems, the human contribution must be included in the risk assessment.

Human reliability analysis (HRA), a component of an integrated probabilistic risk assessment (PRA), is the means by which the human contribution to risk is assessed, both qualitatively and quantitatively. HRA as a discipline has as its goals the identification, modeling, and quantification of human failure events (HFE) in the context of an accident scenario. Analysts have included assessments of human reliability in military system safety evaluations since the 1960s (Swain, 1963), but the first widely publicly available guidance for HRA was described in the WASH-1400 report (U. S. Nuclear Regulatory Commission, 1975), which addressed the safety of nuclear power plants in the U. S. The Technique for Human Error-Rate Prediction (THERP) HRA method (Swain & Guttman, 1983), which evolved from the HRA performed for WASH-1400, provided the first systematic method of identifying, modeling, and quantifying human errors, and is viewed as the father of HRA methods today.

At the time THERP was published, HRA as a discipline was barely beyond its infancy (Swain & Guttman, 1983). A generation later, there are literally dozens of HRA methods to choose from, Good Practices have been developed for HRA (Kolaczkowski et al., 2005), many of the HRA methods have been evaluated against these Good Practices (Forester et al., 2006), and new HRA methods are still being developed in the U.S. and other countries around the world. However, many difficulties remain. A principal difficulty, and one that hampers use of HRA results in risk-informed decision-making, is the large variability associated with the analysis results, from one method to another, and between analysts for a given method. This was a difficulty first highlighted by the so-called Ispra study of 1989, in which four orders of magnitude were observed among the estimates of human error probability developed by teams analyzing a common benchmark problem (Commission of the European Communities, 1989). The more recent international HRA empirical study, sponsored by the U. S. Nuclear Regulatory Commission, has found that, twenty years after the Ispra study was published, disturbingly large variability appears to remain (U. S. Nuclear Regulatory Commission, 2010).

An important part of any comprehensive HRA is a task analysis. Task analysis is the name given to a range of techniques that can be used to examine the ways in which humans undertake
particular tasks. Some of these techniques focus directly upon task performance, while others consider how specific features of the task, such as the interfaces, operating procedures, and team organization or training, can influence task performance.

Because PRA, and by extension, HRA, focuses on low-frequency/high-consequence scenarios, empirical data on task performance in actual scenarios are lacking. Surrogate data can be gathered by collecting data on task performance and other relevant factors from a facility simulator. The value of such simulators to HRA has long been acknowledged within the community of practice. (Swain & Guttman, 1983) and (Kolaczkowski et al., 2005) characterized observations from such simulators as an important ingredient in the task analysis, however it is performed. These observations are important in order for the HRA team to be able to realistically model procedure implementation, interactions between the crew and the system, and interactions among the crewmembers themselves during low-frequency high-consequence scenarios, for which direct observational data on human performance are lacking. Without such observations, the HRA is likely to deviate significantly from reality. Failure to carry out such observations has been cited as a weakness in applications of some of the major HRA methods, such as THERP (Forester et al., 2006). Simulator observations are also a major source of information for some of the newer second-generation HRA methods, and guidelines promulgated by the U.S. nuclear industry\(^1\) emphasize the importance of gathering simulator data (Wakefield et al., 1992). However, as pointed out by (Forester et al., 2006), the industry guidelines do not provide guidance on how analysts could benefit from the wealth of information that can be obtained from observing simulator exercises to support understanding of crew characteristics and behavior, and other general plant-specific factors that could influence performance in particular scenarios. A current use of simulator studies is to inform the development of new HRA methods, and this effort is faced with the same lack of guidance on effectively and efficiently employing the abundance of information produced by these simulator studies.

The Ispra HRA benchmark study (Commission of the European Communities, 1989) recognized the importance of simulator observations, and thus the analysis teams were provided with a video showing the operator teams performing the tasks that were to be analyzed. However, lack of tools and procedures for incorporating this information into the task analysis hampered its use and the resulting variability in the task analysis was judged to be a contributor to the large variability in the human error probabilities produced by the analysis teams participating in the study.

Shortly after the publication of the Ispra HRA benchmark study, a large-scale HRA effort was undertaken as part of the risk analysis performed to better understand the contribution of intersystem loss-of-coolant accidents (ISLOCA) to the risk of U. S. nuclear plant operation (Galyean W. J., et al., 1991), (Kelly et al., 1992a), (Kelly et al., 1992b), (Galyean W. J. et al., 1993), (Galyean W. J., et al., 1994). For these studies, simulator observations were made as part

\(^1\) HRA has largely been developed by the nuclear industry, although it has been applied in other industries, such as medicine and aerospace. Current HRA research, such as the empirical studies described in this dissertation, is being carried out by the nuclear industry; thus, there is an unavoidable nuclear reactor context to most recent work in HRA. Appendix D provides a brief overview of typical nuclear reactor systems mentioned in this dissertation.
of a two-week visit by the analysis team to each plant being analyzed. A detailed data collection form was developed and this form, using the guidance in (Well et al., 1990), and was filled in manually by the analysts observing the simulator exercises. Needless to say, this approach is both time-consuming and inefficient. Again, tools were lacking to aid in incorporating the simulator observations into the HRA task analysis.

In both the Ispra study and the ISLOCA evaluations, the analysis teams had in-depth knowledge of the systems being analyzed. In other applications, this may not be the case. For example, (Kelly et al., 1993) describes a novel application of HRA in the medical domain. In this application, the analysts were not familiar with the treatment modality under analysis. As a result, much time was spent interviewing physicians and other treatment staff, observing patient treatments, and collating the resulting information to produce a process model, including salient variations in the process, which could then be analyzed from a human reliability perspective.

The recently completed international HRA empirical study (U. S. Nuclear Regulatory Commission, 2010) has also highlighted the importance of simulator observations to the HRA task analysis. One of the participants in the study has even extended the role of simulator observations beyond the task analysis, noting that “simulator studies provide rich qualitative and quantitative data sources, and their usage would lend credibility to HRA overall, particularly for existing plant[s],” (Corporate Risk Associates, Ltd., 2009). A similar conclusion was put forth by the Organisation for Economic Co-operation and Development in a 2008 report on recommended actions to support the collection and exchange of HRA data (Committee on the Safety of Nuclear Installations, 2008). This report identified data collection in nuclear power plant training and research simulators as a priority for future activities by the Nuclear Energy Agency.

As pointed out above, a principal difficulty with simulator studies is how to make efficient use of the wealth of information that is collected in a typical study. Tools are lacking to aid in analyzing large amounts of simulator data efficiently and translating the data into the information required for the task analysis. There are no tools in the HRA domain that can aid directly with this in the HRA task analysis, by identifying underlying models of crew performance, and highlighting deviations from expected performance, along with crew-to-crew variations. Such deviations from expected performance are an important ingredient in analyzing complex commission errors, which are a focus of recently developed HRA methods such as ATHEANA (U. S. Nuclear Regulatory Commission, 2000) and (Forester et al., 2007).

A common thread through all of the HRA studies described above is the lack of tools to aid in efficiently identifying the underlying process (including variations) that operators of complex technological systems follow, and in incorporating large amounts of data from facility observations into the HRA task analysis. This dissertation will explore one potential solution to this problem.

1.1 Research questions

An HRA that follows accepted good practices should incorporate data from a system simulator into the task analysis. Such data should include operator actions taken, process variable values as a function of time, because such variables influence operator behavior, and alarms and annunciators received by the operators. Put another way, what needs to go into an HRA is well understood. The problem the community has faced for many years is how. The resources
required to analyze the resulting information are likely a reason for the infrequent use of simulator observations in support of the HRA task analysis. The goals of the qualitative analysis are to provide insights about process improvements that reduce risk, and to produce a model of operator performance for later quantification, in particular the principal process (and deviations from this process) followed by operators in responding to a plant upset condition. What is missing are tools to allow analysts to more efficiently and consistently make use of the sometimes vast amount of information gathered in the qualitative analysis, particularly during observations of operator responses in plant simulators.

RQ1: What are the requirements for a tool to aid in the analysis of large amounts of simulator data in support of HRA?

RQ2: How do current HRA methods approach the issue of simulator observations and are these approaches suitable for incorporating simulator observations into the HRA task analysis?

These questions will be investigated in Ch. 2, which lays out the overall framework for HRA and examines representative existing approaches to HRA modeling and task analysis.

RQ3: Are there tools in other domains that are more suitable and which, if adopted (and adapted to their new domain), could improve the state of the art in HRA modeling and task analysis?

This question will be examined beginning in Ch. 3.

RQ4: What are the limits of applicability of these tools from other domains, and what improvements are needed in order to make them practical for use by an analyst who is not a specialist in such tools?

This last question will be examined in Chapters 4-6.

1.2 Contribution of research

This research illustrates how process mining, applied to event logs from a facility simulator, can be used to efficiently develop a model of operator performance in an accident scenario, illustrating both the nominal process and significant deviations from this process, which could lead to risk-significant errors of commission. Such errors are known to be important contributors to risk, but have heretofore been largely absent from risk analyses of complex technological systems. This represents an advance in HRA task analysis, which requires input from simulator observations.

This research illustrates how process mining can aid in the construction of crew response trees, which are a proposed framework for task analysis in a new hybrid HRA method being developed by the U.S. Nuclear Regulatory Commission, in collaboration with a team (of which the author is a member) comprising researchers from Sandia National Laboratories, Idaho National Laboratory, the University of Maryland, the Electric Power Research Institute, and the Paul Scherer Institute.
This research illustrates the potential for process mining to improve data reduction and analysis for future simulator experiments at the Halden Reactor Project and elsewhere. It also illustrates some of the limitations in current process mining tools, which will need to be overcome in order for these techniques to be able to be applied broadly by analysts in the field who are not process mining specialists.

Several potential future contributions of process mining to HRA and facility analysis more generally have been identified, although these are not explored in detail in this dissertation. These are 1) the potential to employ process mining in testing computerized procedures, 2) expanding the process model by incorporating process variables such as pressure and temperature, which are often collected at very short intervals (e.g., 1 msec), 3) use of the underlying Petri net model produced by process mining to simulate operator performance in an accident scenario, 4) coupling of process mining to advanced simulation tools used for dynamic PRA, and 5) use of process mining to identify process deviations in a nuclear reprocessing facility, where such deviations could be indicative of an attempt to divert special nuclear material from the facility.
2 Overall framework for human reliability analysis

As discussed in Ch. 1, human reliability analysis (HRA) is the means by which the human contribution to risk is assessed, both qualitatively and quantitatively. HRA as a discipline has as its goals the identification, modeling, and quantification of human failure events (HFE) in the context of an accident scenario. These three goals are the main elements in the overall HRA framework described in (Kolaczkowski et al., 2005). This chapter first discusses these HRA elements in some detail, along with other important HRA characteristics, such as team formation. It then describes two representative HRA methods, one so-called first-generation method, and a newer second-generation method. It next describes the role of task analysis in HRA, and illustrates how task analysis supports (or is intended to support) the two representative HRA methods. Finally, and most relevant to this dissertation, it discusses the need to incorporate simulator data into the HRA task analysis, and reviews existing methods for doing so.

As will be discussed in more detail below, HRA attempts to predict how humans will perform when interacting with a complex technological system, most often in the context of a scenario that is unlikely to occur (i.e., low frequency), but which can have very undesirable consequences if it should occur. Ideally, in making its predictions about operator performance, HRA methods would utilize data from actual scenarios; such data would be of the highest fidelity in terms of actual system response. Of course, because such scenarios occur rarely, data of this form are thankfully lacking.

Facility simulators have been developed in various industries (e.g., aviation, nuclear), largely for the purpose of training operators in both normal (high-frequency scenarios) and abnormal (low-frequency scenarios) facility operation. Note that these simulators, with few exceptions, were not developed with the goal of supporting HRA in mind. However, because facility simulators provide an environment of ever increasing realism, they can provide extremely valuable insights into operator performance, especially in the low-frequency high-consequence scenarios of most concern to HRA. Because we cannot crash planes and melt reactor cores in our zeal to create realistic HRA models, we must turn to facility simulators, which substitute a virtual facility, allowing any number of complex scenarios to be run with enough realism that valid data can be collected on operator performance in these scenarios.

However, given that these facility simulators have been designed primarily to support operator training, we must consider the question of how information produced during a simulation can be best used in support of HRA. This is the first research question, which will be explored by examining the major elements of the HRA process, and the requirements they place on simulator data collection and analysis.

Research question

What are the requirements for a tool to aid in the analysis of large amounts of simulator data in support of HRA?
2.1 Major elements in HRA

Human reliability is defined by (Swain & Guttman, 1983) as “the probability that a person (1) correctly performs some system-required activity in a required time period (if time is a limiting factor) and (2) performs no extraneous activity that can degrade the system.” This is still a useful working definition of human reliability today. To estimate the human error probability (HEP) suggested by this definition, one requires as inputs complete and accurate information on human factors considerations in the context of both normal and abnormal system operation. This in turn requires a qualitative analysis of the human-system interaction, and presents an HRA practitioner with a significant challenge: acquisition of human performance data that is as complete and accurate as possible, and of related causal information, in the context of normal and abnormal operational settings, to support more realistic evaluations of system unreliability and risk. The acquisition of this information, which can be quite resource-intensive, is the goal of the HRA task analysis, described in a later section of this chapter.

The three main elements of HRA listed in (Kolaczkowski et al., 2005) are HFE identification, modeling, and quantification. Each of these elements is discussed by (Kolaczkowski et al., 2005) for two broad classes of events. The first is pre-initiator HFEs (also called latent HFEs), which occur prior to the accident sequence initiating event and complicate operator response to the initiator. The second, which encompasses the majority of the HRA, is post-initiator HFEs, which are errors made in responding to the accident sequence initiating event. The focus in this dissertation is on post-initiator HFEs, as these are the ones for which simulator observations can be useful. It will also focus on modeling of HFEs rather than identification and quantification for, as pointed out in (Kolaczkowski et al., 2005), constructing HRA models is the activity for which simulator observations are most useful.

As described in (Kolaczkowski et al., 2005), for a risk assessment of a complex system to realistically include human actions, the HRA modeling must consider human actions in the context of a complete accident scenario (i.e., a sequence of events leading to transgression of the envelope of safety for the system under analysis). Such an accident scenario will typically be a low-frequency concatenation of both hardware and human behavior, and the hardware performance and human behavior can, in the words of Charles Perrow, be tightly coupled (Perrow, 2000). Thus, HRA requires the analyst to consider the bidirectional interaction of hardware behavior and operator response.

(Kolaczkowski et al., 2005) lists a number of characteristics that need to be considered in modeling human actions in HRA. These are:

- Plant behavior and conditions
- Timing of events and the occurrence of human action cues
- Parameter indications used by the operators and changes in those parameters as the scenario proceeds
- Time available and locations necessary to implement the human actions
- Equipment available for use by the operators based on the sequence
- Environmental conditions under which the decision to act must be made and the actual response must be performed
- Degree of training, guidance, and procedure applicability.
Assessing these characteristics requires an integrated HRA team. According to (Kolaczkowski et al., 2005), the HRA team should include the following:

- Risk analyst
- HRA specialist (i.e., someone experienced in HRA)
- Human factors specialist
- Thermal-hydraulic analyst
- Operations, training, and maintenance personnel
- Other specialists as necessary (e.g., structural engineer)

Each discipline specialist is envisioned as providing a portion of the context knowledge needed to adequately address human-system interactions. The HRA team needs to perform a number of activities in order to glean the necessary insights regarding the behavior of the human-machine system being analyzed. From (Kolaczkowski et al., 2005), these include:

- Walkdowns and field observations of areas where decisions and actions are to take place in order to understand the equipment involved, including the need for any special tools; the plant layout, including review of such issues as equipment accessibility, use (or not) of mimic boards, instrumentation availability, labeling conditions, etc.; whether any special fitness needs are required; the time required to reach the necessary locations and perform the desired actions; and the environment in which the actions will need to be performed (e.g., nominal, radiation-sensitive, high-temperature, etc.)
- Talk-throughs of scenarios and actions of interest with plant operators, trainers, or maintenance staff. Such talk-throughs should include a review of procedures and instructions to learn about the potential strengths and weaknesses in the training and procedures relevant to the actions of interest, identifying possible workload or time pressures or other high-stress issues, identifying potential complexities that could make the desired actions more difficult, and learning of any training biases that may be important to the actions of interest. In addition, in identifying and searching for errors of commission (i.e., operators perform an undesirable action that brings the system closer to or causes it to transgress a safety limit), it is important to obtain a good understanding of operators’ intentions in a given scenario. Clearly, inappropriately developed intentions could lead crews to take undesirable actions (e.g., terminate operation of an automatic safety system). Talk-throughs with crews and trainers provide an opportunity to obtain an understanding of operators’ expected intentions in given scenarios.
- Simulator exercises as a means to observe crew activities in an environment that is as nearly realistic as possible [emphasis added]. While it is realized that simulator exercises may not always be possible, it is good practice that at least a representative set of scenarios for the issue under investigation be simulated and observed by the HRA team. In addition to allowing analysts to obtain scenario-specific and related timing information relevant to implementing salient procedure steps, simulator exercises allow the analysts to observe how plant crews perform as a team and how they implement their procedures. These observations could lead to identification of important crew characteristics, such as clarity of communications (e.g., whether direction and feedback are clear or potentially ambiguous), the degree of independence that is allowed among individual crew members (e.g., what actions can be performed without general crew knowledge and the extent to
which review occurs to ensure that the appropriate actions were taken), the level of aggressiveness of the crew (e.g., whether some actions can be and are typically implemented out-of-sequence of the anticipated step-by-step procedural flow), etc. Moreover, observation of simulator exercises also provides a basis for discussions with operators and trainers about both the scenarios that are observed and those that cannot be observed due to time or resource constraints.

Because of the focus on low-frequency high-consequence scenarios, for which actual data are lacking, simulator exercises are generally of considerable importance to the fidelity of the HRA. Without detailed observations “in the wild,” the HRA can be seriously in error. A principal reason for this is that most complex technological systems under the control of human operators have procedures that guide both normal operation and response to upset conditions. These procedures constitute what (Guenther C. W., 2009) refers to as a loosely controlled process. Because such processes are not strictly controlled, deviations during upset conditions can be expected to be frequent, and possibly severe, especially when instrument failure or unavailability masks the actual accident scenario. A main thesis of this dissertation is that the important human failures that lead to undesired outcomes are not so much random in nature as they are caused by unanticipated process variations that lead to inappropriate operator actions in context. Conversely, it may be that such unanticipated (by the analyst) process variations can enhance the likelihood of successful operator termination of a developing accident sequence; the focus of HRA should also be upon identifying the positive aspects of operator performance. Observations of accident scenarios in a simulator can prove invaluable in identifying possible deviations in the underlying process established by the operating procedures, especially for scenarios with failed or unavailable instrumentation (so-called masked scenarios). Without such observations, the resultant HRA will likely model the procedural process under the often erroneous assumption that it is the one actually followed by the operators in their response to an upset condition.

The second research question will be explored in the following sections, which describe two illustrative full-scope HRA methods.

**Research question**

*How do current HRA methods approach the issue of simulator observations and are these approaches suitable for incorporating simulator observations into the HRA task analysis?*

**2.2 Illustrative HRA methods**

This section briefly describes two representative HRA methods from the perspective of the overall HRA framework described above. As noted by (Forester et al., 2006), most HRA methods do not address all three elements of the HRA framework (i.e., HFE identification, modeling, and quantification) described in (Kolaczkowski et al., 2005); in fact many (if not most) address only quantification, the final step in the process. The two representative HRA methods described briefly in this section are examples of the few that do address, to some degree, all three elements of the overarching framework described in (Kolaczkowski et al., 2005). One of these, the Technique for Human Error Rate Prediction (THERP) (Swain & Guttman, 1983), is a so-called first-generation method, meaning that its emphasis is more on operator behavior in response to stimuli from the system with which the operator is interacting than on operator cognition. As noted by (Forester et al., 2006), THERP has been one of the most widely applied
HRA methods, with applications to a variety of human-machine systems in the nuclear, process chemical, aviation, and medical industries. The second representative method is A Technique for Human Event Analysis (ATHEANA), described in (U. S. Nuclear Regulatory Commission, 2000) and (Forester et al., 2007). ATHEANA, with its increased emphasis on context and operator cognition, is a so-called second-generation HRA method. Unlike THERP, but like other second-generation methods, it has to date seen relatively few applications.

2.2.1 Technique for Human Error Rate Prediction (THERP)

As described in (Swain & Guttman, 1983), THERP is a method for identifying, modeling, and quantifying HFEs in a risk analysis. At some 700 pages, (Swain & Guttman, 1983) provides a comprehensive source of HRA knowledge, at least in the context of nuclear power plant risk analysis, although it has been applied to systems outside the nuclear domain, such as process chemical and medical systems. With its primary focus being rule-based operator behavior in response to system stimuli, THERP is a first-generation HRA method.

(Swain & Guttman, 1983) states that the information required to perform an HRA is best obtained via interviews with plant personnel, demonstrations of procedures, and simulation of abnormal events (emphasis added). Chapter 4 of (Swain & Guttman, 1983), which is devoted to analysis of human-system interactions in today’s terminology, provides a 10-step process for understanding the human-system interfaces for various operator activities. Step 5 of that process is part of the task analysis, which is a necessary and characteristic underpinning of the THERP approach to HRA. Besides information regarding the types of documents to review, THERP highlights the need to include talk-throughs or walk-throughs (including observations of actual tasks and discussions of abnormal events) so that the HRA analyst can become familiar with the operators’ activities.

THERP uses task analysis to decompose human actions into constituent subtasks, and this task decomposition is a distinguishing characteristic of the method. The decomposition is represented classically in an HRA event tree\(^3\), and is usually representative of procedures that govern the actions being decomposed. Thus, in order for the decomposition to be a faithful representation of the behavioral process followed by the operators, it is crucial that observations of operator performance be incorporated into the task analysis. This task analysis can be extremely resource intensive. Difficulties in performing the THERP task analysis in a consistent and reproducible manner led over time to approaches such as the Task-Analysis Linked Evaluation Technique (TALENT), described in (Well et al., 1990) and (Ryan, 1988). TALENT was employed for the THERP analyses described in (Kelly et al., 1992a), (Kelly et al., 1992b), and (Kelly et al., 1993). However, with respect to the crucial issue of simulator observations, TALENT does not provide detailed guidance for how information from these observations should be incorporated into the HRA task analysis. For the analyses described in (Kelly et al., 1992a) and (Kelly et al., 1992b), a detailed data collection form was developed to aid in this process. This form, shown below,

\(^2\) THERP does provide limited treatment of diagnosis via a time-reliability correlation developed using the judgment of the THERP authors.

\(^3\) Most THERP applications today do not use the HRA event trees advocated by the authors of THERP.
served as a template to guide the collection of information to support the THERP decomposition of HFEs into subtasks. A representative HRA event tree illustrating this decomposition, taken from (Kelly et al., 1992a), is shown in Figure 3.

Figure 1  Simulator data collection form, first page, from (Kelly et al., 1992a)
Local or Remote operation? Explain: ________________________________

Type of clothing during action: __________________________

Tasks or subtasks done step-by-step_____ or Dynamic_____.

Dependence: Is the order of the tasks critical? Yes or No

Does the success/failure of one action affect the success/failure of the next

Yes or No Explain: ________________________________

If 2 men do the job, does the action of either one affect the success/failure

of the next? Yes or No Explain: ________________________________

Is the job done with rest stops_____ or continuous performance_____?

Is there any radiation safety or caution for this job? Yes or No

If yes, what dosage?___________ mrem

HF comments of plant-specific PSFs: ________________________________

________________________________________

________________________________________

Additional Comments/observations: ________________________________

________________________________________

________________________________________

Figure 2 Simulator data collection form, page 2, from (Kelly et al., 1992a)
2.2.2 A Technique for Human Event Analysis (ATHEANA)

As described in (U. S. Nuclear Regulatory Commission, 2000), ATHEANA was developed with a goal of improving the state of the art in HRA, especially with respect to how realistically HRA can represent the kinds of human behaviors seen in accidents and near-miss events in complex technological systems. As such, ATHEANA was intended to incorporate the latest understanding by the psychology community of why human errors occur, and this understanding is substantiated by reviews of a number of significant system accidents, both nuclear and non-nuclear. The underlying premise of ATHEANA (and its approach to HRA) is that significant human failures occur because of a combination of influences, plant conditions, and associated human-related factors (taken altogether to be the “context” associated with the human action of interest). This combination that comprises the context triggers error mechanisms in plant operators, especially when these influences provide a context that is quite different from the operators’ experiences and knowledge base. As a result, much of the ATHEANA guidance is focused on identifying these combinations of influences, specifically referred to as “error-forcing contexts” (EFCs) by ATHEANA, and the assessment of their influence and likelihood. Consequently, one of the principal characteristics of the ATHEANA approach to HRA is a formal, systematic search.
scheme for describing context and identifying EFCs. In this regard, ATHEANA’s emphasis on understanding the context and its causal relationship to human performance makes it among the most comprehensive of HRA methods (the comprehensive, and therefore resource-intensive nature of ATHEANA has probably also been a reason for the dearth of applications of the method). In the ATHEANA approach, context is not so much “fitted” into a pre-established set of performance-shaping factors (PSFs) (e.g., level of stress, degree of complexity of task), as is done by many HRA methods, but instead context is allowed to develop into whatever characteristics are needed to identify the more significant aspects that will likely drive human performance for the situation at hand. This approach is intended to identify and address the important influences for the “nominal” case in risk models, as well as the influences associated with more unusual situations that may have a strong EFC.

With respect to simulator observations specifically, in ATHEANA these form a significant part of the process for developing and describing the context, with the goal of the observations being to ensure that context is developed accurately and with plant-specific influences accounted for. In particular, ATHEANA stresses the need to review observations of plant staff in simulated scenarios to learn about (1) crew dynamics (e.g., communication and interaction characteristics) and interaction protocols, such as the extent to which independent operator actions are typically allowed and performed, (2) crew strategies for implementing procedures, and (3) potential variations in control room response due to variability among crews. In the terminology of (Guenther C. W., 2009), which will be used throughout this dissertation, ATHEANA attempts to identify the underlying nominal process followed by the operators, as well as significant deviations from this nominal process, as it is these deviations that lead to unsafe acts by the operators, and ultimately to undesired system states. Note that, from the perspective of using HRA to improve the system, observation of such deviations presents an opportunity to make the system more resilient by, for example, modifying procedures associated with the observed deviations from the desired process.

In contrast to THERP, ATHEANA does not decompose human actions into subtasks. Instead it focuses on searching for error mechanisms that could lead to “unsafe acts,” which could in turn lead to a human error. This is illustrated in Figure 4, taken from (U. S. Nuclear Regulatory Commission, 2000).
2.3 Role of task analysis in HRA

As mentioned in Ch. 1, an important part of any comprehensive HRA is a task analysis. Task analysis is the name given to a wide range of techniques that can be used to examine the ways in which humans undertake particular tasks. Some of these techniques focus directly upon task performance, while others consider how specific features of the task, such as the interfaces, operating procedures, team organization or training, can influence task performance. This variability in the focus of task analysis is reflected in HRA methods that rely heavily on task analysis. For example, THERP is focused on task performance, while ATHEANA is more focused on the context in which the action is being performed. As described in (Swain & Guttman, 1983) and (Kolaczkowski et al., 2005), an important ingredient of the task analysis, however it is performed, and whatever the focus of the HRA method for which it is an input, is observations from system simulators. Because human-system interactions are typically governed by a loosely-defined process, with the process definition provided by the operating procedures and training, it is imperative for HRA credibility that the task analysis be able to “map out” both the nominal process followed by the operators and significant deviations from this process.

2.3.1 Representative examples of HRA Task Analysis

This section presents two representative examples of HRA task analysis. These examples will be examined in the next section from the perspective of where and how simulator observations have a role to play.
2.3.1.1 THERP

The first task analysis example is from (Kelly et al., 1992a), and was performed in support of an HRA that employed the THERP method. The general task analysis framework for this HRA followed the TALENT guidance described in (Well et al., 1990) and (Ryan, 1988). As discussed above, THERP is a decompositional approach, in which HFEs are broken down into constituent subtasks (i.e., a fine-grained model of the operational process is developed). In this example, the decomposition started from a functional description of key steps in the process (e.g., operator opens a certain valve). By breaking down the process (i.e., the human actions) into specific tasks and subtasks associated with individual procedures and equipment, the analysts began to identify specific modes, causes, and effects of failure. The description of each task or subtask was enhanced by referencing specific PSFs that affected a given task. The PSFs, which can either enhance or degrade operator performance, were derived from direct observations of operator performance in the plant and in the simulator, time line analyses, and evaluation of the human-system interface by members of the analysis team who were human factors experts. Examples of PSFs included the following (an asterisk is placed next to those for which dynamic simulator observations were vital):

- Ergonomic quality of the human-system interface
- Written procedures and their use*
- Communications*
- Nature of operator action (e.g., skill-, rule-, or knowledge-based)
- Training and experience
- Stress*
- Task dependence*

The detailed data collection form shown in Figure 1 and Figure 2 served as a template to guide collection of requisite information for the task analysis, in sufficient detail, for each task and subtask that was modeled in the HRA. The output of this effort was an extensive list of operator tasks and subtasks, with associated PSFs, for each HFE in the PRA sequences under consideration. This list was the input for the next step, HRA modeling within THERP. HRA event trees, similar to the example shown in Figure 3, were used to represent the task decomposition, and formed the framework for estimating the overall HEP. The HRA event tree in Figure 3 represents operator failure to close residual heat removal suction isolation valves during plant startup, thus exposing systems designed for low pressure to relatively high pressures from the reactor coolant system, which can in turn lead to rupture of the interfacing low-pressure system.

Following the standard THERP convention, error paths are placed along the descending right-hand branches of the tree, and represent variations in the nominal operational process, in the framework of this dissertation. Successful actions are placed along the left-hand side of the tree, and along with the ordering of the events, represent the expected or nominal procedural flow. For example, on the top left, event “a” is a success path representing entry by the reactor operator into procedure OP/1/A/6200/04, at Step 2.26 of Enclosure 4.1. Failure to enter this procedure is shown on the right-hand branch as event “A” (this is an error of omission in the THERP taxonomy of human error). When a second operator is involved, such as in events “F” and “H,” the actions are generally shown in succession along a branch of the tree. For example, in task
“E” the reactor operator has failed to close valve ND-1B or ND-2A. The Control Room Supervisor, who is in the control room with the reactor operator, has an opportunity to detect this error and correct it. This so-called recovery action by the Control Room Supervisor is represented by event “f,” and is shown as a dotted line on the HRA event tree.

In this fashion, individual error paths are identified and failure probabilities are estimated using the representative HEPs in the tables provided in Ch. 20 of (Swain & Guttman, 1983). For example, path “A” in Figure 3 represents failure by the reactor operator to enter procedure OP/6200/04. This was assessed to be best represented by Item 2 in Table 20-7 in (Swain & Guttman, 1983), and was not adjusted by a PSF multiplier, on the basis of observations made by the HRA team during their two-week visit to the plant. Each HRA event tree generally models several error paths. For example, events “A” and “B” together constitute an error path whereby the reactor operator fails to enter procedure OP/1/A/6200/04 on two occasions: Step 2.26 of OP/1/A/6100/01 and later, in Step 2.34 of the same procedure. In a similar manner, failure path “A-b-C” represents a sequence in which the reactor operator fails to enter procedure OP/1/A/6100/01 at Step 2.26, but recovers from this initial error (“b”), only to then fail at selecting the correct procedure enclosure (“C”). Probabilities for each unique error path are estimated by multiplying each HEP on a given path by other (conditional) HEPs on the same path. For example, the probability for path “A-B” is estimated by multiplying the HEP for failure “A” (0.003) by that for failure “B” (0.003). Other error paths for this HRA event tree include “A-b-c-D,” “a-c-d-e-G-H,” and “A-b-c-d-E-F.” The individual error paths probabilities are then summed to give the total error probability for the HRA event tree, and thus for the HFE modeled by the event tree.

The THERP task analysis must identify critical tasks and subtasks for each HFE in the PRA. These can vary, as a particular HFE can appear in several different accident scenarios in the PRA. For example, HFE ND-OPEN can occur during plant startup, where it represents failure to close at least one residual heat removal suction isolation valve prior to reactor coolant pressure exceeding a certain limit. It can also occur during shutdown sequences, and there it represents opening of the same isolation valves with reactor coolant system pressure too high. In general, the task analysis includes critical procedural steps and transitions from one procedure to another, omission of procedural steps, and simple commission errors (slips, whereby the wrong switch is selected, for example). Recovery paths must also be identified.

2.3.1.2 ATHEANA

As noted earlier, ATHEANA has been only seldom applied, and the applications that do exist have been small-scale ones, which have not been published. However, many of its precepts are being carried forward into the new hybrid HRA method described in Chapter 5. Therefore, it is worthwhile reviewing an example analysis to see how ATHEANA differs from THERP, and what additional challenges it presents for HRA task analysis. The example shown here is from an unpublished analysis in which the author took part during 2004-5, as part of a risk analysis of post-core damage steam generator tube rupture sequences (Kunsman, et al., 2005).

4 THERP treats dependence among subtasks in its quantification procedure, but this is not important here, so a discussion of dependence has been omitted.
The overall PRA study, of which the HRA described here was an integral part, concentrated on so-called station blackout (SBO) sequences, wherein all offsite and onsite AC electrical power to the plant is unavailable (the initiator for such sequences is generally a failure of offsite AC power, perhaps due to a fault in the electrical grid). This focus was chosen for this analysis because SBO scenarios present the situations that most threaten the integrity of the steam generator tubes after the onset of core melt: a high reactor coolant system (i.e., primary) pressure coupled with a dry steam generator (i.e., secondary), which together provide the pressure and thermal gradients necessary to fail U-tubes in a steam generator.

The HFEs addressed in the HRA were limited to operating crew actions associated with SBO scenarios in which there is a failure of the turbine-driven auxiliary feedwater (TDAFW) system, which would normally ensure that energy is transferred from the primary to the secondary coolant system during SBO, preventing core melt. With TDAFW failed, the reactor coolant system remains at high pressure during core melt. If the integrity of the secondary coolant system is lost subsequent to the onset of core melt, for whatever reason, and steam egresses from the secondary system, then the secondary sides of the steam generators become dry because TDAFW is unavailable to provide make-up water, and secondary-side pressure drops, producing large pressure gradients across the steam generator tubes, increasing the likelihood of tube failure.

The PRA examined system and component failures that could put the reactor system into this state. The PRA and HRA analysts, working together and using results from thermal-hydraulic analyses, identified operating crew actions that could mitigate the accident progression. There are many variations of system events and operator actions that can affect the timing and progression of such accident scenarios. These include such things as the availability of the TDAFW pump, demands on pressure relief valves, operator shedding of DC loads to extend battery life, operator depressurization of the secondary and/or primary system per emergency procedures and severe accident management guidelines, secondary system leakage, and AC power recovery.

A streamlined version of the ATHEANA HRA process (U. S. Nuclear Regulatory Commission, 2000) was the basic approach used to perform the HRA. A streamlined approach was adopted for several reasons. First, even though a particular plant and its PRA were chosen to serve as the “generic” plant for purposes of the analysis, resource limitations prevented a plant visit by the HRA team. Thus, no simulator exercises could be observed and no questions could be asked directly of plant operators and trainers. In addition, even though the Westinghouse Owners Group Emergency Response Guidelines (ERGs) and Severe Accident Management Guidelines (SAMGs) were available for the analysis, plant-specific procedures that implement these guidelines were not. Finally, it was decided that questions regarding plant-specific PSFs such as operating crew training and biases, crew understanding of procedures and their usage, crew dynamics and characteristics, key instrumentation and cues for the scenario from the crews perspective, expected workload, human-system interface characteristics, and “informal rules” that might influence their decisions, could not be submitted to the plant, again because of resource limitations. Thus, much of the plant-specific information required by the ATHEANA HRA method (and other methods for that matter) to perform a realistic analysis could not be obtained.

Because of this lack of information, the HRA team made many assumptions (rather than searches for information as is prescribed by the ATHEANA process) regarding the scenario conditions and
how operating crews might respond. However, as noted above, the Westinghouse Owners Group
ERGs and SAMGs were available and the PRA/HRA team included several individuals with
many years experience in performing PRA and HRA, both engineers and psychologists. These
individuals were very familiar with plant control rooms and plant operations and in addition, a
former Senior Reactor Operator was available to support the team.

Thus, assumptions had to be made about the factors and conditions likely to influence operating
crew behavior in the scenarios being examined. In this respect, the HRA team made attempts to
at least “hypothetically” obtain the information that would be needed for an ATHEANA analysis
and use it during the quantification process. It should be realized that without plant-specific
information, the likelihood is much higher than usual for critical incorrect assumptions to be
made that could significantly alter the conclusions with respect to the probability of the modeled
HFEs. It is also likely that the problem is compounded by the absence of simulator observations,
and thus that very little is known about how crews will respond under severe accident conditions.

Since plant-specific information could not be obtained, an early step in the analysis was to study
and understand the Westinghouse ERGs and SAMGs relevant to the scenarios of interest, as they
are expected to govern to a large degree the process by which the operators interact with the
reactor systems during an accident scenario. To facilitate this, flow charts of the relevant
guidelines were developed (per the ATHEANA procedure) and critical decision points were
identified. See Figure 5 for an example. This is a salient difference between ATHEANA and
decompositional approaches such as THERP. The ATHEANA representation of the process is at
a higher level than that of THERP; details of the process have been abstracted away. The flow
charts, in conjunction with the PRA modeling of the accident scenarios, were used to identify
human actions with the potential to be important (including errors of both omission and
commission). The ATHEANA flow charts, in conjunction with the results of the thermal-
hydraulic analyses (including scenario-specific timing information and plots of the expected
behavior of critical parameters), were also used to identify, to the extent possible, aspects of the
scenario context (including PSFs and plant conditions, where the plant conditions were taken
from the output of the thermal-hydraulic analyses) that would be faced by the operating crews in
responding to the accident scenarios and that could influence the probability of the HFEs. In
other words, a search was made for process variations that could lead to unsafe operator actions.
The HRA quantification approach used for ATHEANA is based on expert judgment rather than the scenario-matching approach taken by THERP. It comprises the following six steps and relies on a facilitator-led elicitation of expert opinion, as described in (Forester et al., 2007).

**Step 1: Describe the HFE and associated context**

The purpose of Step 1 is to:

- Collect (or make assumptions about) any additional information that is not already collected and that is needed to describe and define the HFES (and associated contexts),
- Review all information for clarity, completeness, etc., and
- Interpret and prioritize all information with respect to relevance, credibility, and significance.

Table 1 provides examples of information that would normally be identified using the ATHEANA method and that would serve as inputs to HFE modeling and quantification, whether collected during the HFE and context identification steps or as part of Step 1 of HFE quantification.
Table 1  Examples of information useful to HFE quantification with ATHEANA

<table>
<thead>
<tr>
<th>Information Type</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plant conditions &amp; behavior for base case scenarios</td>
<td>Thermal-hydraulic conditions as a function of time, expected plant indications as a function of time, system/equipment operations, expected operator actions.</td>
</tr>
<tr>
<td>Plant conditions &amp; behavior for possible deviations of the scenarios</td>
<td>Expansion of the above including reasonably possible unusual plant behavior and failures of systems, equipment, and indications, especially those that may be unexpected or difficult to diagnose by operators.</td>
</tr>
<tr>
<td>Critical plant functions for accident mitigation</td>
<td>Specific equipment operation, requirements for operator action, possible operator recovery actions for failed systems/equipment.</td>
</tr>
<tr>
<td>Operating crew characteristics (i.e., crew characterization)</td>
<td>Crew structure, communication style, emphasis on crew discussion of “big picture”, behaviors observed in simulator exercises and/or identified by training staff.</td>
</tr>
<tr>
<td>Features of procedures</td>
<td>Structure, how implemented by operating crews, opportunities for “big picture” assessment and monitoring of critical safety functions, emphasis on relevant issue, priorities, any potential mismatches with deviation scenarios.</td>
</tr>
<tr>
<td>Relevant informal rules</td>
<td>Experience, training, practice, ways of doing things - especially those that may conflict with informal rules or otherwise lead operators to take inappropriate actions.</td>
</tr>
<tr>
<td>Timing</td>
<td>Plant behavior and requirements for operator intervention versus expected timing of operator response in performing procedure steps, etc.; input from training staff and results of simulator exercises; based upon perceived needs of the PRA, multiple times or time frames may need to be considered for each HFE.</td>
</tr>
<tr>
<td>Relevant vulnerabilities</td>
<td>Any potential mismatches between the scenarios and expected operator response with respect to timing, formal and informal rules, biases from operator experience and training, etc.</td>
</tr>
</tbody>
</table>
Error mechanisms | Any that may be particularly relevant by plant context or implied by vulnerabilities; applicable mechanisms depend upon whether HFE is a slip or mistake. Examples include: failures of attention, possible tunnel vision, conflicts in priorities, biases, missing or misleading indications, complex situations, lack of technical knowledge, timing mismatches and delays, workload and human-machine interface concerns.

Performance shaping factors | Those deemed associated with or triggered by the relevant plant conditions and error mechanisms.

All of the above three items in Step 1, and especially the third item, were carried out as part of an open discussion among the members of the HRA team. The goal of this discussion was not to achieve consensus but to advance the collective understanding of the HRA team through the sharing of distributed knowledge and expertise. In each case, the scenario (or group of similar scenarios) and the HFE in question were described and the vulnerabilities and strong points associated with taking the right action were discussed openly among the team.

Step 2: Identify the driving factors of the scenario context

The purpose of Step 2 is to identify, from among the multitude of factors that can influence operator performance, those that actually drive behavior/performance for each HFE and its associated context. Each expert participating in the elicitation process individually identifies these factors based on the expert’s own judgment. These factors are those expected to be the main determinants of the final HEP for the HFE in question. Typically, there will be multiple such factors. This is due to the focus of the ATHEANA search process on combinations of factors that are more likely to result in an integrated context. When there is only a single driving factor, it is usually one that is so overwhelming that it alone can easily drive the estimated probability. For example, if the time available is shorter than the time needed to perform the actions associated with the HFE, quantification becomes much simpler and does not need to consider factors beyond time available.

Step 3: Generalize the context by matching it with generic, contextually-anchored rankings or ratings

In Step 3, each expert participating in the elicitation process must answer the following question for each HFE: Based upon the factors identified in Step 2, how difficult or challenging is this context relative to the HFE being analyzed? Answering this question involves independent assessments by each expert. In order to perform this assessment, the specifics of the context defined for the HFE must be generalized or characterized. These characterizations or generalizations then must be matched to general categories of failures and associated failure probabilities.

To assist the analysts (who may not have strong backgrounds in probability) in making their judgments regarding the probabilities of events, some basic guidance is provided in (Forester et al., 2007). In thinking about what a particular probability for an unsafe act will be, analysts are encouraged to try to imagine how many times out of 10, 100, or 1,000 would they expect crews to
commit the HFE, given the identified context. The following examples of what different probabilities mean are provided to the analysts:

- "Likely" to fail ~ 0.5 (5 out of 10 would fail)
- "Infrequently" fails ~ 0.1 (1 out of 10 would fail)
- "Unlikely" to fail ~ 0.01 (1 out of 100 would fail)
- "Extremely unlikely" to fail ~ 0.001 (1 out of 1000 would fail)

The analysts are allowed to select any values to represent the probability of the unsafe act. That is, values other than those listed above can be used. However, the analyst must provide numeric probabilities. The qualitative descriptions above are provided initially to give analysts a simple notion of what a particular probability value means.

**Step 4: Discuss and justify the matches made in Step 3**

In Step 4, each expert was asked to independently provide his estimate for each HFE. Once all the expert estimates were recorded, each expert was asked to describe the reasons why he chose a particular failure probability. In describing his reasons, each expert should identify what factors (positive and negative) were thought to be key in characterizing the context and how this characterization fit the failure category description.

After the original elicited estimates were provided, a discussion was then held that addressed not only the individual expert estimates but also differences and similarities among the context characterization, key factors, and failure probability assignments made by all of the experts. This discussion allowed the identification of any differences in the technical understanding or interpretation of the HFE versus differences in judgment regarding the assignment of failure probabilities. Examples of factors important to HFE quantification that might be revealed in the discussion include:

- Differences in key factors and their significance, relevance, etc. based upon expert-specific expertise and perspective
- Differences in interpretations of context descriptions
- Simplifications made in defining the context
- Ambiguities and uncertainties in context definitions.

**Step 5: Refinement of HFEs, associated contexts, and assigned HEPs (if needed)**

Based upon the discussion in Step 4, the experts formed a consensus on whether or not the HFE definition must be refined or modified based upon its associated context. If the HFE must be refined or re-defined, this is done in Step 5. If such modifications were necessary, the experts "re-estimated" the HEP, based upon the newly defined context for the HFE (or for new HFEs, each with an associated context).

**Step 6: Determine final HEP for HFE and associated context**

The final probability estimate (from the initial quantification process) that will be incorporated into the PRA for each HFE is determined in Step 6.
2.4 Need to incorporate data from simulator studies into HRA task analysis

As described in (Swain & Guttman, 1983) and (Kolaczkowski et al., 2005), an important ingredient of the HRA task analysis, however it is performed, is observations from system simulators. These observations are important in order for the HRA team to be able to realistically model procedure implementation, interactions between the crew and the system, and interactions among the crew members themselves. Without such observations, the HRA is likely to deviate significantly from reality. Failure to carry out such observations has been cited as a weakness in applications of some of the major HRA methods, such as THERP (Forester et al., 2006). Simulator studies are also a major source of information for some of the newer second-generation HRA methods, and guidelines promulgated by industry emphasize the important of gathering simulator data (Wakefield et al., 1992).

(U. S. Nuclear Regulatory Commission, 2000) identified the following roles for simulator observations in support of analysis with ATHEANA:

- Focused opportunity to discuss with teams of operators and other training staff the important characteristics of the accident sequence context.
- Opportunity to observe the styles of teamwork and problem-solving and general operating strategies for operating crews.
- Ability to test the extent to which the context appears to be “error-forcing,” either as simulated or with additional elements as discussed with operators and trainers.
- Opportunity to evaluate the potential failure probability of the crew in the context of the scenario as modelled.

The example ATHEANA application described above suffered from a complete lack of simulator observations. Such observations would have been invaluable for many of the inputs listed in Table 1, and many assumptions had to be made by the HRA team to compensate for their absence.

In contrast with the ATHEANA application, simulator observations were performed for the illustrative THERP analysis, and provided valuable insights. Some examples of these insights are listed below.

- From a procedural standpoint, there was only one alarm in the control room that would warn the operators of a breach in the low-pressure interfacing system, and if this alarm did not function, operators would be delayed in detecting and responding to the breach.
- Procedures did not attribute high pressures in the interfacing systems, or high levels in tanks that receive flow from interfacing system pressure relief valves, to a potential ISLOCA event. Some procedures attributed causes of high tank level to misleading sources.
- Operator diagnostic activities focused on the presence of an auxiliary building ventilation high radiation alarm, which has the deleterious effect of increasing operator stress due to its initiation of preparation for a site evacuation, and the sealing and positive pressurization of the control room.
- Feedback as to the credibility of recovery actions by members of the control room crew.
The Ispra HRA benchmark study (Commission of the European Communities, 1989) also recognized the importance of simulator observations for task analysis. However, because of resource limitations, simulator observations by the participating analysis teams were not possible. To partially compensate, the analysis teams were provided with a video showing the operator teams performing the tasks that were to be analyzed.

The recently completed international HRA empirical study (U. S. Nuclear Regulatory Commission, 2010) has also highlighted the importance of simulator observations to the HRA task analysis. One of the participants in the study has even extended the role of simulator observations beyond the task analysis, noting that “simulator studies provide rich qualitative and quantitative data sources, and their usage would lend credibility to HRA overall, particularly for existing plant[s]” (Corporate Risk Associates, Ltd., 2009). A similar conclusion was put forth by the Organisation for Economic Co-operation and Development (OECD) in a 2008 report on recommended actions to support the collection and exchange of HRA data (Committee on the Safety of Nuclear Installations, 2008). This report identified data collection in nuclear power plant training and research simulators as a priority for future activities by the Nuclear Energy Agency.

### 2.5 Approaches for linking simulator data to HRA task analysis

A principal difficulty with HRA, and one that hampers the use of HRA results in risk-informed decision-making, is the large variability associated with the analysis results, from one method to another, and from one analyst to another for a given method. This was a difficulty first highlighted by the Ispra HRA Benchmark study of 1989, in which four orders of magnitude were observed among the estimates of human error probability developed by teams analyzing a common benchmark problem (Commission of the European Communities, 1989). The more recent international HRA empirical study, sponsored by the U. S. Nuclear Regulatory Commission has found that, twenty years after the Ispra study was published, disturbingly large variability appears to remain (U. S. Nuclear Regulatory Commission, 2010).

As pointed out above, it has long been recognized that simulator observations are a necessary ingredient for a credible HRA. However, as pointed out by (Forester et al., 2006), HRA guidelines do not provide guidance on how analysts could benefit from the wealth of information that can be obtained from observing simulator exercises to support understanding of crew characteristics and behavior, and other general plant-specific factors that could influence performance in particular scenarios.

Although a video was provided to the analysis teams participating in the Ispra benchmark, lack of tools and procedures for incorporating this information into the task analysis hampered its use, and the resulting variability in the task analysis was judged to be a contributor to the large variability in the human error probabilities produced by the analysis teams participating in the study (Commission of the European Communities, 1989).

Shortly after the publication of the Ispra HRA benchmark study, a large-scale HRA effort was undertaken as part of the risk analysis performed to better understand the contribution of intersystem loss-of-coolant accidents (ISLOCA) to the risk of U. S. nuclear plant operation (Galyean W. J., et al., 1991), (Kelly et al., 1992a), (Kelly et al., 1992b), (Galyean W. G., Kelly, Schroeder, & Ellison, 1993), (Galyean W. J., et al., 1994). As discussed above for the
representative application of THERP, simulator observations were made as part of a two-week visit by the analysis team to each plant being analyzed. The detailed data collection form illustrated earlier was developed, and this form was filled in manually by the analysts observing the simulator exercises. Needless to say, this approach is both labor-intensive and inefficient. Again, tools were lacking to aid in incorporating the simulator observations into the HRA task analysis.

2.6 Summary of where current methods fall short in supporting HRA task analysis

As is evident from the above, a principal difficulty with simulator studies is how to make efficient use of the wealth of information that is generated by a typical study. The most popular HRA approach to date, namely THERP, emphasizes the importance of incorporating simulator observations into the HRA task analysis. However, the THERP documentation offers no guidance as to how this is to be done. The TALENT approach to task analysis of (Well et al., 1990) and (Ryan, 1988) was developed to “fill in the gaps” in THERP, and thus make the task analysis more consistent and reproducible. However, although TALENT provides voluminous guidance on what information to collect, it does not provide detailed guidance for how information from simulator observations should be incorporated into the HRA task analysis.

With respect to second-generation HRA methods, represented herein by ATHEANA (U. S. Nuclear Regulatory Commission, 2000), (Forester et al., 2007), although these approaches are more holistic than first-generation methods such as THERP, and attempt to take a broader process view, as opposed to the decompositional, “local” perspective of THERP, they are still lacking in explicit guidance and tools to support the HRA task analysis.

Thus, existing HRA methods lack the ability to analyze large amounts of simulator data efficiently and cannot translate these data into the information required for the task analysis. There are no tools currently used by the HRA community that can aid directly with incorporating simulator observations into the HRA task analysis. A desirable feature of such tools would be an ability to identify underlying models of crew performance, and to highlight deviations from expected performance, along with crew-to-crew variations.
3 Overview of process mining tools and applications

In Ch. 2, we saw that, while the incorporation of simulator observations into the HRA task analysis has long been acknowledged as important, there is a dearth of techniques and tools available to the HRA community for doing so. This leads to the next research question:

Research question

Are there tools in other domains that are more suitable and which, if adopted (and adapted to their new domain), could improve the state of the art in HRA modeling and task analysis?

Thus, we turn to other disciplines for tools and techniques that may be applied or adapted for this purpose. This chapter will examine a leading candidate, process mining. It begins with a general overview of process mining; later sections will make the tie from this general setting to the specific application of process mining to HRA, in particular to HRA task analysis.

3.1 Process mining overview

The business community frequently uses information systems to support business and industrial processes, such as workflow management, customer relations, etc. These process-aware information systems (PAIS) typically rely on automated logging of events in the underlying process, and thus generate large amounts of data for subsequent analysis. The field of business process mining (BPM) has developed in response to this need to efficiently and effectively analyze large amounts of automatically logged data (van der Aalst et al., 2004). As a field of research, it is concerned with analysis of business processes, based on information in event logs, often generated by a PAIS. Because the PAIS output can be very large, perhaps containing hundreds of millions of events (Guenther et al., 2008), process mining often focuses on developing an abstract analytical model with which an analyst can visualize various aspects of the underlying process.

As described in (Guenther, 2009), process mining approaches can be grouped into three main categories, or vectors: discovery, conformance, and extension. In the discovery category, one attempts to discern an underlying model that gives rise to the data contained in the event logs. The conformance category is concerned with discerning how closely an analytical process model conforms to the data in the event logs. Finally, in the extension category the analyst’s concern is with augmenting a pre-existing analytical model with event log data.

At the highest level, one can say that the goal of process mining is to extract information about processes from event logs, and the various approaches can be categorized as described above, based on whether one is attempting to discern an underlying process model (discovery), attempting to identify where the actual process deviates from the expected one (conformance), or augmenting an existing model with data, perhaps to make predictions about future exercises of the process via, for example, simulation (extension).

In addition to this categorical view of process mining, (Guenther, 2009) describes three process mining perspectives. The process perspective focuses on the flow (i.e., ordering) of activities in the process. The resulting models are sometimes said to be mined from the control-flow perspective. There is also the organizational perspective, which focuses on which individuals...
perform the various tasks that the process comprises, and how these individuals interact with one another during the process execution. Finally, there is the case perspective, which looks at variations in the process instances that make up the event log.

From the point of view of using simulator data to inform the HRA, particularly the task analysis, the simulator event logs comprise readings on process variables (e.g., pressure and temperature) taken at pre-defined time intervals during the simulation, alarms or other annunciators received during the simulation, and (most importantly) discrete actions taken by the operating crew in response to the evolving scenario (e.g., stopping a pump). Logs vary in format and content from facility to facility, but a typical excerpt is shown in Figure 6 for operator actions. It is often the case that an event log comprises multiple files, with operator actions in one file, process variables in another, etc. The file formats can be proprietary or open format, binary or ASCII, and this variety of formats, and even variety within a particular format, creates a significant challenge.

![Figure 6 Excerpt of operator action event log from a nuclear plant simulator](image)

In the log excerpt shown in Figure 6, activity is the action performed, the time at which the action is performed is provided, and crew corresponds to process instance; there will be one instance of the process recorded for each crew observed in the simulator. In some logging systems (but not in the excerpt shown in Figure 6), the operator who performs the action may be recorded, so there may be information on the originator of the action. The HRA is potentially interested in all three process mining categories described above. There is obviously an interest in the discovery category, and particularly the control flow perspective in this category, as one of the prime goals of the HRA is to discern the underlying process describing operator behavior. The case perspective, which examines variations in the process among the crews, may be important for identifying deviations that could lead to risk-significant human errors. The conformance category could be useful in comparing observed crew performance with the performance expected from an ideal crew under ideal circumstances. Information mined from this category could be used, for example, in providing feedback to an operator training program. Finally, the extension category
could be used in various ways. First, one might simulate the process to make predictions of human errors and estimates of their probabilities. Second, by augmenting the event logs with information from process variable logs, one could attempt to discover operator decision rules to explain variations observed from crew to crew in the case perspective.

Historically, much of process mining has been focused on the control-flow perspective within the discovery category, with the goal of constructing an analytical model that represents the process recorded in the event logs. The Petri net (Cassandras & Lafortune, 2008) has been the analytical model used most often for this representation. Petri nets were first developed by C. A. Petri in the 1960s, and have been employed as system models in diverse fields, with quite a body of analysis techniques developed for them. There are also extensions to the basic Petri net, such as so-called colored Petri net (Jensen & Kristensen, 2009), which can be used to simulate system behavior. Some details about Petri nets can be found in Appendix B, with more details available in (Cassandras & Lafortune, 2008) and many other references.

3.2 Review of process mining applications

To help decide if process mining holds potential for informing HRA task analysis, several past process mining applications were reviewed, focusing on applications with features in common with the simulator environment that is the target for HRA applications of mining. In such an environment, goal-driven tasks are performed by operators who have been highly trained on these tasks. Furthermore, the process is governed by written procedures. However, the process is not completely constrained by training and procedures, and thus there are opportunities for deviations from the intended process, along with variability in the times at which tasks may occur.

The following applications of process mining were reviewed.

1. Application of process mining to wafer testing for ASML, (Rozinat et al., 2009)
2. Application of process mining to healthcare in a large hospital setting, (Mans et al., 2008)
3. Application of process mining to usage logs of deployed X-ray machines, (Guenther C. W., 2009)
4. Application of process mining to software development, (Guenther C. W., 2009).

Of these applications, the first two bore the most resemblance to the intended application to simulator events logs, and so these are summarized below.

3.2.1 Process mining applied to wafer testing in ASML

This application is more fully described in (Rozinat et al., 2009). ASML is the world’s leading manufacturer of equipment for producing semiconductor chips. ASML designs, develops, integrates, and services advanced wafer scanner systems that produce these chips. The wafer scanners are complex devices that employ a photographic process to image nanometric circuit patterns onto a silicon wafer, much as a non-digital camera produces an image on photographic film. Time-to-market is very important, and so there is an ongoing effort to reduce the line widths on the silicon wafer in order to enhance the performance of the chips, with each new generation of wafer scanners advancing the boundary of what is technologically achievable. Therefore, testing of wafer scanners in the ASML factory is an important, but time-consuming process. Each wafer scanner is tested in the ASML factory and, if all tests are passed, it is
disassembled and shipped to the customer where it is reassembled and tested again. The joint
testing process by ASML and the customer typically consumes several weeks. Because of time-
to-market pressure, ASML actively works at reducing the length of the testing period.

To assist ASML in their efforts to reduce the length of the testing period, and thereby shorten
time-to-market, process mining was applied to the test process. The test process consists of three
phases: 1) calibration, 2) actual testing, and 3) final qualification. Because of continuous
enhancements being made to wafer scanners, the number of manufactured wafer scanners of a
single type is small, typically less than 50. Each new scanner type dictates changes to parts of the
 calibration and actual testing phases.

Sets of calibration and test actions are grouped into job steps, which are executed according to a
specified sequence. The sequence is relatively invariant, although large changes in system design
can result in changes to the job step sequence. Execution of tests can produce cases in which
tests are failed, and this can in turn result in a lengthy re-test of parts of the sequence. ASML’s
goal was to minimize the waiting time for a hardware fix and reduce the re-execution of parts of
the job step sequence. This goal could be met by testing all components thoroughly before and
during the assembly process, but this would increase the total test duration and would thereby
increase time to market. Therefore, the main goal was to reduce the duration of the test process
by applying process mining techniques to the historical test process data to identify bottlenecks
and other areas where the test process might be improved.

An example log excerpt is shown in Figure 7. Each line in the log corresponds to the execution
of one test. The number at the beginning of the line identifies the wafer scanner being tested.
The other entries are the test start and completion times, and a four-letter anonymized code that
identifies the test.

To analyze the data with the ProM software (van Dongen et al., 2005), the ASML log data first
had to be converted to the mining extensible markup language (.mxml) format used by ProM.
This was accomplished by programming a plug-in in Java for the ProMImport framework
(Guenther & van der Aalst, 2006). In the .mxml format used by ProM, a log comprises a set of
process instances or cases, with each case comprising audit trail entries that correspond to events
with various attributes, such as, timestamps, type of event, who performed the event, etc.

![image]

Figure 7  Excerpt from ASML event log (left) and corresponding .mxml format (right),
taken from (Rozinat et al., 2009)

Focusing on the process discovery aspects of this application, it is first worth noting that there
were relatively few process instances in the logs, but each process instance was quite long,
containing many audit trail entries describing the logged tests executed during different parts of
the process. The process is also very flexible, as parts of the process might be repeated.
depending on test outcomes, and this produces many variations in the logged test sequences. These are aspects that this application shares with simulator event logs. Each crew corresponds to a process instance (so there will be relatively few crews), and each simulation may last an hour or more and generate many thousands of audit trail entries. Portions of the procedures may be revisited by the operators during the course of the accident scenario.

(Rozinat et al., 2009) notes that, because of this flexibility, traditional process discovery algorithms are not likely to be helpful, as they assume that the underlying process to be discovered is highly structured, with a very limited number of possible process variations, each of which has a regular form. A heuristic net model (see Sec. 3.4.1) was mined for the ASML process. The initial model discovered by the heuristic miner was a large, very complex “spaghetti model,” reproduced in Figure 8.5

5 Here and elsewhere in this dissertation, certain figures will be shown which are too large for all details to be legible. In these cases, the figure is not intended to convey detail, but instead some other characteristic, typically complexity, as is the case here.
As noted by (Rozinat et al., 2009), such complex models are not the result of limitations in the heuristic mining algorithm, but instead reflect the inherent complexity of the process being mined. (Rozinat et al., 2009) had to apply additional filtering techniques to reduce the complexity of the mined heuristic net. This filtering eventually allowed the somewhat simpler heuristic net shown in Figure 9 to be mined. It is anticipated that simulator event logs will produce similar results when these mining techniques are applied, particularly if events are logged at a low level of detail.
Figure 9  Less complex heuristic net for the ASML testing process, obtained after additional event log filtering, taken from (Rozinat et al., 2009)

This less complex heuristic net provided some insights about the underlying process. First, it showed where dominant feedback loops existed, indicative of where job steps had failed, forcing a previous step to be redone. Once the previous job step is redone, other parts of the sequence often have to re-executed, and this takes time. Second, by identifying where the idle time tends to accumulate in the process, the model makes clear that a small number of tests are contributing to most of the process idle time, and these tests can become the foci for process improvement.

One of the conclusions drawn by (Rozinat et al., 2009) from this application that has particular relevance to the application of process mining to simulator data is that further research is needed to develop process mining techniques suitable for analyzing less structured complex processes, as the overly complex and confusing “spaghetti models” produced by the heuristic miner are of limited use.
3.2.2 Application of process mining in healthcare

This application is more fully described in (Mans et al., 2008). Healthcare in a hospital setting is characterized by quite complex and extremely flexible patient care processes, referred to as “careflows.” The hospital for this application was a large academic hospital in Amsterdam, and the event logs contained data on 627 gynecological oncology patients, for which all diagnostic and treatment activities had been logged for financial purposes. The hospital logs contained a huge number of distinct activities, often at a very low level of detail (e.g., determination of erythrocyte sedimentation rate for a patient). This is a characteristic that may be shared with simulator data logs, where each interaction of the operator with the simulator (e.g., acknowledging an alarm) may be logged. To avoid the overly detailed spaghetti model that would be produced if these low-level activities were retained in the mining, the logs were preprocessed to aggregate or remove these low-level activities.

Like (Rozinat et al., 2009), (Mans et al., 2008) employed the heuristic miner to discover the underlying careflow process. Unfortunately, this approach resulted in the overly complex spaghetti model reproduced in Figure 10. To cope with this complexity, (Mans et al., 2008) attempted to divide the patient care log into two or more sub-logs, which would then be analyzed separately. Clustering techniques (see Sec. 3.4.3 below) were applied to produce groups of process instances with similar properties. This produced nine clusters of more reasonable size. The heuristic net for the largest cluster is shown in Figure 11.

Figure 10  Heuristic net for hospital careflow process, mined after preprocessing of event logs, illustrating overly complex nature of resulting model, taken from (Mans et al., 2008)
Figure 11 Heuristic net for largest cluster in the hospital careflow process, taken from (Mans et al., 2008)

(Mans et al., 2008) also applied the fuzzy miner (see Sec. 3.4.2 below and Appendix C) to the hospital event logs. As described in Sec. 3.4.2, Appendix C, and (Guenther C. W., 2009), the fuzzy miner uses a mixture of abstraction and clustering approaches to achieve a process representation that can be more easily grasped by an analyst. It provides a relatively high-level view of the process by abstracting away undesired details and emphasizing details considered to be more important. It also allows a dynamic view of the process by replaying the event logs in the mined model, with animation to show the flow of the various cases through the model. Frequently taken paths are highlighted in the animation. A fuzzy model mined from the careflow logs is shown in Figure 12.
Figure 12  Fuzzy model mined from hospital careflow data, taken from (Mans et al., 2008)

(Mans et al., 2008) examined their mined models from the performance perspective, employing several techniques. Among these was the so-called dotted chart. Figure 13 shows the dotted chart for the careflow model. Events are displayed as dots, and time is displayed along the horizontal axis. Separate process instances or cases are displayed along the vertical axis and events are colored according to their task ID.

The dotted chart can provide several time perspectives, including actual, relative, and logical. Relative time, for example, shows the duration from the beginning of a process instance to a certain event, and thus indicates the duration of that instance. The dotted chart can also provide performance metrics such as the time of the first and last events, overall case duration, and number of events in a given process instance.

An analyst can obtain insights about the process by examining patterns in the dotted chart. For example, in Figure 13 the higher density of events on the left side of the diagram shows that patients receive more diagnosis and treatment early on in the process. Focusing on long-duration
process instances (i.e., those with events on the right side of the dotted chart), it can be observed that these consist mainly of regular consultation (red dot), phone consultation (blue dot), and lab tests (violet dot). It may also be possible for the analyst to discern meaningful patterns in the dotted chart.

![Dotted chart for hospital careflow logs, taken from (Mans et al., 2008)](image)

**Figure 13  Dotted chart for hospital careflow logs, taken from (Mans et al., 2008)**

While useful insights about the patient care process were gleaned from this application, (Mans et al., 2008) reached a conclusion similar to that reached by (Rozinat et al., 2009), namely that traditional process mining approaches have problems dealing with unstructured processes, and that more work is needed to develop new mining techniques for such processes, and to employ existing mining techniques in innovative ways to obtain understandable, useful models, rather than the spaghetti models produced by the traditional approaches.

**Research question**

*Can the tools developed by the business process mining community be employed to aid both in the HRA task analysis and in the simulator data analysis effort more generally?*

We begin to explore this question by examining some of the process mining tools and techniques in the context of simulator data. Later chapters will continue this examination in more detail by attempting to apply process mining to small-scale (Ch. 4) and large-scale simulator event logs (Ch. 6).

### 3.3 Mining a Petri net process model from simulator event logs

The Petri net model for a single crew, with the crew being the unit of analysis, is quite simple, as there is no branching structure if the crew is the unit of analysis (i.e., we are not modeling the actions of individual crew members). This is illustrated in Figure 14, which shows the beginning of the Petri net model for the data for Crew 1 listed in Figure 6. The rectangles represent
transitions (corresponding to actions taken by the crew) and the circles represent places. The Petri net was mined using the Alpha Algorithm Plugin (de Medeiros & van Dongen, 2004) in the ProM software package (van Dongen et al., 2005).

Figure 14 Petri net model excerpt for data for Crew 1 shown in Figure 6

If we mine a simulator log that contains information on the originator, or for multiple crews (i.e., process instances), then the Petri net begins to show branching structure, corresponding to interactions among the crew members or to differences in the implicit processes used by the different crews. Figure 15 shows the Petri net representing two crews responding to a steam generator tube rupture (SGTR) scenario. Both crews first respond to the alarm indicating the presence of radioactivity in the secondary system, followed by indications of decreasing pressurizer level and pressure. However, the crew response differs after this point, with one crew tripping the reactor manually, and the other crew communicating that there has been an SGTR.
Basic performance information can be extracted from the mined Petri nets from within ProM, by replaying the log traces of the event log in the Petri net. In such a way, one can obtain statistics on overall time, scenario time, and statistics for selected activities in the model, as well as estimates of branching probabilities. Figure 16 shows this analysis for two important actions: entry into the procedure for cooling down the ruptured steam generator, and actually starting the cooldown. This Petri net model was mined for two crews, and the results show the difference in crew response time, with one crew starting the cooldown 1 minute after entering the procedure, and the other crew taking 4.5 times as long to start the cooldown.
3.4 Extensions of process mining to flexible processes and noisy event logs

Figure 17 shows a Petri net model that captures the responses of 14 separate crews to SGTR in the simulator. The model, which was mined with the alpha algorithm (de Medeiros & van Dongen, 2004), is quite messy and of limited usefulness for visualizing the process. Models like this were termed “spaghetti models” by (Guenther C. W., 2009), and this is an apt phrase. This section presents an overview of approaches that have been developed to obtain more readily useful process models in the context of flexible processes and noisy event logs, which are representative of the situation with respect to simulator log data.
Figure 17  Petri net model for 14 crews mined with alpha algorithm, illustrating “spaghetti model” typically produced by mining algorithms with limited ability to handle noise

3.4.1 Heuristic nets

(Weijters & van der Aalst, 2003) describes an approach to discovering a process model from an event log that is more robust to noise than the alpha algorithm. The resulting model is termed a heuristic net. Figure 18 shows an excerpt of the heuristic net mined from the event logs used to produce the “spaghetti model” shown in Figure 17. This model provides some immediately useful information. For example, one can see variability in crew response to the initiating event (SGTR), with 10 crews first detecting the activity in the secondary system, 3 crews first responding to an automatic reactor trip, and 1 crew focusing on increasing water level in the affected steam generator. Information such as this is potentially valuable to the HRA task analysis as it illustrates the flexibility in the process that is governed by the emergency procedures, which are supposed to be guiding operator response to an upset condition. The reality shown by this model is that crews may not adhere rigidly to the procedures, performing steps out of sequence, for example. One could obtain this same information by poring over the detailed simulator logs, but the ability to extract a graphical model quickly in an automated fashion adds substantial value, as the HRA task analysis often must deal with resource limitations, which can be fairly severe.
Figure 18 Heuristic net excerpt for response of 14 crews to SGTR. Compare this with the “spaghetti model” in Figure 17.

The heuristic net can be converted to a Petri net model within ProM for further analysis. Figure 19 shows an excerpt of the Petri net produced from the heuristic net in Figure 18. The resulting model is much simpler to interpret than the “spaghetti model” of Figure 17.

Figure 19 Petri net generated from heuristic net shown in Figure 18
3.4.2 Fuzzy models

As illustrated above, some of the early approaches for mining Petri nets from event logs could easily, because of severe constraints in their assumptions about event log constitution, produce overly complicated models that are of little use for HRA task analysis. The heuristic net of (Weijters & van der Aalst, 2003) overcomes some of these limitations. However, as illustrated by the applications of process mining described in Sec. 3.2, this approach still has limitations, especially when the event log is highly detailed, containing numerous low level events that are of no real interest to the analyst. Figure 20 shows an excerpt of a heuristic net mined from a simulator event log containing such numerous low level events. As this figure illustrates, such event logs produce process models that are essentially opaque to analysis. While they may be accurate reflections of the implicit process followed by the operators, they contain too much detail to aid in the task analysis.

(Guenther C. W., 2009) describes adaptive process simplification techniques that can be used to produce models at a more abstract level of detail. Analogous to maps that do not show every street in a town, such models cluster low-level events together and highlight events of interest and the connections between them. Figure 21 shows such a model, termed a fuzzy model by (Guenther C. W., 2009), which was mined from the same event log used to produce Figure 20. The low-level “noise” that makes the model in Figure 20 so difficult to interpret has been abstracted away by clustering low-level events. The contents of each cluster can be examined, if desired, as illustrated by Figure 22, which expands the first cluster in Figure 21. The significance of paths through the model is shown by the width and darkness of the connecting arcs: significance is proportional to arc width, and more highly correlated events are connected by darker arcs. Details of the algorithms used to produce these models can be found in Appendix C and in (Guenther C. W., 2009).
Figure 20  Excerpt of heuristic net mined from highly detailed event log, illustrating limitations of heuristic net for such event logs

Figure 21  Fuzzy model mined from same event log used to produce Figure 20, clustering low-level events that obscure the high-level process of interest
Figure 22 Expansion of first cluster in Figure 21 showing relatively large number of low-level events aggregated into a cluster
As noted by (Guenther C. W., 2009), a fuzzy model is intended to aid the user in visualizing a complex process, not as an analytical model of the process itself. In this respect the fuzzy model differs from the Petri nets discussed above. As an aid to process visualization, a fuzzy model mined within ProM can be animated to illustrate the process followed by each crew. Figure 23 provides an illustration of this capability. This shows a still shot of the animation from one point in the process. The animation tool within the fuzzy miner allows the analyst to “interact” with the process. The analyst can zero in on one process instance, or can select multiple process instances for display simultaneously, allowing easy detection of variations from instance to another.

![Figure 23 Illustration of ability to animate a fuzzy model mined within ProM](image)

As described in (Guenther C. W., 2009), there are four concepts involved in the creation of a fuzzy model.

- **Aggregation**: low-level features are clustered together, just as features on a map such as individual buildings are placed within the boundaries of a particular city shown on the map.
- **Abstraction**: low-level information that is not significant in a particular context is omitted. For example, paths for pedestrians and bicyclists might be omitted from a map intended for automobiles.
- **Emphasis**: more significant information is highlighted visually by the use of color, contrast, size, etc.
- **Customization**: no one single model is useful for all purposes.

The overall approach to constructing the fuzzy model is as follows:

- Highly significant actions are preserved and highlighted in the model.
• Less significant but highly correlated actions are clustered together.
• Less significant actions with little correlation are omitted.

The details of how these steps are performed is described in Appendix C and in (Guenther C. W., 2009).

3.4.3 Trace clustering

In the approaches described above, there was no up-front exploratory data analysis; all information was pooled together to mine a single model (Petri net, heuristic net, or fuzzy model) that represented crew performance as a whole. In the case of Petri nets and heuristic nets, this approach typically produced an overly complex model, which was of little use in understanding the underlying implicit processes followed by the crews. The fuzzy mining approach, by clustering and abstracting away low-level details, produced a model more useful for understanding the underlying processes. However, unlike a Petri net, the fuzzy model is not suited for detailed quantitative analysis.

Rather than simply mining the composite event logs, pooled across crews, it might be useful to first examine the logs for each crew to determine how crews are similar and different in their response. This might allow partial pooling of log information according to how the information can be clustered. It might then be feasible to mine a Petri net for each cluster thus identified. This type of approach to log preprocessing was found to be useful in the application to healthcare described in Sec. 3.2.2 and in (Mans et al., 2008).

(Guenther C. W., 2009) discusses several approaches to what he calls process type discovery, and discusses several distance measures that can be applied to trace profiles. Loosely speaking, a trace profile is a set of characteristics that makes two process instances similar or different. Various metrics can be employed to define a trace profile: event names, number of events in a trace, who performed each action (originator), and other attributes associated with events in the logs. The description that follows will use event names, event name patterns, performance (based on event timing), and transition measures, based on direct following relations in the traces. Each of these will be equally weighted, for simplicity. The result is a profile for each trace, a vector with five components, one for each measure being used.

Trace clustering is based on the distance between the profile vectors, using a suitable distance metric and clustering algorithm. (Guenther C. W., 2009) describes the application of three distance metrics to event logs: Euclidean distance, Hamming distance, and Jaccard distance. The Euclidean metric tends to produce too many clusters in the logs that have been examined. The Hamming distance is suggested as being more robust than the Euclidean distance against profiles with frequently repeated patterns, a common issue with simulator event logs, so the Hamming distance will be employed in what follows, as it tends to produce a smaller number of clusters. (Guenther C. W., 2009) also discusses several clustering algorithms. In what follows, quality threshold clustering will be employed, as it does not require the analyst to specify the number of clusters in advance, unlike K-means clustering.

Applying quality threshold clustering to the combined event logs (14 crews), with the Hamming distance metric, ProM identifies eight clusters, as shown in Figure 24. Increasing the maximum
cluster size allows the number of clusters to be reduced, as shown in Figure 25. Note also in this figure that hovering the mouse over a particular trace allows the individual events in that trace to be identified.

Figure 24  Results of trace clustering for combined event logs of 14 crews, illustrating 8 identified clusters. Traces placed into cluster 7 are shown on right.
3.5 Summary

The tools of process mining appear to hold promise for improving HRA task analysis by allowing simulator observations to be incorporated into the analysis efficiently and effectively. There is also the potential for improving the analysis of simulator data generally. However, the applications that have been reviewed have served to highlight some apparent limitations. First, nontrivial effort may be required to transform the raw simulator event logs into the .mxml format required by ProM. Second, it may not be possible to obtain useful process models from highly detailed event logs without significant amounts of event log preprocessing. Third, even with preprocessing, traditional process mining techniques, such as the alpha algorithm (de Medeiros & van Dongen, 2004), and extensions such as the heuristic miner (Weijters & van der Aalst, 2003) may not be adequate to the task of producing useful models of the underlying process. New approaches to abstraction, such as the fuzzy mining method of (Guenther C. W., 2009), appear to hold promise, but it is not clear what the benefits will be of these approaches to analyzing simulator data generally, and to HRA task analysis in particular. Note that the author has not undertaken a systematic examination of all available process mining tools in an attempt to identify an ideal approach or combination of approaches to support HRA. Rather, selected techniques have been explored that appeared to be most immediately applicable to simulator event logs, and several approaches have been identified that appear to be promising for providing useful input to the qualitative HRA. The following chapters will explore the potential of these selected approaches by examining applications of process mining to actual simulator event logs, using techniques introduced in this chapter.
4 Application of process mining to international HRA empirical study

Chapter 3 presented an overview of tools developed in the business process mining community, some of which show promise for allowing simulator observations to be efficiently and effectively incorporated into the HRA task analysis. This chapter explores the application of these tools to simulator data collected at the Halden Reactor Project in Norway, under the auspices of the International HRA Empirical Study (Lois, et al., 2008), (U. S. Nuclear Regulatory Commission, 2010), referred to more briefly hereafter as the HRA Empirical Study.

This chapter begins the exploration of the fourth research question. This question will be explored further in the subsequent chapters.

Research question

What are the limits of applicability of these tools from other domains, and what improvements are needed in order to make them practical for use by an analyst who is not a specialist in such tools?

4.1 Obtaining Halden simulator data for case study

This section describes the process by which data were obtained from the Halden Reactor Project in Halden, Norway. It should be noted that a substantial amount of effort was expended in this process, and some insights were gleaned that should lessen the efforts required for future analyses.

The Halden case study was initiated in a discussion during a meeting attended by researchers from Halden, in early 2009. Follow-on discussions were held with staff of the U.S. NRC to arrange for data to be shared under a nondisclosure agreement. More extensive meetings with the Halden researchers were held at Halden in Spring 2009. Discussions during this meeting focused on the overall plan for the case study, and the concept of process mining was introduced to the Halden staff. During this meeting the author toured the simulator facility that had been used in the experiment and reviewed the scenario and associated procedures used by the operators who took part in the experiment.

Simulator data from one of the experiments were provided during this visit, in Microsoft Excel format. Separate files were provided for each of the following: operator actions, annunciators, process variables, and simulator status. The first step was to decide which of these files were candidates for application of process mining. The operator action files are obviously a candidate. Some of the information in the annunciator files is also applicable. The process variable data could not be incorporated directly into the mined model, but these are nonetheless valuable for later analysis. The simulator status files are generally not useful for process mining, as they contain only information as to when a simulation was “frozen,” and interactions of the researchers with the simulator.

The first difficulty in mining these files was that the operator action data were extremely dense and minutely detailed, recording every interaction of an operator with the touch screen controls in
the Halden simulator. Furthermore, all of the low-level logs were in Swedish, and required translation.\textsuperscript{6}

The data were converted by the author to comma-separated variable (.csv) format to allow easier conversion to the Mining Extensible Markup Language (.mxml) format required by ProM. Efforts were next put into developing a plug-in written in Java to allow conversion to .mxml format using the PromIMport package (Guenther & van der Aalst, 2006). A number of difficulties were encountered during this conversion, among them issues with timestamp formatting (e.g., use of comma to indicate decimals), sporadic inclusion of quotes in the event description field, and a mixture of delimiters to separate the fields in the data.

Difficulties in the conversion to .mxml format, along with the lack of an English translation, combined with the difficulties in interpreting the mined models, ultimately led the author to abandon the low-level Halden logs for the first case study. The author made a second trip to Halden in July 2009 to meet with the researchers there to explore additional options. To facilitate the case study, the Halden researchers made available a higher level set of logs, which had been constructed to facilitate analysis of the data by manually abstracting much of the detail from the simulator event logs, and translating the abstracted high-level logs into English. These resulting logs were significantly smaller in size, and contained information that was more at the level of procedural actions carried out by the operators. A sample of the abstracted log is shown in Table 2.

\textsuperscript{6} Because of confidentiality, the raw data cannot be presented. The data are available from the author for inspection if the reviewer accepts the required nondisclosure agreement.
Table 2 Sample of Halden simulator log data after abstraction and translation

<table>
<thead>
<tr>
<th>Activity</th>
<th>Time</th>
<th>Comment</th>
<th>Crew</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detect Activity Alarm</td>
<td>0:03:18</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Detect Pressurizer Pressure decreasing</td>
<td>0:03:45</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Detect Pressurizer Level decreasing</td>
<td>0:03:52</td>
<td>Shift supervisor detects, communicates</td>
<td>1</td>
</tr>
<tr>
<td>Manual Reactor trip</td>
<td>0:05:03</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Start E-0</td>
<td>0:05:30</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Detect PRZ level &lt; 10%</td>
<td>0:07:35</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Manual Start of SI when level prz &lt; 10%</td>
<td>0:07:40</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Verify SI</td>
<td>0:07:56</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Complete step 10 in E-0 (cont.spray)</td>
<td>0:09:53</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Detect increasing level in SG</td>
<td>0:10:53</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Stop auxiliary feedwater to faulted SG before E-3</td>
<td>0:11:41</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Enter step 19 in E-0 (identification)</td>
<td>0:16:25</td>
<td>00:04:15 Shift supervisor says &quot;We must start AOP&quot;, which is the tube leakage procedure (as a response to ARO informing about leak size).</td>
<td>1</td>
</tr>
<tr>
<td>Communicate SGTR</td>
<td>0:16:49</td>
<td>00:17:00 RO wants to go to E-3. SS is OK with this decision, but wants to take a quick meeting here.</td>
<td>1</td>
</tr>
<tr>
<td>Transfer to procedure E-3</td>
<td>0:19:23</td>
<td>E-0 step 19</td>
<td>1</td>
</tr>
<tr>
<td>From what procedure and step the crew entered E-3</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Completion of E-3 step 2</td>
<td>0:20:20</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Steam dump atmosphere valve set point to 70.7 bar</td>
<td>0:20:41</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Isolate faulted SG according to E-3 (Time is to be adjusted to include control room actions, but not response from field operator (E-3, step 3 a-h))</td>
<td>0:24:44</td>
<td>00:18:15 Shift supervisor says in the meeting that they should alert personnel. 00:24:00 RO reads from procedure (E-3) that SS should perform this. He says that it's done (doesn't think he has to do it in the simulator perhaps)</td>
<td>1</td>
</tr>
<tr>
<td>Alert personnel and emergency organisation</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Enter step 7 (cooldown) in E-3 procedure</td>
<td>0:25:00</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Increase auxiliary feedwater flow to the intact SGs before or as cooldown is started.</td>
<td>0:25:50</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Start Cool down when ruptured SG level &gt; 17%</td>
<td>0:26:00</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Cool down with maximum speed</td>
<td>0:28:14</td>
<td>This was ticked early in the cooldown, but the crew then has a steam line isolation which slows them down a bit.</td>
<td>1</td>
</tr>
<tr>
<td>Cool down RCS to temperature corresponding to pressure in faulted SG, equal to or less than Table</td>
<td>0:34:28</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Steam dump atmosphere valve set-points to current SG pressure (to keep RCS temperature constant) in intact SGs</td>
<td>0:34:45</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Enter step 16 (depressurization) in E-3 procedure</td>
<td>0:35:22</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Depressurize RCS with SPRAY</td>
<td>0:37:10</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Depressurize RCS with PORVs</td>
<td>0:40:30</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>End Depressurization (CREW 1-4, added activity after the run)</td>
<td>0:41:32</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Verify RC pressure (according to E-3 Step 18) (CREW 1-4, added activity after the run)</td>
<td>0:41:40</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Check criteria for stop of SI (Pressure stable or increasing, secondary heat sink, pressurizer level, sub-cooling)</td>
<td>0:42:27</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

The logs of the abstracted data were provided to the author in Microsoft Excel format, and were converted to the .mxml format used by ProM, using a Java plug-in written for ProMImport.
An additional complication for the mining of these data was that there was a timestamp but no time duration associated with any of the actions in the high-level log, and thus each action was mapped to a “complete” event during the conversion to .mxml, allowing the timestamp information to be used by ProM.

Once the high-level log data were converted to .mxml format, they were loaded into the ProM software for process mining. An artificial start and end task were added to the log to provide clearer structure for the mined models, as suggested by (Guenther C. W., 2009) for mining loosely defined processes.

Some of the difficulties encountered in processing the data for the first case study will likely be encountered in future applications of process mining to simulator data, as there is significant variety in how simulator data are recorded, and the general issue of data conversion could present a significant hurdle to the application of process mining. In summary, the principal difficulties were

- Multiple log files in Excel format with a mixture of delimiters,
- Problems with timestamp formatting,
- Logs in Swedish required translation to English,
- Logs at extremely low level of detail (e.g., each time operator touched screen was recorded).

To ease these difficulties, the analyst should make every attempt to obtain log files in .xml format, which can be converted much more easily to .mxml. The most critical logs are those containing the operator actions. Without these logs, process mining to discover the implicit process followed by the operators is impossible. Annunciator logs are also useful, but somewhat less critical. Process variable logs are very helpful, but at the moment cannot be directly incorporated into the mined models. Finally, the simulator status logs, which show points at which the simulation was frozen and actions by the analysts running the experiment are the least useful, and can typically be dispensed with.

The analyst should attempt to determine in advance what information is logged in the simulator, and in what format (and in what language!). If possible, the level of detail that is logged should be commensurate with the goals of the analysis that the mined models will support. Logs at an extremely fine detail will likely require substantial preprocessing and filtering, steps that will be described in more detail in Ch. 6.

4.2 Overview of international HRA empirical study

The following description is summarized from (U. S. Nuclear Regulatory Commission, 2010). The Office of Nuclear Regulatory Research of the U.S. Nuclear Regulatory Commission (NRC) supported the initiation and execution of a research project that would develop an empirical basis for evaluating human reliability analysis (HRA) methods. This project was an international collaborative effort, and involved the use of the Organization for Economic Co-Operation and Development’s (OECD) Halden Reactor Project HAMMLAB (HAlden huMan-Machine LABoratory) research simulator, which is a full-scope nuclear power plant simulator located in Halden, Norway. The study aims were to provide empirical information on how well HRA methods can identify factors that may lead to crew failure, and, to a lesser degree, on how well
the HRA methods can estimate failure probabilities. The empirical basis was developed through experiments performed at the HAMMLAB simulator, with actual operating crews from Swedish nuclear plants responding to accident situations similar to those modeled in probabilistic risk assessments (PRAs).

In PRAs, the post-initiator operator actions, which are modeled and quantified via HRA methods, are associated with postulated scenarios that are beyond the design basis of the facility being analyzed. For instance, PRA scenarios often include multiple equipment failures, both active and passive, while the design basis scenarios assume a limiting single active failure. Simulator data is therefore necessary to support modeling and quantification of operator errors in such scenarios, due in part to the low frequency of these multiple-failure scenarios; however, designing and implementing such scenarios in simulator studies presents a challenge because the scenarios must include component and system failures in combinations that lead to the required operator actions of interest, yet the scenarios must remain plausible to the operators.

Two initiating events were chosen for simulation: steam generator tube rupture (SGTR) and loss of feedwater (LOFW). For both of these initiators, both a simple and a complex scenario were developed. Having two scenarios for each initiator was particularly useful because the two scenario variants included similar or related tasks that differed only in terms of their performance contexts. This allowed for a comparison and analysis of the differences in performance in order to determine the effects of context difficulty. To control for ordering effects, such as learning and other potential biases, the scenarios were presented to the crews in a semi-randomized order. This approach provided a more complete understanding of the crew actions than would be afforded by individually examining unrelated scenarios. It also allowed for an evaluation of whether the HRA methods are sensitive to such scenario differences, and whether their predictions need to be adjusted accordingly.

For this dissertation, operator action logs were available for the base case SGTR scenario only. In this scenario, an SGTR is initiated in steam generator (SG) number 1 (SG1), and the rupture is sufficiently large to cause nearly immediate secondary radiation alarms and other abnormal indications/alarms, such as SG1 high water level, as well as decreasing water level and steam pressure in the pressurizer, which contains a mixture of water and steam at saturation pressure. While plant conditions are continually degrading immediately following the tube rupture, they are not severe enough to immediately cause an automatic reactor trip. About three minutes after the tube rupture initiator, the large screen display in the control room will be indicating decreasing pressurizer steam pressure and water level, increased charging flow (as the charging system attempts to make up for the loss of primary reactor coolant through the ruptured steam generator tube), increasing SG1 water level caused by flow of primary coolant into the secondary side of the steam generator through the ruptured tube, and a slight imbalance in feedwater flow to the SGs. If the operating crew also calls up the radiation monitoring display screen, they will see higher radiation indications associated with SG1. It is expected that at this point, or as conditions continue to deteriorate over the next few minutes, the crew will manually trip the reactor. Even if they do not, an automatic trip will occur eventually due to low pressurizer steam pressure or some other trip setting. Whatever the case (manual or automatic reactor trip), the crew is expected to enter emergency procedure E-0 following the trip (see App. E for a discussion of the procedures that govern operator response to transients such as SGTR). About 10 minutes after entering procedure E-0 (if the crew has not been delayed carrying out the steps in this procedure), the
crew should reach step 19, which is the step at which radiation indications of an SGTR necessitate a transfer to emergency procedure E-3 (the SGTR procedure). At this point, secondary radiation is high (as it has been virtually from the beginning of the scenario), and the water level in SG1 is still higher than that in the other SGs. After the reactor trip, imbalances in the auxiliary feedwater (AFW) flow rate may also exist among the SGs. While it is expected that the crew may enter procedure E-3 at this point, it is noted that a couple of steps later, in E-0, there is another step calling for a transition to E-3, based on a check of SG water level (the procedural direction is that if any SG level is rising uncontrollably, the operators should enter procedure E-3).

When the crew enters procedure E-3, the base case scenario proceeds in response to the crew’s actions, with no equipment failures or other complicating factors induced by the simulation design, that is, the plant response will be determined only by the crew’s actions after entry into procedure E-3. In general, the crew is expected to perform four critical tasks in the base SGTR scenario: (a) identifying and isolating the ruptured SG, (b) quickly cooling down the reactor coolant system (RCS) by discharging steam from the unaffected SGs, (c) quickly depressurizing the RCS using the pressurizer sprays or a pressurizer power operated relief valve (PORV) to expedite the depressurization, and (d) stopping safety injection (SI) to the reactor upon indication that the SI termination criteria in E-0 are met. Failure to perform one of these tasks is a human failure event (HFE) in the PRA. For the base case SGTR scenario, these four HFEs are designated sequentially as 1A to 4A.

Figure 26 shows a typical PRA event tree for an SGTR initiating event. It is presented here to provide an overall PRA context for the HFEs to be evaluated in the supporting HRA. Its sequence end states (outcomes) refer to whether in the long term the reactor core is safe or there is extensive core damage (CD). The paths through the event tree that are of interest and the relevant HFEs are highlighted. Those are the only sequences that were simulated in the Halden experiments.

As an accident sequence model, the event tree illustrates at a high-level the manner in which the operators are trained to respond to an SGTR initiating event using the E-3 procedure. However, when performing a PRA, the event tree success criteria are typically based on avoiding irreversible changes to the plant state that will lead ultimately to core damage. For this exercise, the training staff’s expectations about the operator responses were considered in establishing the success criteria. These expectations are reflected in the crews’ training. In working through the procedures, the operators are also trained to be concerned about more intermediate and detailed goals than avoidance of core damage, and these goals are particularly relevant to an SGTR event. From the operators’ perspective, “success” means timely operator intervention that limits offsite radiological releases that occur even without core damage and prevents steam generator (SG) overfill. Because of these two goals, operators are trained to terminate primary-to-secondary leakage expeditiously. They want to limit offsite radiological releases that are, in part, a function of how long it takes before the tube rupture is mitigated, and they do not want to overfill the ruptured SG as this could cause a pressure relief valve to open, thereby allowing a larger release or worse yet, instigating a main steam line leak or break, which would release yet more radioactivity and further complicate attempts to bring the reactor to a safe shutdown state.
Figure 26 PRA event tree for SGTR scenario with HFEs of interest highlighted. From (U.S. Nuclear Regulatory Commission, 2010)
The steps in the E-3 procedure are directed at the goal of limiting the offsite radiological release. For the HFEs shown in Figure 26, the relevant steps to achieve this goal are identifying and isolating the ruptured SG, cooling down and depressurizing the RCS, terminating safety injection (SI), and achieving a pressure equilibrium between the primary and secondary systems. Because of the overall goal of limiting radiological release, the operators are trained to perform these actions expeditiously (and the E-3 procedure is designed accordingly). Furthermore, the operators are trained that failure of any of these tasks has undesirable consequences. For example, until the affected SG is identified and isolated, radiological releases will remain undesirably high – an outcome to be avoided.

In addition to their training about the undesirable consequences of SGTR and the need to perform actions specified in the procedures expeditiously and correctly, the operators are also trained that in order to limit the release, all the actions should be completed before the ruptured SG overfills. So, while operators do not think in terms of clock time, they are aware of the need to move through the procedural steps with some celerity in order to achieve the overall goal of limiting the offsite radiological release. And SG water level acts as a surrogate clock for the operators, with increasing or decreasing level being an important gauge of the scenario timing and physics. Based on this sense of urgency to prevent SG overfill, some level of expectation exists regarding typical response times to perform the various tasks in a simulator scenario used for training. It was on the basis of these temporal expectations, along with what is to be accomplished for each task, that the HFE definitions of success and failure were based. While the threshold times to perform each task as provided in the HFE definitions are not exact, they do represent times by which the operators could be viewed as being slower than expected, since the overall goal of limiting the radiological release offsite could then be jeopardized.

Based on these considerations, the HFEs were defined as follows:

**HFE-1: Failure of the crew to identify and isolate the ruptured SG:**

Success requires that the crew:

- Enter procedure E-3 (preferably from procedure E-0 Step 19),
- Isolate all steam outlet paths from the ruptured SG, and
- Stop all feedwater flow to the ruptured SG for as long as water level in the ruptured SG is at least 10%, as indicated on the narrow range SG level indications (to ensure the SG U-tubes will remain covered with water).

Further expectations are that it should typically take the crew about 8-10 minutes after entering procedure E-0 to reach the vicinity of step 19 in E-0 (the desired transfer point to procedure E-3). Allowing a few minutes before the plant trip for the crew to observe and evaluate the initial indications of the tube rupture, about 8-10 minutes to enter E-0 and reach step 19, an additional 5 minutes for the crew to actually transition to procedure E-3 and perform the initial actions of that procedure, and an additional few minutes for reasonably acceptable variability among crew responses, not performing the above actions by 20 minutes (base case, HFE 1A) after the tube rupture occurs (which is the start of the event) was defined as “failure”, as this would be a slower response than expected or desired. The actions needed to isolate the ruptured SG are the following and would typically take less than 3 minutes to execute:
- **Control room actions.** These are all expected to be performed by the crew in the simulator and are part of the HFE:
  - Verify opening pressure set point for valve that discharges steam to atmosphere set at 70.5 bar,
  - Verify SG blowdown pathways are isolated,
  - Verify main feedwater flow is isolated,
  - Close valve that supplies steam to turbine-driven AFW pump,
  - Close main steam line isolation valve and its bypass valve,
  - Stop AFW when SG water level exceeds 10%.

- **Actions outside control room.** The crew should call for these actions to occur, which are part of this HFE.
  - Verify valve that discharges steam to atmosphere is closed,
  - Lock steam supply valve to turbine-driven AFW pump in the closed position,
  - Verify steam traps are closed.

**HFE-2 (A & B): Failure of the crew to cool down the RCS expeditiously:**

The crew is supposed to cool down the RCS much faster than 100° F/hr in the SGTR base case scenario. This cooldown is anticipated to be performed by discharging steam from one or more intact SGs. Success requires that the crew:

- Perform the cooldown using atmospheric steam dump valves, or by discharging steam to the main condenser at such a rate that an RCS temperature corresponding to the pressure in the faulted SG is reached, along with corresponding adequate RCS subcooling, and terminate the cooldown upon reaching these conditions,
- Maintain RCS temperature below the specified limit value.

Expectations are that this initial cooldown should take about 10 minutes if performed in the desired expeditious manner, once the cooldown procedural step (step 7 in E-3) is reached. Failure to successfully expeditiously cooldown and then terminate the cooldown upon meeting the above criteria within 15 minutes of reaching the cooldown step in E-3 (step 7) constituted “failure”, as this would be a slower response than expected or desired, even allowing for some variability in the speed of the crews.

**HFE-3 (A & B): Failure of the crew to depressurize the RCS expeditiously:**

The purpose of the actions contributing to this HFE is to minimize the flow of RCS inventory into the ruptured SG and to replenish lost RCS coolant via injection into the pressurizer. While the goal is to depressurize the RCS and then subsequently terminate depressurization once the crew achieves an RCS pressure less than the pressure in the ruptured SG, ultimate success (so as to be able to move on in the procedure) requires that the crew:

- Achieve and maintain a pressurizer water level greater than 10%,
- Avoid exceeding a pressurizer water level greater than 75% (the crew should stop depressurization even if the RCS pressure is not less than the pressure in the ruptured SG), and
• Avoid excessive RCS subcooling by virtue of not maintaining the RCS pressure and temperature within the allowed range using the procedurally prescribed subcooling margin.

Further, since there is a sense of urgency associated with these actions, it is desirable that the depressurization be completed in less than ~10 minutes once the depressurization step in E-3 (step 16) is reached. Allowing for reasonably acceptable variability among the crews, failure to expeditiously depressurize while meeting the above success criteria within 15 minutes of reaching the depressurization step in E-3 (step 16) constitutes “failure” for this HFE, as this would be a slower response than expected/desired.

**HFE-4A: Failure of the crew to terminate SI:**

Success requires that the crew:

• Stop injection from all but one high-pressure SI pumps, isolate the SI flow path, and establish charging flow from the single remaining SI pump when the shutoff pressure criteria in the E-3 procedure are met, so that the crew can maintain RCS coolant level and pressure control, and
• Stop SI before the RCS repressurizes to a pressure greater than the ruptured SG pressure (assuming it was lower after the cooldown and depressurization).

It is preferable that the SI termination occurs without overfilling the ruptured SG (sustained 100% level on indicating wide range instruments) but this is not a requirement.

Note that the manipulations involved with successful execution of the actions in the first bullet above require that the following be performed:

• Stop all but one charging pump, with its suction remaining aligned to the refuelling water storage tank (it should already be so aligned) and verify the charging pump’s minimum flow recirculation valves are open,
• Isolate the boron injection tank (BIT), which is an alternative source of water for injection, by closing the two BIT inlet isolation valves, along with the two BIT outlet isolation valves, and verifying the BIT bypass valve is closed.

### 4.2.1 Summary of crew performance for base case SGTR scenario

The Halden researchers analyzed the performance of each crew for each of the four HFEs listed above. This analysis is summarized below. See (Lois, et al., 2008) and (U. S. Nuclear Regulatory Commission, 2010) for more details.

Table 3 shows the performance times for HFE-1A in the base case scenario and the SG level at the time of isolation. Only one crew (crew N) failed to meet the criterion of isolating the ruptured SG within 20 minutes.
Table 3 Summary of HFE results for Halden base case SGTR scenario

<table>
<thead>
<tr>
<th>Crew</th>
<th>SG isolation</th>
<th>cooldown</th>
<th>Depressurization</th>
<th>RCS-SG1</th>
<th>Stop SI</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0:13:33</td>
<td>0:05:55</td>
<td>0:06:20</td>
<td>1.4^2</td>
<td>Yes</td>
</tr>
<tr>
<td>B</td>
<td>0:13:19</td>
<td>0:08:10</td>
<td>0:08:41</td>
<td>-0.8</td>
<td>Yes</td>
</tr>
<tr>
<td>C</td>
<td>0:18:53</td>
<td>0:10:10</td>
<td>0:06:06</td>
<td>-5.2</td>
<td>Yes</td>
</tr>
<tr>
<td>D</td>
<td>0:18:30</td>
<td>0:12:16</td>
<td>0:07:22</td>
<td>-8.3</td>
<td>Yes</td>
</tr>
<tr>
<td>E</td>
<td>0:14:22</td>
<td>0:06:25</td>
<td>0:08:43</td>
<td>-1.2</td>
<td>Yes</td>
</tr>
<tr>
<td>F</td>
<td>0:18:45</td>
<td>0:04:41</td>
<td>0:04:32</td>
<td>0.4^2</td>
<td>Yes</td>
</tr>
<tr>
<td>G</td>
<td>0:21:52</td>
<td>0:08:50</td>
<td>0:06:20</td>
<td>2.4^2</td>
<td>Yes</td>
</tr>
<tr>
<td>H</td>
<td>0:11:59</td>
<td>0:15:10</td>
<td>0:05:42</td>
<td>-2.1</td>
<td>Yes</td>
</tr>
<tr>
<td>I</td>
<td>0:13:37</td>
<td>0:08:05</td>
<td>0:02:38</td>
<td>-1.8</td>
<td>Yes</td>
</tr>
<tr>
<td>J</td>
<td>0:17:38</td>
<td>0:04:07</td>
<td>0:05:31</td>
<td>1.6^2</td>
<td>Yes</td>
</tr>
<tr>
<td>K</td>
<td>0:15:09</td>
<td>0:06:10</td>
<td>0:05:18</td>
<td>15.7</td>
<td>Yes</td>
</tr>
<tr>
<td>L</td>
<td>0:13:06</td>
<td>0:06:20</td>
<td>0:12:59</td>
<td>-0.2</td>
<td>Yes</td>
</tr>
<tr>
<td>M</td>
<td>0:10:23</td>
<td>0:06:15</td>
<td>0:08:54</td>
<td>-3.0</td>
<td>Yes</td>
</tr>
<tr>
<td>N</td>
<td>0:21:29</td>
<td>0:10:30</td>
<td>0:06:09</td>
<td>-3.1</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Criterion^3 | 0:20:00 | 0:15:00 | 0:15:00 | <0     | All     |
Average       | 0:16:10 | 0:08:32 | 0:07:21 | -0.3   | -       |

*Failure - Criterion exceeded but not counted as failure*

^1 This crew is not counted as failing. This crew responded to the rupture by starting the tube failure procedure AOP-3 (small leakage up to 10 kg), which implies not to trip the reactor. This response was correct, as the leakage started slowly, in the range covered by the tube failure procedure. When the leakage increased, they tripped the reactor and started E-0. This behavior caused the crew to start E-0 3-4 minutes later than otherwise. Also, as the tube damage procedure was newly adopted at the home plant at the time of the experiment, the crews were not expected to use it. Hence, no information was provided about AOP3 in the information package. The HRA teams had no reasons to imagine this source of reactor trip delay.

^2 RCS-SG pressure differences between 4-0 bars are not counted as failures, as they do not compromise the achievement of further recovery actions (stop SI and pressure balance). Yet, they are taken into account in the difficulty rating of the HFES.

^3 The HFES definitions included several other criteria. Here only the criteria that produced failures are reported.

4.3 Application of process mining to Halden simulator data

There were 14 crews who participated in the simulator experiments at Halden. Information on how the crews performed was already known to the author before attempts were made to apply process mining to the simulator data, so in a sense this is not perhaps an optimum test of the insights to be provided by process mining; it did, however, provide a useful small-scale test bed for becoming familiar with the tools. Subsequent chapters will explore an application of process mining to another set of simulator data, on a much larger scale, where the results were not known to the author in advance.

As a first step, the alpha-algorithm (de Medeiros & van Dongen, 2004) was used to construct a Petri net for the combined logs (all 14 crews). The resulting model was quite complex, as
indicated by the excerpt shown in Figure 27. This is despite the small size of the event logs that were mined. The overall fitness of this model, as measured by the Conformance Checker plug-in to ProM (Rozinat & van der Aalst, 2007) was about 75%.

Figure 27 Excerpt of Petri net mined with alpha algorithm on combined log data for all 14 crews, illustrating complexity of resulting model

The heuristic miner (Weijters & van der Aalst, 2003) was applied next, as it was designed to overcome some of the limitations of the alpha algorithm, which is more appropriate for a strictly defined process. The resulting model was significantly less complex, and shows some features that are immediately useful for HRA task analysis. One of these is illustrated in Figure 28, which shows the alternative paths taken by some of the crews early on in the scenario. As this model shows, 10 of the 14 crews behaved as expected by responding first to the alarm indicating the presence of radioactivity in the secondary system, which is the compelling procedural indication of SGTR. However, three crews responded first to the reactor trip, while one crew responded first to increasing water level in the ruptured SG.
Figure 28  Excerpt of heuristic net for base case SGTR scenario, illustrating variety in early crew response to the initiating event

Converting the mined heuristic net to a fuzzy model within ProM allows the animation tool in ProM to be used to visualize the flow through the process model for each crew. Figure 29 shows a still from the animation for Crew G, showing their initial response to increasing water level in the ruptured SG, followed by detection of secondary radioactivity.
The heuristic net can also be converted to a Petri net within ProM to allow quantitative analysis. An excerpt of the resulting model is shown in Figure 30. This model is significantly less complex than that illustrated in Figure 27, which was mined directly from the event logs using the alpha algorithm. The fitness of this model is about 83%, compared to 75% for the Petri net mined directly with the alpha algorithm.

The Petri net derived from the mined heuristic net can be used to extract some simple quantitative information by replaying the event logs against the model in ProM. For example, the average time taken by a crew to complete the SGTR scenario was about 53 minutes, with a range of 43 minutes for the fastest crew to 68 minutes for the slowest crew. Branching probabilities are estimated by ProM for each transition with multiple outgoing arcs in the Petri net model. The estimate is obtained by dividing the number of crews taking a particular arc by the total number of crews (14). The relative transition times are indicated by coloring the arcs. This is illustrated in Figure 31.
Figure 30  Excerpt of Petri net produced by first mining a heuristic net, then converting to a Petri net within ProM

Figure 31  Results of basic performance analysis for the Petri net shown in Figure 30, illustrating display of transition probabilities and relative transition times
4.3.1 Application of clustering algorithms to Halden data

As discussed in Ch. 3, it is potentially useful to examine the logs for each crew to determine how crews are similar and different in their response. This might allow partial pooling of log information according to how the information can be clustered. It might then be feasible to mine a Petri net for each cluster thus identified. The trace clustering approach introduced in Ch. 3 and described in more detail in (Guenther C. W., 2009) is applied in the following, using the Hamming distance as the clustering metric, with sensitivity analyses to examine the impact the metric choice has on the resulting clusters. The quality threshold clustering algorithm is employed, as it does not require the analyst to specify the number of clusters in advance.

Running the quality threshold algorithm in ProM, with the Hamming distance as the clustering metric, produces eight clusters, all but two of which contain only one process instance (i.e., crew). One cluster is significantly larger than the others, containing 6 process instances; the other nonsingular cluster contains two process instances. Increasing the maximum cluster diameter to just below the limit allowed in ProM reduces the number of clusters to two, one with 13 instances and one with only 1 instance. The smaller cluster contains only Crew G. As discussed further below, Crew G is an outlier from the other crews in that they first responded to increasing water level in the ruptured SG, while the other crews took the more expected action of responding first to the secondary activity alarm.

The same result is obtained in the limit of maximum cluster diameter when Euclidean distance is used as the metric. With the default maximum diameter of 0.5, the Euclidean distance metric produced eight clusters, but they are not the same as those produced using the Hamming distance as the metric. Table 4 shows the clusters produced with the two metrics. The results are qualitatively similar, with each metric producing one relatively large cluster (i.e., containing multiple crews) and many small clusters, most containing only one crew. However, there is some difference in the cluster assignments. Interestingly, an increasing maximum cluster diameter results in Crew G being an outlier with either distance metric, although with the Hamming distance Crew G is initially clustered with Crew L. It is also interesting to note that Crew N, which was the only crew to “fail” at HFE-1A, is clustered with several other crews using either distance metric, suggesting that the underlying process governing that crew’s performance was not very different from that of several other crews, so perhaps failure was just a matter of the time taken on certain tasks. In other words, perhaps several other crews were “close” to failure on this HFE, because their underlying process was similar to that of Crew N, who failed, and they succeeded merely because they were somewhat faster at working through the tasks.

Table 4 Results of trace clustering with two different distance metrics, using default maximum cluster diameter of 0.5. Clusters are listed in order of increasing distance.

<table>
<thead>
<tr>
<th>Hamming distance</th>
<th>Euclidean distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster 7: A, B, C, E, F, N</td>
<td>Cluster 5: A, B, L, M</td>
</tr>
<tr>
<td>Cluster 2: H</td>
<td>Cluster 7: C, E</td>
</tr>
<tr>
<td>Cluster 3: I</td>
<td>Cluster 1: G</td>
</tr>
<tr>
<td>Cluster 4: J</td>
<td>Cluster 2: H</td>
</tr>
<tr>
<td>Cluster 6: K</td>
<td>Cluster 3: I</td>
</tr>
<tr>
<td>Cluster 9: D</td>
<td>Cluster 4: J</td>
</tr>
<tr>
<td>Cluster 10: M</td>
<td>Cluster 9: D</td>
</tr>
</tbody>
</table>
4.3.2 Insights for HRA task analysis provided by process mining

The ultimate question to be answered is whether process mining can provide insights to the HRA task analysis that would be difficult or impossible to obtain in another way. The mined heuristic net can be used directly to see some of the variations in the operational stories that a task analysis would attempt to capture. For HFE-1A (failure to isolate the faulted SG), some of these are:

1. All crews detected the activity alarm, as shown in Figure 28.
2. As shown clearly by the heuristic net in Figure 33, most crews stopped AFW to the ruptured SG before entering procedure E-3 (typically when reading the note in E-0, step 12), and about 4 minutes after entering procedure E-0, as indicated from the Petri net summary statistics. However, 3 crews entered E-3 before terminating AFW flow. This can be seen by converting the heuristic net to a fuzzy model in ProM and using the animation tool to follow each crew through the model. Figure 32 shows a case where a crew entered E-3 prior to terminating AFW flow.

3. Following entry into procedure E-0, Step 19 (transition to procedure E-3) was reached on average in about 6.5 minutes, as indicated by the Petri net summary statistics. However, the range of times was quite wide, from about 4.5 to almost 11 minutes.
4. As shown by the heuristic net, all 14 crews transferred to procedure E-3 within an average of one minute of reaching step 19 in E-0. However, several crews took a few minutes (up to 3.5) to make the decision to enter E-3. Again, this can be seen easily using the animation tool with a fuzzy model of the process. The reasons for this delay are not
evident in the mined model, but can be gleaning from the videos made of each simulator run. At this stage, although the situation was straightforward, several crews decided to hold a meeting to assess the status and to develop a strategy before entering E-3. This was the case for the crew that failed to isolate the faulted SG within the 20-minute success criterion for HFE-1A, for example.

5. The average time from entering procedure E-3 to isolating the faulted SG was about five minutes, with a few crews taking as long as eight minutes. These longer times seemed to be due to problems encountered with step 3 in E-3, and perhaps due to the overall complexity of the procedure.

6. Crew N, which was the only crew to “fail” at HFE-1A, is clustered with several other crews using either distance metric that was considered, suggesting that the underlying process governing that crew’s performance was not very different from that of several other crews, so perhaps failure was just a matter of the time taken on certain tasks. In other words, perhaps several other crews were “close” to failure on this HFE, because their underlying process was similar to that of Crew N, who failed, and they succeeded merely because they were somewhat faster at working through the procedural tasks.

Figure 33 Excerpt of heuristic net showing that 11 crews stopped AFW flow to the ruptured SG before entering procedure E-3

Some parts of the operational story cannot be seen easily in the mined Petri net or heuristic net. For example, four crews started SI manually before entering procedure E-0; however, it is difficult to glean this information from either the mined heuristic net or the Petri net derived from the heuristic net. It can be seen, however, in the animation tool within the fuzzy miner, after converting the Petri net model to a fuzzy model within ProM. The animation tool allows the analyst to watch each trace (crew) through the fuzzy model. By examining each trace in this way, one can clearly see instances where SI was started manually before entering procedure E-0.
Figure 34 shows a crew that entered E-0 first, and Figure 35 shows an opposite instance, in which SI is manually started before entering E-0.

Figure 34  Fuzzy model animation illustrating instance in which crew enters procedure E-0 before manually starting SI

Figure 35  Fuzzy model animation illustrating instance in which crew manually starts SI before entering procedure E-0
An important desideratum for HRA task analysis is that it be possible to assess performance drivers from the mined models; this does not seem to be possible with the models mined above, and may illustrate a limitation of current process mining tools for supporting HRA task analysis. Assessing performance drivers appears to require direct simulator observation (either in person during the exercise or by reviewing audio and video of the simulation). For example, for the crew that failed at HFE 1A, the following negative performance drivers were gleaned from direct observation:

- The assistant reactor operator did not use the large overhead screen efficiently and took time to navigate to the appropriate screen at the workstations. The shift supervisor did not focus on preventing overfilling of the ruptured SG and did not focus on speeding up the work (unaware of scenario dynamics). The shift supervisor interrupted the crew at times with less important issues.
- Crew followed good operational practices (meeting at procedure transfer, thorough checks and verifications). “…whole crew were clearly updated and coordinated on the situation and chosen strategy” but thoroughness and unwarranted attention to detail slowed them down in this scenario.
- Reactor operator tended to wait for assistant reactor operator and did not work independently, thus slowing down the crew.

With regard to HFE-2A (failure to cool down RCS expeditiously), some insights gleaned from the mined heuristic net and the Petri net summary statistics were as follows:

- The crews entered step 7 in procedure E-3 (start of cooldown portion of E-3) about 16.5 minutes after detecting the activity alarm at the beginning of the scenario, with a range of about 11 to 22 minutes across the 14 crews.
- During the cooldown using the atmospheric steam dump valves, three of four crews that took this path unwillingly activated the steam line protection system (which causes steam line isolation) and used extra time for the cooldown. The crews are aware of the risks connected with using the steam dump (e.g. automatic activation of SI) and are instructed to operate it with care. However, the procedure step for cooldown twice instructs the operators to cooldown at maximum speed without warning the operators about such possibilities as automatic actuation of SI.

The main observation for HFE-2A that could not be obtained from the mined models, but which could be seen in the direct simulator observations, was disruption in teamwork for those crews who had the above-noted difficulty with safety system actuation as a result of the very high cooldown rates achieved using the atmospheric steam dump valves.

For HFE-3A (failure to depressurize RCS), process mining provided the following insights:

- The crews entered step 16 in procedure E-3 (start of depressurization portion of E-3) about 28 minutes after detecting the activity alarm at the beginning of the scenario, with a range of about 21.5 to 36 minutes across the 14 crews.
- The average time to complete depressurization after entering step 16 in E-3 was about seven minutes, with a range of about 4 to 13 minutes.

For HFE-4A (SI termination), process mining provided the following insights:
• The crews terminated SI at about 29 minutes after transferring to procedure E-3, with a range of about 24 to 36 minutes across the 14 crews. This is about 38 minutes on average after the detection of the activity alarm indicating SGTR.

4.4 Conclusions from application of process mining to HRA empirical study simulator data

The Halden simulator data were supplied at a relatively high level of abstraction, as noted earlier. This allowed relatively straightforward mining of a Petri net model, by first using the heuristic miner (Weijters & van der Aalst, 2003), followed by conversion of the heuristic net to a Petri net within ProM. Had the low-level data been available, such mining would likely not have been possible, as the heuristic miner applied to such data would likely produce an extremely detailed (albeit accurate) spaghetti model, which would have been of little use to an analyst.

A salient result from the analysis is a quantitative time comparison of process variations among the 14 crews participating in the experiment. Times at which critical procedural steps occur are readily available from analysis of the Petri net model produced by ProM. However, such quantitative results can be obtained without the tools of process mining, via traditional analysis of the data in the simulator log files.

Trace clustering provides some insights as to high-level similarities and differences among the 14 crews participating in the simulator exercises. However, its overall usefulness for HRA task analysis is less clear.

Converting the heuristic net to a fuzzy model proved quite useful, as the animation tool in ProM allows the progress of each crew through the fuzzy model to be followed easily, clearly highlighting differences in the underlying process governing each crew’s performance.

Some of the difficulties encountered in processing the data for the first case study will likely be encountered in future applications of process mining to simulator data, as there is significant variety in how simulator data are recorded, and the general issue of data conversion could present a significant hurdle to the application of process mining. In summary, the principal difficulties were

• Multiple log files in Excel format with a mixture of delimiters,
• Problems with timestamp formatting,
• Logs in Swedish required translation to English,
• Logs at extremely low level of detail (e.g., each time operator touched screen was recorded).

To ease these difficulties, the analyst should make every attempt to obtain log files in .xml format, which can be converted much more easily to .mxml. The most critical logs are those containing the operator actions. Without these logs, process mining to discover the implicit process followed by the operators is impossible. Annunciator logs are also useful, but somewhat less critical. Process variable logs are very helpful, but at the moment cannot be directly incorporated into the mined models. Finally, the simulator status logs, which show points at
which the simulation was frozen and actions by the analysts running the experiment are the least useful, and can typically be dispensed with.

The analyst should attempt to determine in advance what information is logged in the simulator, and in what format (and in what language!). If possible, the level of detail that is logged should be commensurate with the goals of the analysis that the mined models will support. Logs at an extremely fine detail will likely require substantial preprocessing and filtering, steps that will be described in more detail in Ch. 6.

In the next chapter, we introduce the new hybrid HRA method being developed under sponsorship of the U.S. NRC. We will then examine whether process mining tools can provide insights that can aid in the construction of the crew response models being explored for use in this method.
5 Overview of new hybrid HRA method\textsuperscript{7,8}

The following is adapted from (Hendrickson, et al., 2010). In a Staff Requirements Memorandum (SRM) to the Advisory Committee on Reactor Safeguards (ACRS), the US Nuclear Regulatory Commission (NRC) Commissioners directed the ACRS to “work with the staff and external stakeholders to evaluate the different human reliability models in an effort to propose a single model for the agency to use or guidance on which model(s) should be used in specific circumstances.” As a first step toward meeting this directive, an effort has been undertaken to develop a comprehensive approach for qualitative human reliability analysis (HRA). Current HRA methods in use as well as psychological and cognitive theories were referenced in the development of this new hybrid approach.

The qualitative analysis under investigation in this effort is a three-stage process. The first stage is the construction of crew response trees (CRTs). These CRTs resemble event trees and describe the evolution of the scenario and human failures (potential paths) through the procedures (e.g., emergency operating procedures [EOPs] and alarm response procedures [ARPs]). Instances of possible human failures are identified within these CRTs, and these instances are then modeled in more detail to identify underlying proximate failure causes, based upon a psychological model of human information processing. The proximate failure causes then link performance shaping factors (PSFs) to the possible human failures identified within the CRTs. Therefore, the proximate failure causes represent a middle layer in the qualitative analysis approach, with the CRTs representing a top layer, and the PSFs representing the bottom layer. The mid-layer linkage is an important component as it ensures that the correct PSFs (and scenario context) are identified for quantifying the probability of the possible human failures. The final step is the identification of relevant (PSFs), which influence the likelihoods of the identified failure mechanisms.

5.1 Underlying human performance model

As shown in Figure 36, the individual operators, the crew as a whole, and the plant interact dynamically within a physical and organizational environment. There are two options for modeling the crew in HRA. First, the crew itself can be taken as the “unit of analysis,” and the model of crew response can be predicated on this assumption. Second, one could focus on the individual operators making up the crew, and model interactions among the operators. The approach for the new hybrid method is the first one, as it simplifies the modeling of many of the interface issues between the plant and the operators, and among the operators themselves. This obviously entails approximations to and abstractions from the more complex reality. From a process mining perspective, this means that details such as who originates a task may not be needed.

\textsuperscript{7} The author would like to acknowledge the contributions to this chapter of the research team members involved in developing the new hybrid HRA method.

\textsuperscript{8} The material in this chapter discusses ongoing research work carried out under the auspices of the U. S. Nuclear Regulatory Commission. The information presented does not currently represent an agreed-upon NRC staff position. The NRC has neither approved nor disapproved its technical content.
Some of the key characteristics of control rooms (especially nuclear power plant control rooms) that need to be considered in HRA are the following. First, the control room is information rich, and the operators have to filter some of the available information in order to focus their attention on important details. Second, the process that governs operator performance during most conditions encountered in operating the plant and responding to upset conditions is highly proceduralized; actions of the operators are governed by written procedures (see App. E for an overview of the procedural structure). Third, the environment is highly regulated, and operator actions are expected to comply with the conditions of the plant’s operating license, as well as with various emergency response plans, quality assurance requirements, and organizational expectations. Fourth, there are sometimes fairly tight time constraints, and thus some actions may have to be performed quickly under conditions of high workload. Finally, the typical control room crew comprises a number of individuals; a typical nuclear plant control room crew is made up of a shift supervisor, shift technical advisor, senior reactor operator, two or more reactor operators, and several auxiliary operators.

Despite the highly proceduralized and regulated nature of the control room environment, the operating crew is not blindly adhering to procedures in responding to an upset condition. Rather, the operators are trained to use the procedures as a guide in helping them to provide essential safety functions threatened by the upset condition to which they are responding. The intent of these safety functions is to place the plant in a safe, stable state, with prevention or mitigation of undesired offsite releases. However, written procedures cannot cover all of the conditions that might arise in a complex facility, especially one as complex as a commercial nuclear power plant. Thus, the control room environment is more properly seen as a goal-oriented one, in which the operators’ goals are the provision of the required safety functions (e.g., inventory control, heat control, etc.).

Figure 36 Illustration of embedding of crew in environment defined by plant and organizational factors
removal, offsite release prevention and mitigation), with the written procedures offering guidance for how to achieve these goals.

5.2 Proposed analysis approach

The proposed approach to modeling in the new hybrid HRA method has two elements: 1) a process-oriented event tree (the CRT) that focuses on errors related to interactions between the crew and the plant, and 2) midlayer models, linked to the CRT, which connect observable proximate failure causes in the psychological literature to PSFs that influence the likelihood of those failure causes.

The CRT is a forward-branching event tree model of the procedural flow, along with operator (i.e., crew) cognitive activities and actions. The relationship between the CRT and a typical PRA event tree used to develop accident scenarios is shown in Figure 37. The PRA event tree is shown above the time arrow. The nodes or branching points in the CRT can include operator decisions and actions, as well as key plant functional states that serve to define the context for the HFE in the PRA event tree to which the CRT is notionally linked.

Depending on whether the unit of analysis is the crew or individual operators, the “branch points” of the CRT can include (1) operator action options, (2) operator decision options, (3) crew member interactions, and (4) key plant functional states. Both the PRA event tree and the CRT are synchronized (symbolized by the green time arrow in Figure 37), but their dynamic nature is largely implicit. Both start with a PRA initiating event (for full power applications). The CRT is intended to be developed by an interdisciplinary team of PRA and HRA analysts, as this development requires knowledge of both plant behavior and human response to that behavior.

The CRT is intended mainly to be an analysis aid that helps ensure systematic coverage of crew-plant interactions, consistent with the scope of the analysis, and which provides traceability for the analysis. Thus, the CRT identifies the opportunities for and types of HFEs in the context of an accident scenario. Note that event trees play essentially this same role in traditional PRA, but their level of resolution is usually not adequate to be a representative model for the HRA task analysis. In contrast, the CRT is envisioned as being an abstract graphical representation of the task analysis, much as were the HRA event trees used by THERP (see Ch. 2). Each sequence of events in the CRT is a graphical representation of a possible crew response across the entire accident sequence. It is hoped that this formalism can increase consistency and reduce variability (or at least make the sources of the variability more transparent) in the HRA task analysis, and provide a common language for different HRA analysis styles, whether they are PSF-based, as is THERP (Swain & Guttman, 1983) and SPAR-H (Gertman et al., 2005), or based more on narratives, such as in ATHEANA (U. S. Nuclear Regulatory Commission, 2000), (Forester et al., 2007).
The CRT branches are mainly at the functional level and do not typically cover the underlying mechanisms of human failure and their associated causes. Therefore, the CRT structure captures only some of the contextual factors and causes of operator responses. In the proposed hybrid HRA method, the remaining aspects of context are captured in a set of supporting midlayer models of crew behavior, which may be linked to the CRT branches as appropriate, very much in analogy with the modeling division between event trees and fault trees in traditional PRA. A notional example of this linking is illustrated in Figure 38.

The midlayer models are linked in turn to a set of PSFs, which influence the likelihood of the human failure mechanisms in the midlayer model. The overall model has three layers, as shown in Figure 39.
A preliminary set of proximate failure causes has been identified for use within the qualitative analysis. These failure causes are linked to the possible human failures identified within the CRTs using a model of human information processing that is based on the IDA cognitive model (Smidts et al., 1997). The IDA cognitive model represents a three-stage model originally developed to model the response of nuclear power plant operators in an emergency situation. The stages of the IDA cognitive model are:

1. **Information.** This stage focuses on the perception of the environment and cues presented to the operator. The information derives from sources external to the operator. Cognitive processing of the information is primarily limited to the task of perceiving the information, with limited processing of the information at this stage.

2. **Diagnosis/Decision.** This stage is internal to the operator. During this phase, the operator uses information perceived in the previous stage (I), along with stored memories, knowledge, and experience to develop an understanding of and a mental model of the situation. Following this situational assessment, the operator engages in decision-making strategies to plan the appropriate course of action. Operators may use external resources such as procedures to assist them in both parts of this stage.

3. **Action.** In this final stage, the operator puts the decided-upon course of action into play.

Error is defined in terms of the operator failing to meet a plant need, which is typically related to a required safety function. The focus is on the safety impact of operator actions, some of which may be identical to HFEs defined in the PRA (as top events in the event tree or as basic events in fault trees.

The IDA model accords well with the information processing paradigm commonly used in cognitive psychology and human factors. Information processing theory outlines how information from the environment is sensed and perceived (corresponding to the “I” phase of IDA), used for decision making (corresponding to the “D” phase of IDA), and translated into behavior (corresponding to the “A” phase of IDA).
Using such an information processing model, HFEs (errors defined based on mismatch between the operator’s action and a plant need) can be traced through the I, D, A chain. An error could therefore be rooted in (1) action execution failure (A) given correct decision; (2) failure in situation assessment, problem-solving, or decision-making, given correct information (D); or (3) failure in the information-gathering stage. In this view, failures of I, D, or A are “minimal cut sets” for the HFEs.

The failure mechanisms provide the means by which the PSFs are connected to the HFE. Where possible, this link between failure mechanism and PSF is being informed by the psychological literature and by expert inference where this is not possible.

5.2.1 Example

A typical PRA event tree for a steam generator tube rupture (SGTR) initiating event is shown in Figure 40. Following the SGTR initiator, the secondary system is required to function as a heat sink for the radioactive decay heat generated in the reactor core, given successful reactor trip to stop the fission process and safety injection to replenish the primary coolant lost through the ruptured steam generator (SG) tube. Auxiliary feedwater (AFW) is the system designed to provide this heat sink. If AFW fails, main feedwater (MFW) may have the potential to be returned to service by the operators as a back-up heat sink. If both AFW and MFW fail, the only means of removing heat from the core is for the operators to open the pressurizer power-operated relief valves (PORV) and enter “feed-and-bleed” cooling.
Figure 40  PRA event tree for SGTR initiating event
Figure 41 Example CRT representing HRA task analysis for SGTR initiator shown in Figure 40

The accompanying CRT is shown in Figure 41. A detailed review of procedures is the first step in constructing the CRT. Ch. 6 will consider how simulator observations may alter the initial CRT structure from that developed solely by consideration of procedures. The following discusses how the procedures were used to identify the CRT branches, with the numbers referring to the designated branches in Figure 41. Following the signal that initiates safety injection (SI), the AFW system may actuate automatically; however, there is a probability that automatic actuation of AFW may fail. Thus, the first branch in the CRT is identified as “AFW Auto Start.”

The failure of the AFW system to start automatically may be caused by a problem with the actuation signal (the operator can back up this failure by starting AFW manually in the event of automatic failure), or there may be a hardware failure of AFW that prevents it from being started manually. Therefore, the second branch point on the CRT represents physical failure of AFW.

If there is a failure of AFW to start automatically, the emergency procedure directs the operator to check the status of the AFW pumps, and a later step in the same procedure directs him to check the flow rate being provided by the AFW pumps to the SGs. If there is a signal failure, the operator may start the AFW system manually (third CRT branch point). Otherwise, the operator is supposed to transition to another procedure that is intended to restore a secondary heat sink. The transition to this procedure (designated FR-H.1) is the fourth branch point on the CRT.

If the operator does not enter procedure FR-H.1, another crewmember may catch this error and return the sequence to a success path through the CRT. This is shown as the fifth branch point on the CRT. If the operator transitions to procedure FR-H.1, it may be the case that MFW is
physically failed (branch point six), or the operator may fail to restore MFW (branch point seven). If AFW and MFW are both failed, procedure FR-H.1 directs the operator to initiate feed-and-bleed cooling (branch point eight).

Returning to the success path from the first branch point (AFW auto start), the operator may turn off AFW (branch point nine) if he thinks that SG level is too high. As with the fifth branch point, there is an opportunity for recovery, and this is the tenth branch point.

Once in procedure FR-H.1, the operator may leave this procedure (branch point 11) if he assesses the plant condition incorrectly during the first step of that procedure. Step two of FR-H.1 directs the operator to restore AFW (branch point 12). This branch is conditional upon the availability of the AFW system (hardware and actuation signal); therefore, if the operator fails at this task, the analysis assumes that he will be unlikely to succeed in more complicated tasks, and the failure path ends in core damage, which is a modeling simplification.

5.3 Relationship of human performance model to HRA task analysis

As stated above the CRT is an abstract graphical representation of the HRA task analysis. It embodies the integrated picture of the plant-crew interactions. Plant operating and emergency procedures are used as a guide to develop the nominal scenarios represented in the CRT. Decision points and action opportunities in the procedures are the basis for branch points in the CRT, with the CRT branches determining “modes” of operator response (e.g., failure or success) from a “functional” point of view (e.g., failure to close a valve).

This functional response mode is defined at a level consistent with the level of detail in the PRA event trees and fault trees, but the operator failures in the CRT are not necessarily identical to the HFEs in the PRA event trees and fault trees. The main steps in CRT construction are as follows.

1. Identify the accident sequence initiating event.
2. Define safety functions required in response to the initiating event.
3. Delineate the accident sequences
4. Construct a CRT for each identified safety function.

The fourth step is the nexus to the HRA task analysis, and the point at which simulator observations can play a crucial role. Procedures have to be reviewed step by step in a detailed task analysis process. At each procedural step, decisions and actions have to be considered, or anticipated, and each such decision or action that might occur is a candidate for inclusion in the CRT. Items that might be included in the CRT are the following:

- Actions the operators must take to ensure the safe functioning of plant systems. These constitute the success path in the CRT
- Actions and decisions that may compromise the safe functioning of plant systems; these are the CRT failure paths and include
  - Human-caused failures,
  - Hardware failure or unavailability that triggers human-caused failures,
  - Procedural cautions, which serve as a useful information source for human- and hardware-caused failures.
- Opportunities to branch to another procedure
- Other paths observed from operations experience, including simulator studies used as inputs to the task analysis.

The potential for process mining to support CRT construction lies most obviously in the last two bulleted items, whose goal is identifying variations in the process that is loosely governed by the operating and emergency procedures. As discussed in Ch. 1, deviations from expected performance are an important ingredient in analyzing complex commission errors, which are a focus of recently developed HRA methods such as ATHEANA (U. S. Nuclear Regulatory Commission, 2000) and (Forester et al., 2007). Tools are needed to aid in analyzing large amounts of simulator data efficiently and in deriving the information required for the task analysis from the simulator data. The process mining tools that were discussed in Ch. 3 and 4 have the potential to be of use for this purpose.

In the next chapter we will apply the tools of process mining to simulator data collected in a recent exercise at a U. S. plant, with the goal being to explore how these tools can be used to support the HRA task analysis, and construction of the CRTs proposed for use in the new hybrid HRA method described in this chapter.

**Research question**

*Can the tools of process mining enhance the development of CRTs in the new hybrid HRA method, specifically by identifying additional structure in the CRTs that might otherwise not be postulated by analysts using traditional task analysis approaches?*
6 Extending the international HRA empirical study 9,10

The HRA Empirical Study discussed in the previous chapters, and in more detail in (Lois, et al., 2008), comprised two parallel sets of activities. After simple and complex cases for SGTR and LOFW scenarios were developed, actual nuclear plant crews were run through the scenarios and their performance catalogued. There were 14 European crews participating in this study, which is a relatively large number of crews; a typical nuclear plant would have a complement of four or five crews. However, there is some uncertainty about the performance of European crews in the Halden study compared with how U.S. crews might have performed. Some HRA teams involved in the study indicated that there were some actions performed that might not be expected of U.S. crews. Furthermore, the HRA Empirical Study did not capture many details about the process of conducting an HRA. While the outputs of the HRA methods used in the HRA Empirical Study were documented, the process involved with using particular methods was not documented systematically. A goal for future simulator studies is that they should specifically record details about how the HRA task analysis was conducted by each team in order to gauge commonalities, identify potential efficiencies or inefficiencies, and determine the general usability of various HRA methods.

It was also not possible for analysis teams to observe or interact with the actual crews taking part in the experiment; thus, the HRA Empirical Study did not fully embody industry good practices for HRA. It is standard practice for human reliability analysts to observe and interact with crews in order to gain information that is relevant to conducting an analysis, as discussed by (Swain & Guttman, 1983), (Ryan, 1988), and (Kolaczkowski et al., 2005).

To attempt to answer the question about U.S. crew performance, and to address the limitations of the HRA Empirical Study mentioned above, a follow-on simulator benchmark exercise has been designed, employing U.S. crews at a U.S. nuclear plant simulator. The research questions and goals for this follow-on study are summarized in the next section.

6.1 Project research questions and goals of the follow-on benchmarking exercise

The project research questions for the follow-on benchmarking study are focused on specific goals of the exercise, with respect to the HRA methods under evaluation in the exercise. An asterisk has been placed next to questions for which it is anticipated that process mining may provide answers or insights. The first goal is to assess the “quality” of various HRA methods. The associated project research questions are:

1. Does the HRA method have good predictive power?
   a. Does the method identify the correct performance drivers?
   b. Does the method predict the correct operational stories?*

9 The author would like to acknowledge the contributions to this chapter of the research team members involved in the follow-on benchmarking exercise.
10 The material in this chapter discusses ongoing research work being carried out under the auspices of the U. S. Nuclear Regulatory Commission. The information presented does not currently represent an agreed-upon NRC staff position. The NRC has neither approved nor disapproved its technical content.
c. Does the method provide reasonable human error probabilities (HEPs)?
2. What observations can be made about the guidance to the analyst for a particular method, along with its traceability?
3. Does the method provide qualitative insights useful for error reduction?*
4. What are the method’s overall strengths and limitations?

The second goal is related to the HRA qualitative task analysis. The project research questions for this goal are:

1. How should qualitative analysis be done?
   a. What is the required level of detail for the qualitative scenario analysis?*
   b. What information is required by a particular method and how do HRA analysts make use of available information?
2. How do various HRA methods generate HEPs?
3. How can HRA analysts gather information about crew behavior, and what information should be collected?*

The third goal is related to potential differences in operational culture between U.S. and European crews that could have a significant influence on crew performance. The relevant project research question is:

1. How can results from the Halden Study be generalized to U.S. operation and vice versa?

6.1.1 Study methodology

The follow-on benchmarking exercise will include several HRA teams for each HRA method being evaluated, in order to be better able to identify issues related to particular HRA methods, as opposed to issues related to the analysis team itself. The HRA teams will collect and utilize information in general accordance with industry good practices (Kolaczkowski et al., 2005). This will involve walk-throughs and talk-throughs with operator trainers and observation of simulator scenarios, although not the actual scenarios upon which the HRA methods are being assessed. This latter point is important, because in actual practice, observations would be made upon the scenarios being analyzed, wherever possible to do so. Due to the relatively large number of HRA teams involved in this follow-on exercise, logistics limitations prevented observations from being made on the actual scenarios under analysis.

An improvement in this exercise in comparison with the HRA Empirical Study described in (Lois, et al., 2008) is that a representative of each HRA team will be afforded an opportunity to visit the plant. The HRA teams are provided a small information package before their plant visit, which includes descriptions of the scenarios being analyzed. There are three scenarios that will be run, and four crews will participate in each scenario. This is significantly fewer than the 14 crews involved in each scenario in the HRA Empirical Study. Thus, there will be larger uncertainties associated with the results of this exercise, particularly quantitative results. This dissertation focuses on the application of process mining to one of the scenarios, a description of which can be found in Appendix A.
6.2 Application of process mining in support of HRA task analysis

As discussed in Ch. 5, the new hybrid HRA method described in (Mosleh, et al., 2010) and (Shen et al., 2010) has proposed the use of crew response trees (CRTs) as a graphical representation of the procedural flow in the HRA task analysis. At the close of Ch.5, the question was posed of whether the tools of process mining could aid in the construction of these CRTs, specifically by helping to establish the structure of the CRTs. This section of the dissertation begins to explore that question, using simulator data collected at a U.S. plant during January 2010 for the LOFW + SGTR scenario described in Appendix A.

6.2.1 Description of simulator data

For each of the four crews that participated in this scenario, three simulator data files were available for analysis: 1) actions taken by the control room crew, such as starting pumps; 2) alarms and annunciators in the control room; 3) time histories of process parameters, such as pressures and temperatures at various locations in the plant.11 For the control room actions and annunciators, there is a timestamp associated with each logged event. The process parameters are in the form of a time stream, with each variable’s value recorded every two seconds.

The files provided from the simulator were Excel files (.xls), with all information contained in the first column of the spreadsheet; a more difficult format with which to work is hard to imagine. In order to import the action logs into ProM, ASCII text files were extracted from the Excel spreadsheets, and these files were cleaned up using a text editor with the ability to edit using regular expressions. As an example of the cleaning that was necessary before importing these files into ProM with the ProMImport tool (Guenther & van der Aalst, 2006), the raw operator action files used a mixture of separators (tabs, spaces, and commas) and there was the occasional use of quotes, which causes problems in some of the later mining. The resulting cleaned files are quite large (up to about 10 megabytes). Thus, a significant amount of file preparation is required, including the development of a Java plug-in filter for ProMImport, before process mining can begin.

The imported simulator action event logs are also very large after conversion to .mxml format, containing up to about 72,000 audit trail entries each. Many of these audit trail entries are low-level actions, such as acknowledging of annunciators, which are not at a high enough level to be of interest in the HRA task analysis, particularly the construction of CRTs. Thus, once the simulator action logs were imported into ProM, a first step was to use the filters available in ProM to remove many of these low-level audit trail entries. Figure 42 shows an excerpt from an .mxml file, showing low-level audit trail entries associated with resetting annunciators.

---

11 The actual data could not be included in this dissertation because they were obtained under a nondisclosure agreement between the author and the U.S. Nuclear Regulatory Commission. Likewise, the operating procedures are proprietary to the reactor vendor and thus cannot be reproduced herein. App. E gives an overview of the procedures.
The following sections summarize the analysis for each of the four crews individually, followed by a comparative analysis of variability across the crews. More details of the analysis can be found in Appendix A.

### 6.2.2 Process mining of data for individual crews

Process mining techniques were applied to the simulator action logs in the discovery category, from the control-flow perspective, looking at each crew’s performance individually. The overall goal of this analysis was to mine a model of crew procedural flow that could be used to support HRA task analysis generally, and CRT construction in the new hybrid HRA method (see Ch. 5) specifically. A secondary goal was to determine how process mining tools might streamline the general data reduction and analysis process for simulator experiments, which can be quite resource-intensive. As mentioned above, a first step in this application of process mining was to remove low-level actions such as alarm acknowledgement and annunciator reset from the converted simulator action logs, via the Event Log Filter in ProM. This reduced the number of audit trail entries for each crew from several tens of thousands to several hundred. Without this initial manual filtering, the resulting models were too complex, and cluttered with insignificant actions, to be useful. This is illustrated by the fuzzy model shown in Figure 43, and the excerpt from the same model shown in Figure 44, which is taken from the beginning of a fuzzy model mined from the unfiltered operator actions of Crew 1.\(^\text{12}\) This model lies at the limit of comprehensibility with respect to node significance; lowering the node significance cut-off value very slightly produces the enormously more complex spaghetti model shown in Figure 45. One can invert the frequency metrics used by ProM for filtering. This has the effect of removing

\(^{12}\) Here and elsewhere in this dissertation, certain figures will be shown which are too large for all details to be legible. In these cases, the figure is not intended to convey detail, but instead some other characteristic, typically complexity as is the case here.
high-frequency events preferentially rather than low-frequency events. Note that inverting the frequency metrics used by the fuzzy miner in ProM did not alleviate this problem.

One drawback to this manual filtering is the potential for the unintentional removal of salient events. Thus, an undesired level of subjectivity is introduced. The analyst doing the filtering must be extremely familiar with the process being modeled: the facility and how it is operated, including the procedures used for emergency operation.

Another problem with using a model such as this for HRA task analysis is the relatively large clusters that are produced. As discussed in Appendix C and (Guenther C. W., 2009), a cluster is activated in the fuzzy model whenever any of its constituent events is activated. The problem this presents for task analysis is that the events in the cluster are not necessarily related from a functional viewpoint. An expanded view of one of the clusters in the unfiltered fuzzy model for Crew 1 is shown in Figure 46. The events in this cluster include events that are functionally related, such as the two events involving actuation of a pushbutton to shut down two trains of equipment. However, other events in the cluster are functionally unrelated to one another, such as the cross-connect of AFW and the events related to closing electrical breakers on the main generator. Thus, in order to be useful for HRA task analysis, some up-front manual filtering of the simulator action logs appears likely to be necessary, with more filtering being required the lower the level of the events that are logged.

To reiterate the concern mentioned above, the relatively large amount of filtering that is required for realistic simulator log files places a burden on the analyst to ensure that the resulting models that are mined from the files have not been stripped of details essential to visualizing the underlying process followed by the operators. To prevent this, the analyst must have a clear understanding of facility operation, or must work closely with someone who does, during the filtering stage of the mining exercise.
Figure 43  Fuzzy model for Crew 1, mined without first manually filtering low-level actions. This model is at the limit of comprehensibility with respect to the node significance cutoff, with a slightly lower cutoff producing a drastically more complex model.
Figure 44 Excerpt from beginning of fuzzy model shown in Figure 43, showing low-level actions such as keyboard alarm reset and backtrack points that make interpretation of the model difficult
Figure 45 Unfiltered fuzzy model for Crew 1, with node significance cutoff value slightly less than for the model shown in Figure 43; note drastically increased complexity of model.
6.2.2.1 Analysis of Crew 1

The scenario (see Appendix A for more details of the scenario) begins with a total loss of all main feedwater flow a little over two minutes into the simulation. A fuzzy model for the actions of Crew 1 during the scenario is shown in Figure 47, with a magnified view of the early portion of the model shown in Figure 48. This model shows the trip of the reactor by the operating crew, which occurs at about 180 sec., anticipating the automatic trip that will occur when the low SG level set point is reached. Because of the lack of feed flow, SG level is falling rapidly in all four SGs. This event is the first example where process mining might alter the structure of the CRT in a significant way. Because reactor trip is an automatic safety function, an analyst who was modeling only the process established by the plant procedures, as was the case in Ch. 5, would not routinely include a branch in the CRT to represent manual action by the operators in anticipation of the automatic action. In this case, the anticipatory trip could be important to include because of the additional time it provides to initiate feed-and-bleed cooling, as indicated in Figure 50. This anticipatory trip is in accordance with the training received by the operators at this particular plant.
Figure 47  Fuzzy model for Crew 1
Figure 48  Magnified view of beginning of fuzzy model in Figure 47, showing Cluster 136, which contains the trip of the reactor

Figure 49  Expanded view of Cluster 136 in Figure 48, showing anticipatory trip of reactor
Figure 50  Time available to initiate feed-and-bleed cooling as a function of time from loss of feed to reactor trip

The fuzzy model also shows the operators’ attempts to start the feedwater booster pump and the startup feedwater pump, per the procedure. Because these pumps fail in the scenario, SG level continues to fall in all SGs.

With no flow from main feedwater to the SGs, the operators are first concerned with attempting to remove decay heat from the reactor core via AFW flow. The operators tripped the reactor within about 50 sec. of the loss of main feedwater. From the plot in Figure 50, this means the operators have approximately 30 minutes to establish feed-and-bleed cooling should AFW not be available. As discussed above, the design of this scenario is such that all AFW pumps will fail to provide flow; however, AFW pump 12 will start and indicate full flow in the control room, but will not be feeding the SG because of a recirculation valve being open inadvertently. There is no indication of this valve’s position in the control room. The plot of AFW flow shown in Figure 51 illustrates the information available to the operators in the control room, with Figure 52 showing the actual indication in the control room. Thus, according to this indicator, there is AFW flow going to SG B, while level in SG B is decreasing. The operators could be confused by the misinformation about AFW flow, and this could cause them to delay the start of feed-and-bleed cooling to remove decay heat directly from the primary system.
The process governing operator behavior at this point is the Critical Safety Function (CSF) status tree for establishing a decay heat sink, shown in Figure 53. The operators enter this tree on the far left, and the answer to the first question, regarding SG level as indicated on narrow-range (NR) instruments in the control room, will be “No,” as level in all four SGs will be very low.
This leads them to the uppermost question in Figure 53, regarding AFW flow to the SGs. Because there is indication of AFW flow to SG B, the answer to this question is ambiguous; if the operators believe the indication of AFW flow is accurate, they should answer “Yes” to this question, and this will lead them to a series of other questions related to SG pressure. Falling SG pressure could ultimately lead the crew to the lowermost yellow branch in Figure 53, which will direct them to enter procedure FR-H.5, whereas the red branch leads them to a different procedure, FR-H.1. It is this latter procedure that directs the operators to initiate feed-and-bleed cooling. *In constructing the CRT for procedural flow, an analyst might reasonably have a branch at this point to represent the ambiguity in the procedural flow.*

![Figure 53](image_url)

**Figure 53  CSF status tree for establishing decay heat sink. Operators could be on either the upper red path or the lowermost yellow path**

In this simulation of the scenario, Crew 1 took the red branch in Figure 53 to enter FR-H.1, and quickly reached Step 10, which directs them to actuate feed-and-bleed cooling.

According to procedure FR-H.1, the operators should stop the running reactor coolant pumps (RCPs) in Step 2 of that procedure. This is an important action because the RCPs add a significant amount of energy to the primary coolant, which must be removed. Stopping the RCPs eliminates this source of energy, and extends the time available to restore AFW or initiate feed-and-bleed cooling. However, the mined fuzzy model shows that Crew 1 delayed stopping the RCPs until after actuating SI in Step 10. *This event is the second example where process mining might alter the structure of the CRT in a significant way.* An analyst constructing the CRT to represent anticipated procedural flow in this scenario would likely model tripping of the RCPs at Step 2 of FR-H.1, as this is the procedural direction to the operators. It could be an important branch to include because of its potential effect on the time available for restoring AFW or initiating feed-and-bleed cooling.
By the design of this scenario, SGTR will occur in the first SG to which AFW flow is restored. Crew 1 restores flow first to SG B. Following the SGTR, the higher pressure on the primary side will cause reactor coolant to flow into the secondary side of the SG, causing secondary side level to rise. Radioactivity in the primary coolant could be detected by radiation monitors on the steam lines leaving the SG, and this is a primary symptom of SGTR, and a cue for the operators to enter procedure E-3, which is tailored for SGTR. However, in this case the steam lines were isolated prior to restoration of AFW flow, and the SG blowdown and sampling lines, which are also instrumented with radiation monitors, were isolated by the SI signal. As a result, the radiation monitors on the steam lines, blowdown lines, and sampling lines will not be available. This is a critical issue for CRT construction, and one which an analyst might overlook in the absence of detailed simulator observations.

In order to deal with the SGTR that has occurred in SG B, the operators must leave FR-H.1 and transition to procedure E-3. Step 6 of procedure FR-H.1 directs the operators to check SG levels, and if NR level is greater than 35% in one or more SGs, they should exit FR-H.1. The criterion to enter E-3 is an uncontrolled increase in any SG level or any abnormal SG radiation level. By the design of this scenario, the operators are expected to transition to E-3 based on uncontrollable level in the ruptured SG, which in this case is SG B.

The fuzzy model illustrates that the operators next reduced AFW flow in response to rising level in SG B. Once the operators have determined that an SGTR has occurred, they transition to procedure E-3, which directs them to isolate the ruptured SG. Unfortunately, the simulator did not log procedural transitions, unlike the simulator at Halden, so the exact time at which the operators transitioned to procedure E-3 cannot be ascertained from the log files. The operators isolated SG B at about 45:00.

6.2.2.2 Analysis of Crew 2

The same scenario was run for each of the four crews participating in this experiment. Therefore, the initial indications, such as main feedwater flow, were the same for each crew. The fuzzy model for Crew 2 is shown in Figure 54.
As for Crew 1, there was an anticipatory reactor trip based on decreasing SG level. The operators have indication of AFW flow to SG B. Also, they are presented with the same dilemma with respect to which procedure to enter from the CSF status tree in Figure 53. Like Crew 1, Crew 2 also chose to enter FR-H.1 from the CSF status tree. However, when Crew 2 entered procedure FR-H.1, they, in contrast to Crew 1, executed the procedure steps in the anticipated order, thus stopping the RCPs at Step 2. This is an example of where process mining has been able to illustrate crew-to-crew variability, or a deviation from the anticipated procedural flow on the part of Crew 1. Crew 2 initiated feed-and-bleed cooling at about 1,200 seconds. This is slightly later than Crew 1.

Crew 2 takes about the same overall time as Crew 1 to re-establish AFW flow. The next portion of the scenario is spent in procedure FR-P.1, attempting to re-establish a steam bubble in the pressurizer. Steam flow from SG B was isolated relatively early. However, isolating AFW flow took considerably more time. The level in the ruptured SG stabilizes, albeit at a higher level than that achieved by Crew 1. Crew 2 eventually isolates the ruptured SG, with no release of radioactivity; however, the extra time spent in procedure FR-H.1, attempting to establish a secondary heat sink so that feed-and-bleed cooling can be terminated, has resulted in a longer time to identify and isolate the ruptured SG relative to the time taken by Crew 1.

From the standpoint of constructing a CRT to represent the procedural flow in the HRA task analysis, process mining illustrates the additional complexity of the loosely controlled process followed by Crew 2 while in procedure FR-H.1. Thus, the structure of the CRT considering the performance of Crew 2 will be more complex than had an analyst considered only Crew 1, with additional branches required to represent the process followed by Crew 2.

6.2.2.3 Analysis of crew 3

The simulator action log for Crew 3 was significantly larger than for the other crews; the imported log contained over 100,000 audit trail entries, suggestive of a more complex process for this crew than for Crews 1 and 2. As before, manual filtering using the Event Log Filter in ProM was a necessary first step for mining this event log. In this case, the filtered log contained slightly more than 500 audit trail entries.

The fuzzy model for Crew 3 is shown in Figure 55. This model has a lower level of node connectivity than the corresponding models for Crews 1 and 2 shown above. This will make the model somewhat more difficult to employ in support of task analysis and simulator data reduction; however, efforts to increase the connectivity in the fuzzy miner by tuning the parameters in the fuzzy miner did not produce significantly better results; such efforts are probably not worthwhile, given the ability of the fuzzy miner animation to visualize crew response, even when the model is not fully connected. Crew 3 responded to the loss of all main feedwater by anticipating the reactor trip, very similar to Crews 1 and 2.
Figure 55  Fuzzy model for Crew 3
As was the case for Crews 1 and 2, Crew 3 has indication of AFW flow to SG B. Crew 3 transitioned to procedure FR-H.1 more quickly than Crews 1 and 2, and initiated feed-and-bleed cooling upon entry.

Following the entry into FR-H.1 and the establishment of feed-and-bleed cooling, Crew 3 worked to restore AFW flow to the SGs. Crew 3 restored AFW flow to SG C, in contrast to Crews 1 and 2, who restored flow to SG B. They also took significantly longer than Crews 1 and 2 in restoring AFW flow. By the design of the scenario, SGTR occurs in the first SG to which AFW flow is restored, SG C in this case. This causes level in SG C to begin rising.

Crew 3 also managed to isolate the ruptured SG by isolating steam flow and terminating AFW flow. They were considerably slower than Crews 1 and 2 in terminating AFW flow to the ruptured SG.

6.2.2.4 Analysis of Crew 4

The fuzzy model for Crew 4 is shown in Figure 56. Crew 4 responded to the loss of all main feedwater by anticipating the reactor trip, very similar to Crews 1-3.
As was the case for Crews 1-3, Crew 4 has indication of AFW flow to SG B. Like Crew 2, Crew 4 initiated feed-and-bleed cooling later than Crews 1 or 3.

Following entry into procedure FR-H.1, Crew 4 restored flow to SG B shortly after 1,200 sec. This precipitates a rupture in SG B, as was the case for Crews 1 and 2.
Crew 4 did not transition to procedure E-3 as quickly as Crews 1-3. Radioactivity release (simulated) occurred before the ruptured SG could be isolated, as is shown in Figure 57.

![Plot of SG pressure for Crew 4. Radioactivity is released from SG B, as indicated by cycling of pressure at the relief valve set point shortly after 2,000 sec.](image)

**Figure 57** Plot of SG pressure for Crew 4. Radioactivity is released from SG B, as indicated by cycling of pressure at the relief valve set point shortly after 2,000 sec.

### 6.2.3 Comparison of crews – process mining of combined simulator action logs

Even with the filtering of low-level actions for each of the simulator action logs described above, the combined log containing the actions of all four crews is still too large for quantitative insights to be drawn from an underlying Petri net model. Therefore, the combined log was filtered a second time to focus on the most salient operator actions from the perspective of plant safety functions, such as tripping the reactor, initiating feed-and-bleed cooling, etc. This produced a high-level log similar to the Halden log analyzed in Ch. 4. After this second round of manual filtering with the Event Log filter in ProM, there were 742 audit trail entries in the combined log. A heuristic net was mined from this log, and was converted to a Petri net for quantitative analysis. The resulting heuristic net and Petri net are too large to be displayed in their entirety. However, some important insights can be derived from Petri net performance analysis in ProM.
The first insight is the variability in the timing of reactor trip. This is important, because as Figure 50 indicates, the time available to initiate feed-and-bleed cooling depends strongly upon how quickly the reactor is tripped following the loss of feedwater. An excerpt of the mined Petri net is shown in Figure 58. The time between the loss of feedwater and the anticipatory reactor trip ranges from 34 seconds for Crew 4 to 57 seconds for Crew 3. From Figure 50, the corresponding time available to initiate feed-and-bleed cooling ranges from about 56 min. for Crew 4 to about 20 min. for Crew 3. Thus, a relatively small difference in reactor trip timing corresponds to a fairly significant difference in time available to provide heat removal via feed-and-bleed.

![Figure 58 Excerpt of Petri net mined from combined simulator action logs, illustrating variability in time between loss of feedwater and anticipatory reactor trip](image)

Other times of interest are shown in Table 5. Note that for Crews 2 and 3, the time of SGTR isolation could not be determined definitively from the mined Petri net. The reason for this is that the ruptured SG may have been isolated earlier in the scenario by Crews 2 and 3, well before reaching the isolation step in procedure E-3. For Crew 4, a release of radioactivity occurred before the SG was eventually isolated.

### Table 5 Summary of times (min. from loss of feedwater) extracted from Petri net model for combined simulator action log

<table>
<thead>
<tr>
<th>Event</th>
<th>Crew 1</th>
<th>Crew 2</th>
<th>Crew 3</th>
<th>Crew 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuation of feed-and-bleed cooling</td>
<td>11.5</td>
<td>16.7</td>
<td>10.1</td>
<td>14.8</td>
</tr>
<tr>
<td>AFW flow restoration (SGTR)</td>
<td>21.5</td>
<td>20.5</td>
<td>26.0</td>
<td>18.4</td>
</tr>
<tr>
<td>SGTR isolation</td>
<td>43.45</td>
<td>NA</td>
<td>NA</td>
<td>51.3</td>
</tr>
</tbody>
</table>

It is also possible to mine a fuzzy model from the combined data, and use the animation in ProM to make a visual comparison of the Crews. The fuzzy model is shown in Figure 59. An excerpt
of the animation shown in Figure 60 highlights an important variation in crew performance, where Crew 1 trips the RCPs after actuating SI. As noted above, this indicates a deviation of Crew 1 from the written procedure, which directs operators to trip RCPs before actuating SI. In contrast, the other three crews did trip the RCPs before actuating SI, as directed by procedure. Again, the animation of the fuzzy model makes this variation stand out, as shown by Figure 61, which is an excerpt of the animation for Crew 3, the crew who actuated SI in the shortest time after tripping the reactor, as shown in Table 5.

Figure 59  Fuzzy model mined from combined simulator action log after second filtering pass to reduce size of log
Figure 60  Excerpt from animation of fuzzy model shown in Figure 59, showing Crew 1 tripping RCPs after actuating SI

Figure 61  Excerpt from animation of fuzzy model shown in Figure 59, showing Crew 3 tripping RCPs before actuating SI
6.3 Summary implications of process mining for HRA task analysis

As discussed in Ch. 2, a facility simulator can be an invaluable tool to aid in understanding how operators might interact with complex engineered systems, especially during low-frequency, high-consequence scenarios. It is obviously impractical from any rational perspective to run experiments on actual facilities to gain these insights about operator performance in such scenarios, and so simulating the facility’s physical behaviour, while making observations on operator performance, must be used as a surrogate to melting reactor cores, crashing planes, etc.

This chapter has examined one such simulator study, involving a scenario initiated by a loss of feedwater complicated by numerous equipment failures and SGTR. Process mining has provided some useful insights into some significant variations in the implicit process that governs crew performance in this scenario. First, it is clear from examining the mined models that all crews manually tripped the reactor in advance of the automatic trip on low SG level. Therefore, in terms of structuring the initial phase of the CRT for the qualitative portion of an HRA, a branch for anticipatory manual reactor trip is needed. Additionally, there is enough variability in the time taken between the loss of feedwater that initiates the scenario and the anticipatory reactor trip to lead to significant variation in the time available for actuation of feed-and-bleed cooling, in accordance with Figure 50.

Second, it is clear from the mining that the crews also vary in how quickly they are able to restore secondary heat removal (AFW flow) after initiating feed-and-bleed cooling. This is readily seen in the Petri net performance analysis summarized in Table 5. However, it is not straightforward to determine the reasons for this variability from the fuzzy models mined from the simulator action logs. In part, this is because the simulator did not record procedure transitions. As a result, it is not always evident which procedures were being used at a particular point in time. If video of the simulation were available to couple with the mined fuzzy models, the procedure transitions would be clearer, and this would lend power to the fuzzy models as explanatory tools.

Finally, there are important variations in the process governing the period between the occurrence of the SGTR and isolation of the ruptured SG. One of the crews (Crew 4) was unable to control SG pressure during this phase of the scenario, and this resulted in (simulated) release of radioactivity via the SG relief valves, as indicated in Figure 57. As above, this appears to be because Crew 4 did not transition to procedure E-3 expeditiously, but because the procedure transitions are not indicated in the simulator logs, and no video is available, this conclusion is not certain.

Thus, the fuzzy models mined from the simulator actions logs, in conjunction with the process parameter logs, have been able to provide some important insights into how the CRT should be structured for this scenario in order to account for important variations in crew response. This work has also identified where there are limitations to the inferences that can be drawn about crew response, one of these being the omission of procedural transitions from the simulator action logs. Logging procedural transitions in future experiments, as is done at the Halden Reactor Project simulator, would be a boon to efforts to employ process mining for simulator data. Foregoing this improvement, were video of the simulation available to supplement the information in the logs, procedural transitions could be added to the fuzzy models manually. This
would significantly enhance their ability to support the HRA task analysis, and the construction of CRTs in particular.
7 Analysis and reflections

Human error has long been acknowledged as a significant contributor to the risk of complex engineered systems, with (Reason, 1990) being a cornerstone reference for this point. Human reliability analysis (HRA) has been attempting to identify, model, and quantify the human contribution to risk since the 1960s, with (Swain, 1963) being one of the earliest references to this effort. The first widely publicized HRA guidance was contained in the WASH-1400 report (U. S. Nuclear Regulatory Commission, 1975), which addressed the safety of nuclear power plants in the U. S. The Technique for Human Error-Rate Prediction (THERP) HRA method (Swain & Guttman, 1983), which evolved from the HRA performed for WASH-1400, provided the first systematic method of identifying, modeling, and quantifying human errors, and is viewed as the father of HRA methods today.

A generation later, there are literally dozens of HRA methods to choose from, good practices have been developed for HRA (Kolaczkowski et al., 2005), many of the HRA methods have been evaluated against these good practices (Forester et al., 2006), and new HRA methods are still being developed around the world. Limited studies have been performed to validate the numerous approaches available. These studies, limited though they have been, have consistently identified large variability in the analysis results, from one HRA method to another, and between analysts for a given method. Much of this variability comes down to variability in the qualitative portion of the HRA, and more specifically in the task analysis (Boring et al., 2010). As described in (Swain & Guttman, 1983) and (Kolaczkowski et al., 2005), an important ingredient of the task analysis is observations from system simulators. Because HRA focuses on low-frequency high-consequence scenarios, it is not possible to observe operator performance on an actual facility. Thus, the facility is replaced by a computer simulator that allows such scenarios to be run, and operator interactions with the simulated facility observed. These observations are important in order for the HRA team to be able to realistically model procedure implementation, interactions between the crew and the system, and interactions among the crewmembers themselves.

However, while simulator observations are acknowledged as a crucial HRA task analysis ingredient, guidance on effectively and efficiently employing the abundance of information produced by these simulator studies is lacking. More specifically, tools are lacking to aid in analyzing large amounts of simulator data efficiently and translating the data into the information required for the HRA task analysis. As discussed in Ch. 2, there are no tools in current HRA use that can aid directly with this in the task analysis, by identifying underlying models of crew performance, and highlighting deviations from expected performance, along with crew-to-crew variations.

7.1 Reflecting back on the principal research questions

The first research question was with regard to the requirements for tools that could be of use for HRA task analysis.

RQ1: What are the requirements for a tool to aid in the analysis of large amounts of simulator data in support of HRA?

Ch. 2, which examined the overall HRA process, listed the following characteristics to be considered in HRA modeling:
- Plant behavior and conditions
- Timing of events and the occurrence of human action cues
- Parameter indications used by the operators and changes in those parameters as the scenario proceeds
- Time available and locations necessary to implement the human actions
- Equipment available for use by the operators based on the sequence
- Environmental conditions under which the decision to act must be made and the actual response must be performed
- Degree of training, guidance, and procedure applicability.

The first three and the last of these are captured in simulator observations, which are an essential ingredient to a good HRA, as described also in Ch. 2. Simulators can produce very large output files, in a variety of formats. Manually analyzing such output data is very resource intensive, and has in the past limited the use of simulator experiments and observations in support of HRA. Thus, one requirement for the analysis tool is that it be capable of accepting data in a flexible format, and that it be able to handle large amounts of data, as process variable values may be recorded at millisecond intervals over a scenario lasting one hour or more. A second requirement is that the analysis cannot be purely statistical, because the HRA is concerned with the process followed by the operators, and not solely with statistical variables, such as the time at which a certain action is performed.

RQ2: How do current HRA methods approach the issue of simulator observations and are these approaches suitable for incorporating simulator observations into the HRA task analysis?

Ch. 2 examined two representative HRA methods, THERP and ATHEANA, both of which are considered complete methods, in that they address all three aspects of the HRA: identification, modeling, and quantification of HFES (Kolaczkowski et al., 2005). As noted in Ch. 2, most HRA methods address only HFE quantification, and thus are much more limited than THERP and ATHEANA. Both THERP and ATHEANA discuss the need to incorporate simulator observations into the HRA. However, neither method addresses how this is to be done. Ch. 2 also examined the TALENT approach, which was intended to constrain the variability observed in applying THERP (TALENT was developed before ATHEANA), by providing guidance for performing the HRA task analysis. However, TALENT does not address the question of how to incorporate simulator data into the task analysis. This led to the conclusion that there are no extant tools in the HRA community of practice that are suitable for analyzing large amounts of simulator data in the context of an HRA task analysis.

RQ3: Are there tools in other domains that are more suitable and which, if adopted (and adapted to their new domain), could improve the state of the art in HRA modeling and task analysis?

Ch. 3, which provided an overview of business process mining tools, along with some selected industrial applications of these tools, showed that these tools have potential for application in support of HRA task analysis specifically, and simulator data analysis more generally. Tools developed for statistical data mining are of very limited use for this purpose, because they are not
concerned with the underlying process. Business process mining appears to be the only community to have addressed data mining from a process rather than a statistical perspective.

Based on the industrial applications of process mining reviewed in Ch. 3, there was promise that simulator event logs could be mined from the control-flow perspective to provide information on the underlying process describing operator behavior, information that could be used to support existing approaches to HRA task analysis, as described in Ch. 2, or techniques such as the CRT proposed for use in the new hybrid HRA method described in Ch. 5, which remains under development at the completion of this dissertation.

RQ4: What are the limits of applicability of these tools from other domains, and what improvements are needed in order to make them practical for use by an analyst who is not a specialist in using such tools?

To begin answering this question, Ch. 3 examined some of the process mining tools and techniques in the context of analyzing simulator data. The most promising techniques appeared to be those developed by (Guenther C. W., 2009) for mining flexible processes, in particular the fuzzy model, which is described in more detail in App. C. Because simulator log files are typically very large, traditional process mining approaches can be expected to produce an overly complex “spaghetti model” that would be quite opaque to analysis. The fuzzy model abstracts away the irrelevant details, leaving the salient aspects of interest for the task analysis.

Ch. 4 began exploring how the tools of process mining might be applied to simulator data. It summarized the process of gathering and processing the data from the simulator lab in Halden, and described the difficulties encountered. Because of these difficulties, the examination of the process mining tools began with a relatively small set of logs from the Halden empirical study described in (Lois, et al., 2008). These logs, which were provided in Microsoft Excel format, had to be first converted to .csv format, and then to the .mxml format required by ProM. This file format conversion proved to be the first challenge, as a conversion plug-in for ProMImport had to be written in Java. A number of difficulties were encountered during this conversion process, among them issues with timestamp formatting, inclusion of quotes in the event description field, and a mixture of characters used to delimit the fields in the data. These problems became even more severe in the application of Ch. 6, which involved much larger event logs.

Even for the relatively small event logs examined in Ch. 4, traditional process mining approaches, such as the alpha-algorithm (de Medeiros & van Dongen, 2004) for mining a Petri net, produced an overly complex model that would be of little use for task analysis. The heuristic miner (Weijters & van der Aalst, 2003) produced a model that was significantly less complex, and showed some features that are immediately useful for HRA task analysis, such as alternative paths through the operating procedures taken by some of the crews who participated in the study. Converting the heuristic net to a fuzzy model within ProM allowed the fuzzy model to be animated to visualize the flow through the process model for each crew. The heuristic net can also be converted to a Petri net in ProM to allow more quantitative analysis.

Ch. 4 concluded that certain process mining tools, especially the fuzzy miner, appeared to be of use in support of task analysis, because they could clearly highlight differences in the underlying process governing each crew’s performance. The heuristic miner proved to be useful as long as the event logs are relatively small, as they were for the analysis in Ch. 4. Ch. 6, which examined
much larger and more typically sized event logs, concluded that the heuristic miner was of less practical use for logs of that size. Ch. 4 found that trace clustering, another technique described by (Guenther C. W., 2009) for analyzing flexible processes, provided some insights as to high-level similarities and differences among the 14 crews participating in the Halden simulator exercises. The trace-clustering technique may be helpful for follow-on simulator experiments involving large numbers of crews. However, its overall usefulness for HRA task analysis may be somewhat limited, as the number of crews in most facilities is much smaller than the 14 crews involved in the Halden exercise. For example, there were only four crews participating in the follow-on exercise described in Ch. 6, so trace clustering and related tools described in (Guenther C. W., 2009) were not explored further.

Ch. 6 examined the application of these process mining tools to simulator data collected at a U.S. plant during a follow-on to the Halden empirical study. Conversion of simulator logs to the .mxml format required by ProM proved to be a particularly severe problem to overcome. The files provided from the simulator were Excel files (.xls). ASCII text files were created from the Excel spread sheets, and this was a nontrivial problem. The resulting files had to be cleaned up using a sophisticated text editor with the ability to edit using regular expressions. Thus, a significant amount of file preparation was required, including the development of a Java plug-in filter for ProMImport, before process mining could begin.

Such up-front file preparation presents the opportunity for errors to be introduced into the resulting mined models. Much of this preparation would be obviated if the log files were output from the simulator in .xml format, thus allowing easy conversion to the closely related .mxml format used by ProM. The process mining community has recently been working on making the tools more useful to the nonspecialist, and a conversion package called Nitro that has the potential to reduce the conversion burden significantly is now being marketed by Fluxicon (http://fluxicon.com/nitro/).

The fuzzy miner proved to be the tool of choice in ProM for mining the large event logs of Ch. 6. However, the imported simulator action event logs were very large after conversion to .mxml format, containing up to about 72,000 audit trail entries each. The resulting fuzzy models were not useful, as they contained extremely large clusters, and these clusters contained both relevant and irrelevant actions. Lowering the node significance cut-off to reduce the cluster size produced models that were still too complex, and too cluttered with insignificant actions, to be useful. Many of these audit trail entries were low-level actions, which are not of high enough level to be of interest in the HRA task analysis; see Figure 42 for an example. Thus, in order for tools such as the fuzzy miner to be useful for HRA task analysis, some considerable up-front manual filtering of the simulator action logs appears likely to be necessary, with more filtering being required the lower the level of the events that are logged. Such filtering has the potential to introduce errors into the resulting mined models, and so the analyst who does the filtering must have detailed knowledge of facility procedures and operations, or have access to someone who does, to ensure that such errors are not introduced.

Applying the fuzzy miner to the filtered logs provided some especially useful insights for HRA task analysis, and particularly for construction of the crew response trees (CRTs) being considered for use in the new hybrid HRA method described in Ch. 5. The first such insight was that branches in the CRT are needed to model actions taken by the operating crew that anticipate
automatic plant safety functions, such as reactor trip. A second such insight was the variability in ordering of significant procedural steps, such as not stopping reactor coolant pumps (RCPs) before initiating feed-and-bleed cooling. In the particular scenario considered in Ch. 6, anticipating the automatic reactor trip lowers the heat load substantially, and provides significantly more time for the operators to initiate feed-and-bleed cooling, while not stopping the RCPs at the proper point in the procedure increases the heat load (the RCPs are very large motor-driven pumps), lessening the time available.

It should be noted that, while errors of commission are explicitly evident in the mined models, errors of omission are not; the analyst must identify such errors, based on his knowledge of the facility and its operation. If logs from more than one crew are available, comparing the crew models with the animation tool in the fuzzy miner can be an aid in identifying such errors, as it will be evident if one crew performs an action that another crew omits.

Identification of variations generally is possible because process mining provides a graphical illustration of the crew-to-crew process variability. From the standpoint of constructing a CRT to represent the procedural flow in the HRA task analysis, the process mining in Ch. 6 illustrates the additional complexity of the loosely controlled process followed by Crew 2 while in procedure FR-H.1. Thus, the structure of a CRT that is developed considering the performance of Crew 2 will be more complex than had an analyst considered only Crew 1, with additional branches required to represent the more complex process followed by Crew 2.

Mining the simulator action log for all four crews combined provided some additional comparative insights. However, even with the filtering of low-level actions for each of the simulator action logs described above, the combined log was still too large for quantitative insights to be drawn from an underlying Petri net model. Therefore, the combined log was filtered a second time to focus on the most salient operator actions from the perspective of plant safety functions, such as tripping the reactor, initiating feed-and-bleed cooling, etc. A heuristic net was mined from this log, and was converted to a Petri net for quantitative analysis.

One insight from the Petri net model was the variability in the timing of reactor trip. The time between the loss of feedwater and the anticipatory reactor trip ranges from 34 seconds for Crew 4 to 57 seconds for Crew 3. The corresponding time available to initiate feed-and-bleed cooling ranges from about 56 min. for Crew 4 to about 20 min. for Crew 3. Significant variations were also observed in the time taken to initiate feed-and-bleed cooling, to restore AFW flow, and to isolate the ruptured SG, with one crew failing to isolate the SG before releasing (simulated) radioactivity to the environment through the SG pressure relief valve. Animation of the fuzzy model for the combined crew data was extremely helpful in highlighting significant process variations.

7.2 Methodological insights

As discussed in Ch. 3, this research has not attempted to be a systematic and exhaustive examination of process mining tools for application to HRA. Rather, selected techniques have been explored that appeared to be most immediately applicable to simulator event logs, and several approaches have been identified that appear to be promising for providing useful input to the qualitative HRA. The fuzzy model developed in (Guenther C. W., 2009) appears to hold the
most promise, as it has the ability to produce tractable models from which insights can be gleaned, even with quite large datasets.

7.2.1 Simulator data requirements

Three sets of information need to be captured in the simulator logs. These are the operator actions, the annunciator logs, and time histories of plant parameters upon which operators base their decision-making, using the procedural structure described in App. E.

Operator actions

The important actions to collect are those involving procedure use, including transitions among procedures, as multiple procedures may be used in parallel, and interactions with plant components, such as pumps and valves. Caution should be used to avoid collecting data on operator interactions with the plant at a level of detail that is finer than needed to support the HRA. Examples of this encountered in the case study include operator acknowledgment of alarms and adjustments to pump flow speed. Including such data drastically increases the size of the log files, to the point where significant manual filtering may be required. Such filtering relies on analyst expertise to avoid filtering out salient actions, and thus introduces an undesirable element of subjectivity into the analysis. To avoid this, it is recommended that the analyst interact with the simulator personnel ahead of time to ensure that only relevant actions are logged, if this is possible.

Annunciator logs

It is crucial to ensure that annunciators that provide important procedural cues to the operators are logged. Generally, most simulators will log all annunciators, but the analyst should ensure that this occurs.

Process parameters

Process parameters such as pressure, temperature, and level at various points in the plant are important procedural cues to which the operators respond, so it is crucial that information about such parameters be logged. In this dissertation, plots of these parameters were made outside the process mining tool, as the tool does not allow such parameters to be linked directly to the mined models. Such a feature would be a significant benefit to future applications.

7.2.2 Removing operator actions from the model

It is possible that certain operator actions that are of little interest will be included in the simulator logs. These can be dealt with in two ways. In the fuzzy mining tool within ProM, they can be aggregated into clusters or abstracted from the model entirely. These actions, which are a main attractive feature of the fuzzy miner, may be sufficient to produce a tractable model showing the salient operator actions.

The primary means of controlling the level of aggregation and abstraction is via the Node filter in the fuzzy miner. The analyst will have to experiment with settings of the slider to find the ideal level of clustering. It is also possible to control the level of abstraction and aggregation by
adjusting the metrics used by the fuzzy miner. However, this approach is more indirect and does not provide the “interactive” capability of using the Node filter slider.

The second approach to dealing with these low-level actions is to manually filter them using the log-filtering capabilities in ProM. In practice, both approaches may be necessary. In both cases, the analyst must exercise caution to ensure that salient actions are not removed from the resulting model.

7.3 Contributions and limitations

This research has shown that there is promise in applying the tools developed by the business process mining community in support of HRA task analysis. This represents an advance in HRA task analysis, which benefits from simulator observations, as discussed in Ch. 2.

The principal promise of process mining is in illustrating important process differences and crew-to-crew variations that need to be taken into account in the HRA task analysis. For the new hybrid HRA method described in Ch. 5, process mining has been useful in clearly identifying CRT branch points, such as anticipation by the operating crew of automatic safety functions. It has also identified crew-to-crew variations in how procedures are implemented, especially variations in the order in which procedural steps are carried out.

The research has also highlighted some limitations of process mining that will need to be addressed before the tools can be applied by analysts in the field who are not process mining specialists. First, there is the issue of data conversion. Simulator logs are generated in a variety of formats, with Microsoft Excel appearing to be a common one. A .csv file can be generated from the Excel file, although this is not always straightforward. The .csv file may require substantial cleaning before it is ready for conversion to the .mxml format required by ProM. In the conversion to .mxml a number of issues arose that required special handling, such as timestamp formatting variations and the use of multiple delimiters. While the ProMImport tool has a number of conversion plug-ins, including one for .csv files, dedicated Java plug-ins still had to be developed. An HRA analyst cannot generally be expected to write Java programs for data conversion. Ongoing work to develop conversion tools, such as the Nitro tool by Fluxicon mentioned above (www.fluxicon.com/nitro), should alleviate the problem of file conversion.

Second, with very large simulator logs, substantial manual filtering of the logs using the Event Log filter in ProM is needed before useful models can be mined, even with the fuzzy miner. This seems to be a result of the large range of tasks captured in the simulator logs, with actions that are very salient to the task analysis being logged along with low-level actions of essentially no interest. There are tools built into ProM that can ameliorate this problem (e.g., filtering based on frequency of the events in the log), but manual filtering was still needed to produce reliable logs for mining. This requires the analyst to have detailed knowledge of facility procedures and operations to avoid introducing errors into the resulting mined models.

A third limitation is related to the event logs themselves. The names given to the actions in the simulator logs may not be clear to the analyst (they may be entirely cryptic), and thus the mined model will also suffer from this problem. The analyst may need to use the replacement filters available in ProM to substitute meaningful names for these kinds of events.
Taken together, these first three limitations indicate that a substantial amount of file preparation and log filtering may be required before useful process models can be mined from simulator logs. The file format limitation is most readily overcome by outputting the simulator log files in .xml format. If this is not possible, tools such as Nitro (www.fluxicon.com/nitro), which are being developed, should accelerate the file format conversion. It might be possible to alleviate the second limitation by controlling the actions that are output in the simulator log files, outputting only the actions of concern to the HRA, and omitting low-level actions of no interest.

As Ch. 6 made clear, it is the insights provided by the combination of mined process models and process variable time histories that are of most use for task analysis. Coupling the process variable logs to the mined model would be extremely helpful in this respect. In addition, logging procedural transitions would be enormously beneficial, as it can sometimes be difficult to infer from the operator actions alone precisely when a procedural transition occurred. Video would also be helpful in evaluating interactions among the crewmembers, along with assessing certain performance shaping factors of concern to the HRA, such as stress.

Finally, while ProM is a powerful research tool, parts of it could benefit from enhancements to make it more usable by HRA analysts in the field, who are not process mining specialists. Some suggestions along these lines are listed below, not necessarily in order of priority.

1. Develop a powerful, flexible, easy-to-use converter for files in .csv format. Allow for multiple delimiter types. Note that this capability does not have to reside in ProM itself, but may be provided by third-party tools such as Nitro (www.fluxicon.com/nitro), discussed above.

2. Enhance the fuzzy mining tool in ProM by allowing the user to manually control the abstraction in a more straightforward way than is currently available. As examples, it would be helpful for the analyst to be able to specify events that either should be or cannot be abstracted away, or to select events to be added to or removed from a cluster. It would be ideal if the analyst could do this by selecting events graphically in the Transformer interface of the fuzzy mining tool.

3. Enhance the animation of a fuzzy model by allowing the user to “see inside” clusters.

7.4 Future research

This dissertation has succeeded in demonstrating the potential usefulness of certain process mining tools for supporting HRA task analysis and simulator data analysis more generally. However, more work is needed to demonstrate the actual usefulness of these tools in the field. The main elements of the work needed to make the current process mining tools practical for use in the field have been outlined above, and are more in the nature of further development of existing tools and interfaces than in new research.

While this dissertation has focused on nuclear plant simulators, it is obvious that process mining could be applied to the analysis of data from simulators of other types of facilities, such as those used in the fields of aviation, marine transport, and medicine. HRA in these non-nuclear domains, with its requisite for a task analysis, could potentially benefit from process mining of simulator data.
In the vein of new research, several directions have been made apparent by the work in this dissertation. First, it would be tremendously beneficial to have information on the process variables incorporated directly into the mined model, as data attributes, for example. This would allow the analyst, via a time-dependent process view, to associate values of variables such as level or pressure more directly with operator actions in the mined model. As described in Ch. 6, the combination of mined models and information regarding salient process variables is significantly more useful than either would be alone in supporting the HRA task analysis. Inclusion of the process variable histories could also be used to explore decision mining as an aid in determining why particular operator actions occurred.

A second area of research is the use of colored Petri nets (Jensen & Kristensen, 2009) to simulate the performance of future crews. In addition to the structure of the mined model, timing information, which is indicative of the inherent variability in crew performance, could be used to infer probability distributions, which could then be used in simulation to predict human error probabilities for certain actions.

A third potential research direction involves the potential to employ process mining in post-processing of data produced by dynamic simulation models being developed in the risk analysis community. Approaches such as the Advanced Dynamic Simulator (ADS) described in (Hueh & Mosleh, 1993), (Hsueh & Mosleh, 1996), and (Hsueh et al., 1993) produce large amounts of output data in the form of possible accident scenarios. These scenarios comprise complex chains of actions involving operators and equipment. Recent work described in (Coyne & Mosleh, 2008) has applied an extension of the ADS framework to identifying potential sources of crew-to-crew variability. This work highlighted the need for improved methods for illustrating and communicating the analysis results. Process mining could be applied to such simulated (as opposed to simulator) data to mine models that could highlight where significant crew-to-crew variability exists. In conjunction with the improvement described above to incorporate process variables into the mined models, an extremely rich model could be produced.

Finally, process mining from the conformance perspective could prove valuable in the area of nuclear nonproliferation. With respect to nuclear fuel reprocessing facilities in particular, comparison of an actual process with the expected one could identify process deviations, where such deviations could be indicative of an attempt to divert special nuclear material from the facility.
References


Appendix A  Details of U.S. plant analysis

This appendix contains details of the analysis for each of the four crews participating in the follow-on benchmarking exercise at a U.S. nuclear plant simulator that was summarized in Ch. 6.

A.1 Scenario description: Loss of All Feedwater (LOFW) + Steam Generator Tube Rupture (SGTR)

The plant has three main feedwater pumps (11, 12, 13) and four auxiliary feedwater (AFW) pumps (11, 12, 13, 14). AFW pump 14 receives motive power from a steam turbine, while the other three AFW pumps are motor-driven. At the start of the scenario, the Shift Technical Advisor (STA) is not in the control room. He arrives within five minutes of being summoned by the operators on duty in the control room. The operating crew on duty comprises four licensed control room operators plus the STA. The plant is operating at 100% rated power at the beginning of the scenario.

Two minutes into the scenario (+ 2 min) the plant experiences a total loss of all feedwater, presenting as a failure of main feedwater pump 11, followed within ten seconds by failure of pumps 12 and 13. If the crew does not trip the reactor in response to the loss of feedwater, steam generator (SG) level will decrease due to boiling and an automatic reactor trip will occur shortly thereafter on low SG level. This reduces the energy input rate to the SGs to reactor decay heat levels. Flow from AFW is designed to make up for SG secondary-side inventory lost through boiling.

The AFW pumps will receive a signal to start automatically on low SG level. In the simulation, AFW pump 14 will start but its speed will be too high, and it will suffer irreparable damage as a result. AFW pump 11’s shaft will seize and it will trip and be unavailable. AFW pump 13 will start, but its shaft will shear, and no flow will be indicated to the operators. AFW pump 12 will start and will indicate full flow, but will not be feeding the steam generators because of a recirculation valve being open inadvertently. There is no indication of the recirculation valve’s position in the control room.

As a result of these failures, there will be no AFW flow to the SGs, and SG level will continue to drop as a result of boiling caused by the reactor decay heat load. The criteria to transfer to the procedure intended to deal with a loss of all secondary heat removal (FR-H.1) are met. However, because there is flow indicated (although not available) from AFW pump 12, the operators may not be aware of the need to transfer to this procedure in the short term. Upon transferring to procedure FR-H.1, the operators are directed to establish heat removal from the primary system via so-called feed-and-bleed cooling. This involves actuation of safety injection (SI) pumps, which provide the “feed,” and opening of pressurizer relief valves to provide the “bleed.”

Before directing actuation of feed-and-bleed cooling, procedure FR-H.1 directs the operators to attempt to restore secondary heat removal via AFW, as this is the more desirable means of removing decay heat from the reactor. To attempt to establish AFW flow to the SGs, the operators can do the following:

- Dispatch a plant operator to close the open recirculation valve that is preventing flow from AFW pump 12 from reaching its designated SG,
• Attempt to cross-connect flow from AFW pump 12 to one of the other three SGs

By the scenario design, if the crew attempts to close the recirculation valve before the start of feed-and-bleed cooling, the plant operator will delay closing the valve until after feed-and-bleed cooling is established. If the crew tries cross-connecting before establishing bleed-and-feed cooling, the valve breaker will open and the valve will remain closed. If the crew cross-connects after establishing bleed-and-feed cooling, the valve will open as desired.

After bleed-and-feed cooling has been established, procedure FR-H.1 directs the crew to attempt to restore secondary heat removal, so that feed-and-bleed cooling can be terminated. The crew will be able to establish AFW flow to one or several SGs, thus restoring secondary heat removal. However, a tube rupture occurs in the first SG that is fed. The higher primary pressure drives reactor coolant into the secondary side of the SG via the ruptured tube, causing SG level to increase. The crew will want to fill a SG in order to be able to exit FR-H1, and the SGTR signal of increasing SG level may be masked by AFW flow to the SG, as long as it is being fed. The leak size of the ruptured tube is about 2,000 liter/min at 100% power, but the actual flow from the ruptured tube will depend on the differential pressure between the reactor coolant system and the secondary side of the ruptured SG.

Because of minimal steam flow from the SGs at the time of the SGTR, one of the primary signals of this event, high secondary radiation, will be masked from the operators. The following operator actions are expected in this scenario prior to the occurrence of SGTR.

• When main feedwater stops, start the feedwater booster pump and the startup feedwater pump (these pumps do not start by scenario design)
• Manually trip the reactor when there is no feedwater flow to the SGs (anticipate the automatic trip that will occur on low SG level)
• Immediate action after the reactor trip:
  o Try to open the cross-connect valve from AFW to SG B (cannot be opened)
• Following this immediate action, check if AFW flow needs to be reduced. Send plant operators to check the following:
  o Circuit breaker for AFW pump
  o Cross-connection valve from AFW to SG B
  o Check valve line-up, determine why there is no AFW flow
  o Reset overspeed trip on AFW pump 14 turbine
• Transfer to procedure ES-0.1 and start monitoring Critical Safety Functions (CSFs)
• Detect that there is no AFW flow to SG B (no flow to any SG)
• Decide to transfer to procedure FR-H.1 based on loss of secondary heat removal
• Start feed-and-bleed cooling according to procedure FR-H.1

Following the occurrence of the SGTR, the following operator actions are expected.

• Terminate feed-and-bleed cooling when level in one SG is below the procedural set point.
• Transfer to procedure ES-11 or ES-1, depending on pressure in the reactor coolant system
- Transfer to procedure E-3 and isolate the ruptured SG
- Transfer to procedure ECA-31 from Step 6 in procedure E-3, based on SG pressure

The following human failure events (HFEs) are defined.

1. HFE-1: Failure to start feed-and-bleed cooling before SG dry-out, which is expected within 90 min. The actual time to dry-out is a function of how long after the loss of feedwater the reactor trip occurs, as shown in Figure 62. Successful feed-and-bleed cooling requires at least one high-pressure safety injection (HPSI) pump to be running, with both pressurizer power-operated relief valves (PORV) open, along with the associated PORV isolation valves.

![Time to Feed & Bleed](image)

*Figure 62  Time available to initiate feed-and-bleed cooling as a function of time from loss of feed to reactor trip*

2. HFE-2: Failure to identify and isolate the ruptured SG before it overfills, expected to be about 33 min after the SGTR (about 75 min from the start of the scenario)

3. HFE-3: Stop safety injection to the reactor coolant system before the ruptured SG overfills, timing expected to be same as for HFE-2.

### A.2 Process mining of data for individual crews

The following sections describe the detailed mining results for each of the four crews participating in the exercise. The principle tool was the fuzzy miner in ProM, described in (Guenther C. W., 2009) and in Appendix C to this dissertation. The analysis was supplemented by plots of important process variables.
A.2.1 Analysis of Crew 1

The scenario begins with a total loss of all main feedwater flow a little over two minutes into the simulation, as shown by the plot in Figure 63. A fuzzy model for the actions of Crew 1 during the scenario is shown in Figure 66, with a magnified view of the early portion of the model shown in Figure 67. This model clusters the trip of the reactor by the operating crew (Cluster 136 in Figure 67), which occurs at about 180 sec. as shown in Figure 63, anticipating the automatic trip that will occur when the low SG level set point is reached. An expanded view of Cluster 136 is shown in Figure 68. Because of the lack of feed flow, SG level is falling rapidly in all four SGs, as indicated in Figure 65. This anticipatory trip is in accordance with the training received by the operators at this particular plant.

Figure 63  Plot of reactor power vs. time for Crew 1, with the sharp drop in power at about 180 sec. indicating the trip of the reactor by Crew 1
Figure 64  Feedwater flow for Crew 1, showing immediate loss of all main feedwater to three of the four steam generators, followed very quickly by loss of flow to the fourth generator (SG A)
Figure 65  Steam generator level vs. time for Crew 1, illustrating decreasing levels in all four generators
Figure 67  Magnified view of beginning of fuzzy model in Figure 66, showing Cluster 136, which contains the trip of the reactor

Figure 68  Expanded view of Cluster 136 in Figure 66, showing anticipatory trip of reactor

The fuzzy model also shows the operators’ attempts to start the feedwater booster pump and the startup feedwater pump, per the procedure. Because these pumps fail in the scenario, SG level continues to fall in all SGs.
With no flow from main feedwater to the SGs, the operators are first concerned with attempting to remove decay heat from the reactor core via AFW flow. The operators tripped the reactor within about 50 sec. of the loss of main feedwater, as can be deduced from the plots in Figures 63 and 64. From the plot in Figure 62, this means the operators have approximately 30 minutes to establish feed-and-bleed cooling should AFW not be available. As discussed above, the design of this scenario is such that all AFW pumps will fail to provide flow; however, AFW pump 12 will start and indicate full flow in the control room, but will not be feeding the SG because of a recirculation valve being open inadvertently. There is no indication of the valve’s position in the control room. The plot of AFW flow shown in Figure 69 illustrates the information available to the operators in the control room, with Figure 70 showing the indication in the control room itself. Thus, according to this indicator, there is AFW flow going to SG B, while level in SG B is decreasing. The operators could be confused by the misinformation about AFW flow, and this could cause them to delay the start of feed-and-bleed cooling to remove decay heat directly from the primary system.

Figure 69  Indicated AFW flow vs. time, showing indication of AFW flow to SG B
The process governing operator behavior at this point is the Critical Safety Function (CSF) status tree for establishing a decay heat sink, shown in Figure 71. The operators enter this tree on the far left, and the answer to the first question, regarding SG level as indicated on narrow-range (NR) instruments in the control room, will be “No,” as level in all four SGs will be very low, as shown in Figure 65. This leads them to the uppermost question in Figure 71, regarding AFW flow to the SGs. As shown in Figure 70, there is indication of AFW flow to SG B, making the answer to this question ambiguous; if the operators believe the indication of AFW flow to be accurate, they will answer “Yes” to this question, and this will lead them to a series of other questions related to SG pressure. Falling SG pressure, as shown in Figure 74, could ultimately lead the crew to the lowermost yellow branch in Figure 71, which will tell them to enter procedure FR-H.5, whereas the red branch leads them to a different procedure, FR-H.1. It is this latter procedure that directs the operators to initiate feed-and-bleed cooling.
Figure 71 CSF status tree for establishing decay heat sink. Operators could be on either the upper red path or the lowermost yellow path

In this simulation of the scenario, Crew 1 took the branch to enter FR-H.1, and quickly reached Step 10, which directs them to actuate feed-and-bleed cooling. This can be seen in the fuzzy model excerpt shown in Figures 72 and 73. This is corroborated by the plot of SI flow from the three SI trains shown in Figure 75.
Figure 72  Excerpt of fuzzy model for Crew 1, showing Cluster 131, which contains SI actuation

Figure 73  Expanded view of Cluster 131 showing actuation of SI
Figure 74  SG pressure vs. time, showing that pressure in all four SGs meets the criteria for the downward branches in the CSF status tree in Figure 71
According to FR-H.1, the operators should stop the running reactor coolant pumps (RCPs) in Step 2. This is an important action because the RCPs add a significant amount of energy to the primary coolant, which must be removed. Stopping the RCPs removes this source of energy, and extends the time available to restore AFW or initiate feed-and-bleed cooling. However, Crew 1 delayed stopping the RCPs until after actuating SI in Step 10. The ordering of SI actuation and stopping the RCPs can be inferred in the fuzzy model shown in Figure 72, where the actions to stop the RCPs have been clustered into Cluster 130. An expanded view of Cluster 130 is shown in Figure 76. Animating the fuzzy model brings this detail out with even greater clarity. Figure 77 shows an excerpt of the fuzzy model animation, illustrating the trip of the RCPs following the trip of the reactor.
The “bleed” portion of feed-and-bleed cooling is achieved by opening a pressurizer relief valve, as directed by Step 13 in procedure FR-H.1. This can also be seen in Figure 72, where the corresponding audit trail entry from the simulator action log is PRESS RLF PORV OPEN. This is corroborated by the pressurizer relief valve discharge temperature, which increases
dramatically when the valve is opened, as shown in Figure 78. The valve discharges to the pressurizer relief tank (PRT), which is located inside containment. The level in the PRT begins to rise once the relief valve is opened, as indicated in Figure 79.

![Pressurizer relief valve discharge temperature for Crew 1](image)

**Figure 78** Pressurizer relief valve discharge temperature for Crew 1, indicating start of feed-and-bleed cooling shortly before 1,000 sec
Figure 79  Pressurizer relief tank (PRT) level, for Crew 1, indicating start of feed-and-bleed cooling shortly before 1,000 sec

Once feed-and-bleed cooling is established, later steps in procedure FR-H.1 direct the operators to continue attempting to restore secondary heat removal via AFW, as this means of removing decay heat is preferred to feed-and-bleed cooling. The operators carry out these steps, as can be seen in the excerpt of the fuzzy model shown in Figure 80.

By the design of this scenario, SGTR will occur in the first SG to which AFW flow is restored. As indicated by the plot of SG level in Figure 81, Crew 1 restores flow first to SG B. Following the SGTR, the higher pressure on the primary side will cause reactor coolant to flow into the secondary side of the SG, causing secondary side level to rise. Radioactivity in the primary coolant could be detected by radiation monitors on the steam lines leaving the SG, and this is a primary symptom of SGTR, and a cue for the operators to enter procedure E-3, which is tailored for SGTR. However, in this case the steam lines were isolated prior to restoring AFW flow, and the SG blowdown and sampling lines, which are also instrumented with radiation monitors, were isolated by the SI signal. As a result, the radiation monitors on the steam lines, blowdown lines, and sampling lines will not be available.
Figure 80  Fuzzy model excerpt showing actions taken by Crew 1 to restore AFW following initiation of feed-and-bleed cooling

Figure 81  Level in SG B vs. time for Crew 1, with rapidly rising level shortly after 1,000 seconds, which is indicative of SGTR in SG B
At this point in the scenario the operators are in procedure FR-H.1 and perhaps FR-P.1 (as a result of feed-and-bleed cooling, procedure FR-P.1 has the operators re-establish a steam bubble in the pressurizer). In order to deal with the SGTR that has occurred, the operators must leave FR-H.1, transition to procedure ES-11, and from there transition to procedure E-3 via the conditional information page (CIP) at the back of ES-11. Step 6 of procedure FR-H.1 directs the operators to check SG levels, and if NR level is greater than 35% in one or more SGs, they should exit FR-H.1. The criterion to enter E-3 from the CIP is an uncontrolled increase in any SG level or any abnormal SG radiation level. By the design of this scenario, the operators are expected to transition to E-3 based on uncontrollable level in the ruptured SG, which in this case is SG B.

The fuzzy model illustrates that the operators next reduced AFW flow in response to rising level in SG B. This is shown in the excerpt from the fuzzy model in Figure 82. Once the operators have determined that an SGTR has occurred, they transition to procedure E-3, which directs them to isolate the ruptured SG. The simulator did not log procedural transitions, unlike the simulator at Halden. The operators isolated SG B at about 45:00. The action to isolate the SG is clustered into Cluster 110 in the fuzzy model excerpt in Figure 83. An expanded view of Cluster 110 is shown in Figure 84. Successful isolation of the ruptured SG is confirmed by 1) the plot of AFW flow to SG B in Figure 85, which shows flow dropping to zero shortly after 2,000 sec., 2) the plot of SG pressure in Figure 86, which shows the pressure in SG B decreasing without actuating a SG relief valve, and 3) the plot of SG level in Figure 87, which shows level stabilizing following isolation of SG B.

![Figure 82 Excerpt of fuzzy model for Crew 1 showing action to reduce AFW flow to SG B (event shown in red) in response to increasing SG level](image-url)
Figure 83  Excerpt of fuzzy model for Crew 1 showing isolation of SG B clustered into Cluster 110

Figure 84  Expanded view of Cluster 110 in fuzzy model for Crew 1 showing isolation of SG B
Figure 85 Plot of AFW flow to SG B for Crew 1, showing isolation of flow to SG B shortly after 2,000 sec.
Figure 86 Plot of SG pressure vs. time for Crew 1, showing decreasing pressure in SG B following isolation shortly after 2,000 sec.

Figure 87 Plot of level in ruptured SG for Crew 1, showing level stabilizing, and indicating isolation of SG B
A.2.2 Analysis of Crew 2

The same scenario was run for each of the four crews participating in this experiment. Therefore, the initial indications, such as the plot of main feedwater flow shown in Figure 64, were the same for each crew. The fuzzy model for Crew 2 is shown in Figure 88.
Figure 88 Fuzzy model for Crew 2
As for Crew 1, there was an anticipatory reactor trip based on decreasing SG level. This can be seen in the fuzzy model excerpt shown in Figure 89, which is very similar to that for Crew 1 in Figure 67.

Figure 89  Excerpt of beginning of fuzzy model for Crew 2, showing anticipatory reactor trip

As was the case for Crew 1, the operators have indication of AFW flow to SG B, as shown in Figures 69 and 70. Also, they are presented with the same dilemma with respect to which procedure to enter from the CSF status tree in Figure 71. Like Crew 1, Crew 2 also chose to enter FR-H.1 from the CSF status tree. However, when Crew 2 entered procedure FR-H.1, they, in contrast to Crew 1, executed the procedure steps in the anticipated order, thus stopping the RCPs at Step 2. However, because of the rather disconnected structure of the fuzzy model for Crew 2, it is not straightforward to infer this ordering; animation of the fuzzy model makes the ordering somewhat clearer. Crew 2 initiated feed-and-bleed cooling at about 1,200 seconds, as shown by the plot of PORV discharge temperature in Figure 90. This is slightly later than Crew 1 (compare to Figure 78).
Figure 90  Plot of PORV discharge temperature vs. time for Crew 2, showing opening of PORV for feed-and-bleed cooling at about 1,200 sec., slightly later than the same action for Crew 1

Once feed-and-bleed cooling is established, later steps in procedure FR-H.1 direct the operators to continue attempting to restore secondary heat removal via AFW, as this means of removing decay heat is preferred to feed-and-bleed cooling. Crew 2 takes about the same overall time as Crew 1 to re-establish AFW flow, as can be seen from plots of SG level in Figures 91 and 92. The actions of Crew 2 to re-establish AFW flow can be seen by animating the fuzzy model shown in Figure 88. One limitation to the animation of the fuzzy models generally, which is particularly a problem for Crew 2 because of the somewhat larger size of some of the clusters, is that the animation does not give any detail about what events in the cluster are activated. It is only possible to “see inside” a cluster in the static view of the model, not with animation, meaning that the analyst must toggle back and forth between the static and animated views to examine the events contained in a particular cluster of interest. It can also be difficult to follow the flow of the animation, as nodes in the model are not activated in a continuous linear fashion; the animation can “jump” from a node in one part of the model to a node in a completely different part of the model. With magnification high enough to discern the node names, this jumping around of the animation can be difficult to follow.
Figure 91  Plot of SG B level for Crew 1. Compare time at which level begins to increase with similar plot for Crew 2 in Figure 92

Figure 92  Plot of SG B level for Crew 2

The next portion of the scenario is spent in procedure FR-P.1, attempting to re-establish a steam bubble in the pressurizer. Steam flow from SG B was isolated relatively early, as shown in Figure 93. However, isolating AFW flow took considerably more time, as shown in Figure 94. The level in the ruptured SG stabilizes, as indicated in Figure , albeit at a higher level than that achieved by Crew 1. Crew 2 eventually isolates the ruptured SG, with no release of radioactivity, as indicated by the plot of SG pressure shown in Figure 96; however, the extra time spent in
procedure FR-H.1, attempting to establish a secondary heat sink so that feed-and-bleed cooling can be terminated, has resulted in a longer time to identify and isolate the ruptured SG relative to the time taken by Crew 1.

Figure 93  Plot of steam flow from SG B vs. time, showing isolation of steam flow from the ruptured SG at about 24 minutes into the scenario

Figure 94  AFW flow from SG B, indicating isolation of AFW flow to the ruptured SG at about 2,400 into the scenario
Figure 95  Plot of level in ruptured SG for Crew 2, showing level stabilizing, indicating isolation of SG B

Figure 96  Plot of SG B pressure for Crew 2, showing pressure stabilized following isolation of ruptured SG

A.2.3 Analysis of Crew 3

The simulator action log for Crew 3 was significantly larger than for the other crews; the imported log contained over 100,000 audit trail entries, suggestive of a more complex process for this crew than for Crews 1 and 2. As before, manual filtering using the Event Log Filter in ProM
was a necessary first step for mining this event log. In this case, the filtered log contained slightly more than 500 audit trail entries.

The fuzzy model for Crew 3 is shown in Figure 97. This model has a lower level of node connectivity than the corresponding models for Crews 1 and 2 shown above. This will make the model somewhat more difficult to employ in support of task analysis and simulator data reduction. Crew 3 responded to the loss of all main feedwater by anticipating the reactor trip, very similar to Crews 1 and 2. This is illustrated in the excerpt from the fuzzy model shown in Figure 98.
Figure 97  Fuzzy model for Crew 3
Figure 98  Excerpt of Crew 3 fuzzy model showing anticipatory reactor trip

As was the case for Crews 1 and 2, Crew 3 has indication of AFW flow to SG B, similar to what is shown in Figure 70. Despite the indication of AFW flow, SG level drops monotonically in a fashion similar to what is shown in Figure 65.

Crew 3 transitioned to procedure FR-H.1 more quickly than Crews 1 and 2, and initiated feed-and-bleed cooling upon entry. The plot of PORV discharge temperature vs. time in Figure 99 shows that the pressurizer PORV was opened at about 700 sec.
Following the entry into FR-H.1 and the establishment of feed-and-bleed cooling, Crew 3 worked to restore AFW flow to the SGs. Crew 3 restored AFW flow to SG C, in contrast to Crews 1 and 2, who restored flow to SG B. They also took significantly longer than Crews 1 and 2 in restoring AFW flow, as can be seen by comparing the time at which level begins to rise in the ruptured SG. By the design of the scenario, SGTR occurs in the first SG to which AFW flow is restored, SG C in this case. This causes level in SG C to begin rising, as shown in Figure 100.
Figure 100 SG level for Crew 3, with rise in level in SG C at about 1,600 seconds indicating SGTR

Crew 3 also managed to isolate the ruptured SG by isolating steam flow and terminating AFW flow, as indicated in Figures 101 and 102. They were considerably slower in terminating AFW flow to the ruptured SG, as can be seen by comparing Figure 102 with Figure 85 for Crew 1 and Figure 94 for Crew 2.

Figure 101 Plot of steam flow from SG C for Crew 3, showing isolation of steam flow
Figure 102  Plot of AFW flow to SG C for Crew 3, indicating isolation of AFW flow to the ruptured SG at about 3,000 sec.

A.2.4 Analysis of Crew 4

The fuzzy model for Crew 4 is shown in Figure 103. Crew 4 responded to the loss of all main feedwater by anticipating the reactor trip, very similar to Crews 1-3. This is illustrated in the excerpt from the fuzzy model shown in Figure 104.
Figure 103  Fuzzy model for Crew 4
As was the case for Crews 1-3, Crew 4 has indication of AFW flow to SG B, similar to what is shown in Figure 70. Despite the indication of AFW flow, SG level drops monotonically in a fashion similar to what is shown in Figure 65.

Like Crew 2, Crew 4 initiated feed-and-bleed cooling later than Crews 1 or 3, as indicated by the plot of PORV discharge temperature in Figure 105.
Figure 105  Plot of PORV discharge temperature for Crew 4, showing initiation of feed-and-bleed cooling at about 1,200 sec.

Following entry into procedure FR-H.1, Crew 4 restored flow to SG B shortly after 1,200 sec., as indicated in Fig. 106. This precipitates a rupture in SG B, as was the case for Crews 1 and 2.

Crew 4 did not transition to procedure E-3 as quickly as Crews 1-3. Radioactivity release occurred before the ruptured SG could be isolated, as is shown in Figure 107.
Figure 106 SG B level for Crew 4, indicating restoration of AFW flow to SG B shortly after 1,200 sec.

Figure 107 Plot of SG pressure for Crew 4. Radioactivity is released from SG B, as indicated by cycling of pressure at the relief valve set point shortly after 2,000 sec.
Appendix B: Petri net fundamentals

This appendix summarizes some basic features of Petri nets. The reader is referred to (Cassandras & Lafortune, 2008) for more details on Petri nets and pointers to the extensive and expanding technical literature on Petri nets. In a Petri net, an event is associated with a state transition. The conditions that have to be met in order for a transition to occur are associated with places in the net. See Figure 108 for an example. In this figure, places are shown as circles, and transitions are shown as rectangles, a standard Petri net symbology.

Figure 108 Example Petri net

Each transition can have a set of input and output places, and markings or tokens are used to indicate place occupancy in the net. The nodes (places and transitions) in the net are connected by arcs, and arcs cannot directly connect nodes of the same type; arcs connect place nodes to transition nodes and transition nodes to place nodes. Note that more than one arc can connect two nodes, with the number of arcs referred to as the weight. Thus, we have the following mathematical definition of a marked Petri net:

Definition: Marked Petri net

A marked Petri net is a five-tuple \((P, T, A, w, x)\) where

- \(P\) is the finite set of place nodes in the graph
- \(T\) is the finite set of transition nodes in the graph
- \(A \subseteq (P \times T) \cup (T \times P)\) is the set of arcs from place nodes to transition nodes, and from transition nodes to place nodes
- \(w : A \rightarrow \{1, 2, 3, \ldots\}\) is the weight function on the arcs
- \(x\) is a marking of the set of places, \(P\), with \(x(p_i) \in \mathbb{N}\).

The state of the Petri net is defined by its marking vector, \(x = [x(p_1), x(p_2), \ldots, x(p_n)] \in \mathbb{N}^n\), which is usually taken to be a row vector. Because there is no limit on the number of tokens in a place in the net, the set of states of the Petri net is countably infinite. A transition from one state to another is enabled when \(x(p_i) \geq w(p_i, t_j)\) for all \(p_i \in I(t_j)\), that is, when the number of tokens in place \(p_i\) is at least as large as the weight of the arc connecting place \(p_i\) to transition \(t_j\), for all \(p_i\) that are inputs to \(t_j\). Most importantly for modeling of systems, the state of the Petri net evolves dynamically in time as tokens are moved through the net. A state equation can be written down that describes the time evolution of the Petri net as the result of a particular transition “firing.” Define the firing vector, \(u\), as a row vector of 0s representing those transitions that did not fire,
with a 1 in the place of the transition that did fire. Next define the Petri net incidence matrix, $A$, with $\{j, i\}$ entry given by

$$a_{ji} = w(t_j, p_i) - w(p_j, t_i)$$  \hspace{1cm} (1)

We can now write the state equation in coordinate-free form as

$$\ddot{x} = \dot{x} + \vec{u} \ A$$  \hspace{1cm} (2)

The path of the Petri net through state space is thus a sequence of states (markings) $\{x_0, x_1, \ldots\}$ that results from a transition firing sequence $\{t_1, t_2, \ldots\}$. Given the initial state, $x_0$, the sequence of states can be generated from

$$\ddot{x}_{k+1} = \ddot{x}_k + \vec{u}_k \ A.$$  \hspace{1cm} (3)

If a state is reached from which no more transitions can fire, the Petri net is said to be *deadlocked* in that state.
Appendix C Fuzzy models

This appendix contains some details on the fuzzy modeling approach described in (Guenther C. W., 2009). Some material has been taken from that reference and is reproduced here with the kind permission of the author. (Guenther C. W., 2009) introduced fuzzy modeling, based on the metaphor of a city map, which are characterized by the fact that they are deliberately imprecise, in comparison with more traditional process models such as Petri nets. They are able to aggregate and abstract away actions that are not of interest to the analyst.

Figure 109, taken from (Guenther C. W., 2009), shows a small excerpt of a fuzzy model. The beige rectangles represent so-called primitive nodes, and would correspond to discrete operator actions in the simulator log. The green octagon represents a cluster node, comprising primitive nodes that exhibit a strong binary correlation, in a manner to be defined below. A cluster represents the observation of any number of events in the log. Arcs in the fuzzy model connect nodes, and represent precedence relations. An arc from node A to node B indicates that an observation of event A may be followed by an observation of event B.

Figure 109  Small excerpt of a fuzzy model, taken from (Guenther C. W., 2009)

Fuzzy models are intended to provide an efficient, simplified visualization of the process. Thus, they are designed as descriptive models of an underlying process, as opposed to Petri nets, which are of a more prescriptive nature. Thus, fuzzy models would not be useful for simulating process behavior, as they do not define the process logic in a sufficiently precise manner. This sacrifice of precision for simplicity, and thus comprehensibility, is deliberate.

The executional semantics for interpreting a fuzzy model are correspondingly relaxed. In the models mined herein, all events are of one class (Complete), and this simplifies the semantics somewhat over the more general case in which a log contains multiple event classes. Activation of a primitive node corresponds to the observation of that event. Activation of a cluster node corresponds to observation of one or more events contained in that cluster.

Each node (primitive or cluster) in the fuzzy model employs an AND split semantics: This means that whenever a node has been activated, it will enable the future activation of all its successors. Every node in a fuzzy model has a memory-less XOR join semantics. This means that, even if a node has multiple predecessors, it is enabled as soon as one predecessor has been activated. If a node is enabled multiple times, e.g. by multiple predecessors, the effect on that node is no different from a single enabling. Thus, nodes in a fuzzy model are memory-less, i.e. they do not “remember” how often they have been enabled.
The process may start at any arbitrary node in a fuzzy model, while different cases of the process may start in different nodes. Thus, it is allowed for starting nodes to have incoming arcs, and there is no exclusive starting point in the process. Similar to the previous point, the process may terminate in any subset of nodes in a fuzzy model. Thus, terminal nodes in the process may have outgoing arcs, and a synchronization of concurrent paths through the model in one single terminal node is not required. The complete process is implicitly considered terminated when no further activation of nodes occurs.

Figure 110, taken from (Guenther C. W., 2009), illustrates the executional semantics of a fuzzy model graphically.

Due to the universally-defined XOR join / AND split-semantics of nodes, fuzzy models describe a number of control flow patterns in the same way. The process in Figure 110 can describe a parallel split in node A, followed by executing nodes B, C, and D in parallel, and with a final synchronization in node E. Since nodes in a fuzzy model have an AND split-semantics, this behavior is trivially supported for node A. Every subsequent activation of nodes B, C, and D can enable node E independently. However, the memory-less XOR join-semantics of nodes allows them to emulate the behavior of a synchronization, simply by waiting for all possible enablements before activation. On the right side of Figure 110, traces 1 and 2 are examples for executing the model with this AND split/join semantics.

The model in Figure 110 can also be interpreted while assuming XOR split/join semantics. Node A then represents an exclusive choice, followed by either one of B, C, or D, before joined in a simple merge in node E (patterns 4 and 5 in. Any activation of node A will of course still enable all successors. However, since enablement does not force activation in fuzzy models, all successors except for one can simply abstain from activation. The XOR join-semantics of nodes can also trivially emulate the simple merge behavior. This XOR split/join behavior is represented by traces 3 and 4 in Figure 110.

Further, the model in Figure 110 can be interpreted while assuming an OR split/join semantics. Node A represents a multi-choice split, followed by any subset out of B, C, and D. Finally, node E implements a structured synchronizing merge. As with the XOR semantics, a subset of B, C,
and D simply abstains from activation despite being enabled. Combined with a “lazy” activation policy for the synchronizing node E (as with the AND semantics) fuzzy models are thus able to emulate split/join behavior with OR semantics. This behavior is represented in traces 5 and 6 in Figure 110.

Fuzzy models are also capable of expressing iteration, i.e. loops, in a process. Figure 111 shows a structured loop, which allows for the execution of the subsequence G, H, J an arbitrary number of times, after activating F and before activation of K. This iterative behavior, which is illustrated in traces 1 and 2 in Figure 111, is achieved by not activating K despite it being enabled, if G is activated again.

![Fuzzy model iteration semantics](image)

**Figure 111 Fuzzy model iteration semantics, taken from (Guenther C. W., 2009)**

Note that fuzzy models allow for K to be activated anyway, even after another loop cycle has commenced, as illustrated in trace 3 in Figure 111. This shows that, while fuzzy models are able to emulate well-known control flow patterns, the behavior for which they allow is almost always extended. This is not a flaw in the semantics, but a deliberate choice. By expressing behavior in a fuzzy, non-concrete manner, fuzzy models are able to simplify complex patterns of behavior, which makes them preferable for the task of process analysis and exploration.

While the semantics of fuzzy models allows them to cover a wide range of workflow patterns, and thus to faithfully model a large class of processes, there are also some patterns that cannot be expressed with fuzzy models. One example is multiple instances, where part of a process is executed multiple times simultaneously, by independent threads. This limitation is not expected to be a problem for this application as such instances should not be encountered in simulator logs.

The semantics of fuzzy models places few restrictions upon the behavior within clusters, i.e., what happens during the activation of a cluster is generally not defined, and assumed not to be of interest. Every cluster node in a fuzzy model represents the aggregation of a set of less-interesting events. Thus, the only semantic restriction on the activation of fuzzy models is that they represent the uninterrupted observation of any number of events from those aggregated events.

Figure 112, taken from (Guenther C. W., 2009), illustrates the semantics of fuzzy models with respect to in-cluster behavior. The given model starts with node A, which has another node B and a cluster D as successors. Finally, both B and D have a common successor C. Cluster node D represents the aggregation of events X, Y, and Z, i.e. “hidden” nodes that have been combined into this cluster.
Figure 112 Fuzzy model cluster behavior, taken from (Guenther C. W., 2009)

Figure 112 illustrates the precedence relations between X, Y, and Z in the cluster; however, these precedence relations are not covered by the executional semantics of fuzzy models, i.e. all possible sequences composed of X, Y, and Z are legal in this model, since they are aggregated into a cluster node.

Fuzzy models also allow implicit enablement of nodes. For describing the behavior of interest in a compact and simplified manner, fuzzy models may not explicitly explain all precedence relations that are possible in a process. An observed succession of two events A and B, which are not connected by a respective precedence relation in the fuzzy model, translates to an implicit enablement of the node corresponding to event B.

There are several visualization tools used in the implementation of fuzzy models in ProM. First, the width of arcs represents edge significance. Edges that symbolize more significant precedence relations have a larger width than less significant ones. Thus, important paths in the fuzzy model appear as large, wide “highways”, which immediately draw the attention of the viewer to this part of the model.

Second, correlation between nodes in the process model is also very important knowledge. The viewer needs to be aware of highly correlated parts of the model, since these are meant to represent homogeneous, semantically related groups of behavior. The implementation in ProM uses the contrast of edge color to represent the correlation expressed by each edge. Edges symbolizing highly correlated relations will be visualized in a very dark color (in front of a white background), while edges with a lower correlation will use a brighter color, reducing contrast. Thus, highly correlated parts of the fuzzy model will appear connected with darker edges, which sets them apart and makes them visually identifiable as coherent subgraphs. When hovering over an edge, the tool will dynamically display more detailed information about the represented relation, that is, the calculated values for significance and correlation. Additionally, clicking on a cluster node will open another view that reveals the sub-process represented by that cluster as another, lower-level fuzzy model. This feature allows the analyst to explore fuzzy models in a top-down manner, inspecting lower-level behavior where applicable or interesting.
C.1 Metrics used by the fuzzy miner

The first step in creating a fuzzy model, mining the initial (non-simplified) process model is straightforward. All events found in the log are translated to activity nodes, whose importance is expressed by unary significance. For every observed precedence relation between events, a corresponding directed edge is added to the process model. This edge is described by the binary significance and correlation of the ordering relation it represents. Three transformations are then applied to the model to successively simplify it.

1. In the binary conflict resolution phase, conflicting relations between two nodes are clarified. If two nodes are connected by precedence relations in both directions, these two relations are said to be in conflict. The approach taken by the fuzzy miner in ProM can distinguish between concurrency, short loops, and exceptional behavior as potential causes for these conflicts, and resolve them by removing undesired relations.

2. In the edge filtering phase, the set of precedence relations (i.e., edges) in the model, is further limited, down to those relations that are most significant, or important for routing. All relations are evaluated locally, that is, for every node, based on their significance and correlation. The least desirable of these relations are subsequently removed from the model.

3. In the final aggregation and abstraction phase, the fuzzy miner simplifies the model by removing primitive nodes. Highly correlated groups of less significant nodes are hidden from view, by combining them into clusters (i.e., they are aggregated). If a node is neither sufficiently significant, nor highly correlated with other nodes, it is removed from the model altogether (i.e., it is abstracted away).

Removing edges from the model first is important – due to the less-structured nature of real-life processes and the fuzzy miner’s measurement of long-term relationships, the initial model contains deceptive ordering relations that do not correspond to valid behavior and need to be discarded. The following sections provide details about the three phases of the simplification approach, given in the order in which they are applied to the initial model.

C.1.1 Binary conflict resolution

Whenever two nodes in the initial process model are connected by edges in both directions, they are defined to be in conflict. Depending on their specific properties, conflicts may represent one of three possible situations in the process:

- Length-2-loop: Two activities A and B constitute a loop in the process model, that is, after executing A and B in sequence, one may return to A and start over. In this case, the conflicting ordering relations between these activities are explicitly allowed in the original process, and thus need to be preserved.
- Exception: The process orders A → B in sequence; however, during real-life execution the exceptional case of B → A also occurs. Most of the time, the “normal” behavior is clearly more significant. In such cases, the “weaker” relation needs to be discarded to focus on the main behavior.
- Concurrency: A and B can be executed in any order (i.e., they are on two distinct, parallel paths), the log will most likely record both possible cases, i.e. A → B and B →
A, which will create a conflict. In this case, both conflicting ordering relations need to be removed from the process model.

Conflict resolution attempts to classify each conflict as one of these three cases, and then resolves it accordingly. For that, it first determines the relative significance of both conflicting relations. Figure 113 shows an example of two activities A and B in conflict.

**Figure 113  Conflicting relations in a fuzzy model**

**Definition 1: Relative significance.** Let \( N \) be the set of nodes in a process model, and set \( \text{sig}: N \times N \rightarrow \mathbb{R}^+ \) be a relation that assigns to each pair of nodes, \( A, B \in N \), the significance of a precedence relation over them. For each pair of nodes, \( A, B \in N \), the following relation assigns the relative importance of their ordering relation:

\[
\text{rel}(A, B) = \frac{1}{2} \sum_{X \in N} \text{sig}(A, X) - \frac{1}{2} \sum_{X \in N} \text{sig}(X, B) \tag{4}
\]

Every ordering relation \( A \rightarrow B \) has a set of competing relations, denoted \( \text{Comp}_{AB} = A_{\text{out}} \cup B_{\text{in}} \). This set of competing relations is composed of \( A_{\text{out}} \), that is, all edges starting from \( A \), and of \( B_{\text{in}} \), that is, all edges pointing to \( B \) (see Figure 113). By dividing the significance of an ordering relation \( A \rightarrow B \) by the sum of all its competing relations’ significances, we get the importance of this relation in its local context.

If the relative significance of both conflicting relations, \( \text{rel}(A, B) \) and \( \text{rel}(B, A) \) exceeds a specified threshold value, this signifies that A and B are apparently forming a length-2-loop, which is their most significant behavior in the process. Thus, in this case, both \( A \rightarrow B \) and \( B \rightarrow A \) will be preserved.

In case at least one conflicting relation’s relative significance is below this threshold, the offset between both relations’ relative significances is determined, that is, \( \text{ofs}(A, B) = |\text{rel}(A, B) - \text{rel}(B, A)| \). The larger this offset value, the more the relative significances of both conflicting relations differ, that is, one of them is clearly more important. Thus, if the offset value exceeds a specified ratio threshold, the fuzzy miner assumes that the relatively less significant relation is in fact an exception, and removes it from the process model.

Otherwise, if at least one of the relations has a relative significance below the preservation threshold and their offset is smaller than the ratio threshold, this signifies that both \( A \rightarrow B \) and \( B \rightarrow A \) are relations that are of little importance for both their source and target activities. This low, yet balanced relative significance of conflicting relations hints at A and B being executed concurrently, that is, in two separate threads of the process. Consequently, both edges are removed from the process model, as they do not correspond to factual ordering relations.
C.1.2 Edge filtering

Although conflict resolution removes a number of edges from the process model, the model still contains a large number of precedence relations (i.e., arcs connecting nodes). To further structure the model, it is necessary to remove many of these remaining edges by edge filtering, which isolates the most important behavior. The obvious solution is to remove the globally least significant edges, leaving only highly significant behavior. However, this approach yields sub-optimal results, as it is prone to create small, disparate clusters of highly frequent behavior. Also, in the subsequent aggregation step, highly correlated relations play an important part in connecting clusters, even if they are not very significant.

Therefore, the edge filtering approach used by the fuzzy miner evaluates each edge \( A \rightarrow B \) by its utility, \( util(A, B) \), which is a weighed sum of the edge’s significance and correlation. A configurable utility ratio, \( ur \in [0, 1] \) determines the weight. The utility for an edge \( A \rightarrow B \) can be defined as follows.

**Definition 2: Utility.** Let \( \mathcal{N} \) be the set of nodes in a process model, and let \( \text{sig}: \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R}_{\geq 0} \) be a relation that assigns to each pair of nodes \( A, B \in \mathcal{N} \) the significance of a precedence relation over them. Further, let \( \text{cor}: \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R}_{\geq 0} \) be a relation that assigns to each pair of nodes \( A, B \in \mathcal{N} \) the correlation of a precedence relation over them. Let \( ur \in [0, 1] \) be the utility ratio. Then, \( util: \mathcal{N} \times \mathcal{N} \rightarrow \mathbb{R}_{\geq 0} \) is a relation that assigns to each pair of nodes \( A, B \in \mathcal{N} \) the utility of their ordering relation: \( util(A, B) = ur \times \text{sig}(A, B) + (1 – ur) \times \text{cor}(A, B) \).

This approach preserves those edges that yield the highest utility value. A larger value for \( ur \) will preserve more significant edges, while a smaller value will favor highly correlated edges.

Figure 114 shows an example for processing the incoming arcs of a node A. Using a utility ratio of 0.5, that is, taking significance and correlation equally into account, the utility value is calculated, which ranges from 0.4 to 1.0 in this example.

![Figure 114 Filtering the set of incoming edges for node A](image)

Filtering edges is performed on a local basis, that is, for each node in the process model, the algorithm preserves the incoming and outgoing edges with the highest utility value. The decision of which edges get preserved is configured by the edge cutoff parameter. For every node \( N \), the utility values for each incoming edge \( X \rightarrow N \) are normalized to \([0, 1]\), so that the weakest edge is assigned a value of 0, and the strongest a value of 1. All edges whose normalized utility value exceeds the cutoff parameter are added to the preserved set. In the example in Figure 114, only two of the original edges, are preserved, using an edge cutoff value of 0.4: \( P \rightarrow A \) (with
normalized utility of 1.0), and \( R \rightarrow A \) (norm. utility of 0.56). The outgoing edges are processed in the same manner for each node.

The edge cutoff parameter determines the aggressiveness of the algorithm, that is, the higher its value, the more likely the algorithm is to remove edges. In very unstructured processes, where precedence relations are likely to have a balanced significance, it is often useful to use a lower utility ratio, so that correlation will be taken more into account and resolve such ambiguous situations. On top of that, a high edge cutoff will act as an amplifier, helping to distinguish the most important edges.

Figure 115 shows the effect of edge filtering applied to a small, but very unstructured process. The number of nodes remains the same, while removing an appropriate subset of edges clearly brings structure to the previously chaotic process model.

![Figure 115  Example of a process model before (left) and after (right) edge filtering](image)

### C.1.3 Node aggregation and abstraction

While removing edges brings structure to the process model, the most effective tool for simplification is removing nodes, enabling the analyst to focus on an interesting subset of activities. The approach taken in the fuzzy miner preserves highly correlated groups of less significant nodes as aggregated clusters, while removing isolated, less-significant nodes. Removal of nodes is based on the *node cutoff parameter*. Every node whose *unary significance* is below this threshold becomes a victim, that is, it will either be aggregated into a cluster or abstracted away (i.e., removed from the model). The first phase of the algorithm builds initial clusters of less-significant behavior as follows.

- For each victim, find the most highly correlated neighbor (i.e., connected node).
- If this neighbor is a cluster node, add the victim to this cluster.
- Otherwise, create a new cluster node, and add the victim as its first element.

Whenever a node is added to a cluster, the cluster will “inherit” the ordering relations of that node, that is, its incoming and outgoing arcs, while the actual node will be hidden. The second phase is merging the clusters, which is necessary as most clusters will, at this stage, only consist of one single victim. The following routine is performed to aggregate larger clusters and decrease their number.

- For each cluster, check whether all predecessors or all successors are also clusters.
- If all predecessor nodes are clusters as well, merge with the most highly correlated one and move on to the next cluster.
• If all successors are clusters as well, merge with the most highly correlated one.
• Otherwise, if both the cluster’s pre- and post-set contain regular nodes, the cluster is left untouched.

Note that the merging of clusters is not strictly deterministic. If a number of predecessor or successor cluster nodes of a cluster are equally correlated, the algorithm will arbitrarily pick the one with which to merge.

It is important that clusters will only be merged if the “victim” has only clusters in its pre- or post-set. Figure 116 shows an example of a process model after the first phase of clustering. Cluster A cannot merge with cluster B, as they are also both connected to node X. Otherwise, node X would be connected to a merged cluster in both directions, making the model less informative. However, clusters B and C can merge, as B’s post-set consists only of C. This simplification of the model does not remove any information, and is thus valid.

Figure 116  Excerpt of a fuzzy model after the first phase of clustering

The last phase, comprising abstraction, removes isolated and singular clusters. Isolated clusters are detached parts of the process that are less significant and highly correlated, and which have thus been folded into one single, isolated cluster node. It is obvious that such detached nodes do not contribute to the process model, which is why they are simply removed. Singular clusters consist only of one, single activity node. Thus, they represent less-significant behavior that is not highly correlated to adjacent behavior. Singular clusters are undesired, because they do not simplify the model. Therefore, they are removed from the model, while their most significant precedence relations are transitivity preserved (i.e., their predecessors are artificially connected to their successors, if such edges do not already exist in the model).

It is important to note that the binary conflict resolution and edge filtering phases occur before the aggregation and abstraction phase.

C.2 Fuzzy model animation

Fuzzy models can convey important information about a process, but they suffer from one important limitation from the analyst’s perspective: they are static. This means, the actual behavior of the process, that is, the actual succession of events as recorded in the log, is nearly impossible to discern from the resulting process model.

Animation of the fuzzy model can bring back the time dimension, adding the possibility to project actual behavior onto the static fuzzy model. Animation of a process model with real-life behavior found in event logs can express a variety of information in an intuitive, accessible manner. Analysts can easily see how various parts of the model perform in real life, simply by observing the animation.
The basic component of an animation is the actual fuzzy model. All elements, that is, nodes and edges, of the model are constantly visible, which helps the user in putting the animated information into context. However, in order to focus the analyst’s attention on the currently active parts of the process model, the **glow metaphor** is used for the visualization of process animation.

The model is visualized in a default state that is dark, similar to a night-view on the process. The background of the process model is painted black, while the nodes and edges assume a relatively uniform, dark grey color in their default, that is, passive state. While fuzzy models normally use color for encoding information such as the type of node, the animation view paints all inactive model elements in a similar, dark grey color. This default visualization in a uniform, dark grey color preserves currently uninteresting parts of the model, while preventing the analyst from being distracted by them.

Any type of activity in the animated process model is represented by the respective model elements taking on a more red color. The metaphor of glowing metal is used to represent node activation, which “heats up” the respective model elements, so that they glow in the dark. Some events during an animation can be very short-lived. An edge may be traversed in a relatively short time, and the activation of nodes happens in an atomic fashion, by definition. It is important to make sure that the analyst does not miss even short-lived events in an animation, which is especially important when many events occur at once. Thus, the glow metaphor is extended by the concept that model elements, once “heated up” by activation, have a certain afterglow. When a model element is activated, it is visualized in a bright red color. Over time, this color gradually fades back into the dark grey symbolizing the inactive state.

Perhaps the most interesting, and thus important, component of process behavior is where control is passed from one event to another. This can be interpreted as one event causing another, and is expressed by arcs in a fuzzy model. This is visualized using **token animation**. Every passing of control from a node A to another node B in the animation is visualized by a token, that is, a solid round shape that travels along the arc from A to B over time. This animation begins with the activation of A, and the token arrives at B when that node is activated. Thus, the user can predict which parts of the model will be activated in the near future, by anticipating the time of “impact” of a token at the target node.

As defined by the executional semantics of fuzzy models described above, the activation of a fuzzy node enables all of its successors (AND-split semantics). Thus, to focus attention on interesting behavior in the model, only actual activations by tokens are visualized – mere enablement is not shown. In this approach, heuristics are used to determine causal successors for each event in the log. These heuristics, described in detail in (Guenther C. W., 2009), can identify the most plausible causal relations, which are then used for token animation.

Tokens are painted in a bright white color, which represents the ultimate, “white heat” in the glow metaphor. This assists the analyst in focusing on these parts of the animation, where control flows between nodes. Every token also has a thick red line trailing it, which is solid at the token itself and gradually fades out towards the token’s origin node. The metaphor used here is the token as a comet, which leaves a trail glowing hot particles. This trail does not only underline the
glow metaphor. It allows the analyst to instantly tell the direction of a token, and it supports the user in identifying tokens in large fuzzy models.
Appendix D  Overview of Nuclear Reactor Systems

Commercial nuclear reactors fall into two broad categories, depending on whether the water used to cool the fuel is allowed to boil inside the reactor vessel. In the pressurized water reactor (the type described here), the reactor coolant (water) is maintained at a high pressure, and thus does not boil inside the reactor during ordinary operation. On the contrary, in the boiling water reactor, the reactor coolant is allowed to boil inside the reactor. In the pressurized water reactor, the water in the reactor absorbs energy from the fissioning uranium fuel, and is pumped to steam generators, where it transfers the energy from fission to water, via heat transfer across a barrier that separates reactor or primary coolant from so-called secondary coolant. The secondary coolant water boils in the steam generators, producing steam to drive a turbine, wherein the mechanical and thermal energy of the steam is converted to electrical energy.

A simple schematic of the primary and secondary systems of a typical pressurized water reactor is shown in Figure 117. The primary includes the reactor vessel, the pressurizer, and four closed reactor coolant loops connected in parallel (of which only one loop is shown). The secondary includes the steam system, the high and low pressure turbines, and the condensate and feedwater system. The secondary systems together are sometimes referred to as the power conversion system. The sole function of the power conversion system is to generate electricity.

D.1  Primary system

Each of the four reactor coolant loops contains a reactor coolant pump, a steam generator, piping, and associated instrumentation. Attached to one of the four loops is an electrically heated pressurizer. The pressurizer maintains the pressure of the reactor coolant at a high value, which prevents the high temperature (>500°F) coolant from boiling. Reactor coolant (pure water with boric acid in solution) is pumped through the reactor core to remove the heat generated by nuclear fission. The heated water exits the reactor vessel, passes through loop piping and enters the steam generator.

Inside the steam generator, reactor coolant flows through U-tubes and transfers heat to the feedwater inside the steam generator (secondary system). The U-tubes act as a barrier between the primary and secondary cycles. Reactor coolant, now cooler, exits the steam generator and is directed to the suction of the reactor coolant pump. The reactor coolant pump returns the reactor coolant to the reactor vessel, completing the primary cycle.

D.2  Secondary system

The power conversion system begins in the shell sides of the four steam generators. At these locations the feedwater contacts the U-tubes and picks up heat from the hot reactor coolant. Since the pressure in the secondary side is less than that of the primary, the heated feedwater boils and becomes saturated steam. Saturated steam is steam that is at the same temperature as boiling water for a given pressure.

---

13 This appendix is adapted from training material prepared by the U.S. Nuclear Regulatory Commission Technical Training Center. The training material is not publicly available, but can be supplied by the author upon request.
Figure 117  Simplified schematic of primary and secondary systems for typical pressurized water reactor
The saturated steam produced in the shell sides of the steam generators exits via the main steam lines. The steam flows through the main steam line isolation valves (MSIVs) to the high pressure turbine. After flowing through the high pressure turbine, the low-energy, moisture-laden steam is routed to the moisture separator re heaters (MSRs). Each MSR, as its name implies, removes moisture from this low pressure steam and re-heats it. The moisture-free steam is superheated by extraction steam from the high pressure turbine and by steam from the main steam lines. Superheated steam is steam that is at a temperature which is greater than the saturation temperature for a given pressure. The dry, superheated steam is directed to the low pressure turbines. This steam passes through the low pressure turbine blades and exits to the main condenser. The high and low pressure turbines are mounted on a common shaft that drives the main generator.

Inside the condenser, the exhausted steam is condensed (cooled and depressurized) by passing over tubes containing water from the condenser circulating water system. The condensed steam (now called condensate) is collected in the condenser’s hotwell. The condensate is pumped from the condenser hotwell by condensate pumps. The condensate pumps discharge the condensate through condensate demineralizers, which remove impurities. The condensate then passes through several stages of low pressure feedwater heaters, in which the temperature of the condensate is increased by heat transfer from steam extracted from the low pressure turbines. The condensate exits the low pressure feedwater heaters and enters the suction of the high pressure main feedwater pumps.

The main feedwater pumps (normally driven by steam turbines) increase the pressure of the condensate (now called feedwater) so that it can enter the steam generators. From the discharge of the main feedwater pumps, the feedwater is heated in the high pressure feedwater heaters by extraction steam from the high pressure turbine. After this final heating, the feedwater passes through the feedwater regulating valves (FRVs), enters the containment, and finally enters the steam generators, thereby completing the secondary cycle.

D.3 Support and emergency systems

Attached to each reactor coolant loop cold leg is an accumulator, pressurized with nitrogen. The purpose of the accumulator is to inject borated water into the RCS if the reactor coolant system pressure boundary ruptures; i.e., a loss of coolant accident (LOCA) occurs. When the pressure in the RCS drops below the pressure in the accumulators, the nitrogen forces the borated water out of the accumulators into the RCS, providing both water to cover and cool the reactor core and boron (a neutron absorber) to keep the reactor shut down.

The residual heat removal (RHR) system is designed to provide both safety and non-safety functions. Its safety function is to provide borated water at a low pressure and a high flow rate to the RCS following a loss of coolant accident. The RHR system pumps water from the refueling water storage tank (RWST) to the RCS for the short term and recycles water from the containment building sump back into the reactor coolant system for long term cooling. Its nonsafety function is to remove decay heat from the core after a shutdown. Decay heat removal is accomplished by pumping hot water from an RCS hot leg through heat exchangers and then back into the RCS via the cold legs.
The safety injection (SI) system is another emergency core cooling system. Its function is to inject borated water from the RWST into the RCS after a LOCA. Although the SI system discharge capacity is much less than that of the RHR system, its discharge pressure is greater.

The auxiliary feedwater (AFW) system supplies, in the event of a loss of the main feedwater, sufficient feedwater to the steam generators to remove primary system stored heat and residual core energy (decay heat). AFW must also be available under accident conditions, such as a small break loss of coolant accident, so the plant can be brought to a safe shutdown condition.

The AFW system is designed to automatically start and supply sufficient feedwater to prevent the relief of primary coolant through the pressurizer safety valves. The AFW system has an adequate suction source and flow capacity to maintain the reactor at hot standby for a period of time and then cool the Reactor Coolant System (RCS) to a temperature at which the RHR system may be placed in operation.
Appendix E  Overview of Emergency Procedures

This appendix describes the emergency response guidelines for a typical pressurized water reactor (PWR). This is a commonly used framework for developing plant-specific procedures; however, the actual guidelines and plant-specific procedures are proprietary and cannot be reproduced. These procedures provide the control room operations crew (herein referred to as the operator) with symptom-based technical guidance for response to emergency transients.

E.1 Operator role in control room emergency operation

Safe and reliable operation of the nuclear power plant is accomplished through a combination of machine functions (i.e., automatic control, protection and safeguards systems) and operator functions that were defined and allocated during the design process. The role of the operator in this human-machine system changes depending on the state of the plant. The role and response pattern of the operator can be divided into three plant operational states: normal operation, abnormal operation and emergency operation. These states are shown schematically in Figure 118.

![Operator response model for procedural development](image)

Figure 118  Operator response model for procedural development

---

14 This appendix is adapted from training material prepared by the U.S. Nuclear Regulatory Commission Technical Training Center. The training material is not publicly available, but can be supplied by the author upon request.
In the normal operational state, the operator's role is to change the plant state in a controlled manner to generate electrical power efficiently. Changes to the plant state are operator-paced and normal operational limits are well defined. The plant control systems maintain plant parameters within normal operational limits and plant protection and safeguards systems are available should plant parameters exceed normal operational limits.

In the abnormal operational state, the operator’s role is to respond to alarm conditions caused by plant parameters exceeding normal operational limits. Depending on the source of the alarm, a spectrum of potential plant responses is possible. The operator must assess the plant condition and take recovery actions to prevent plant parameters from exceeding the reactor protection operational limits and to restore plant parameters to their normal operational limits. Plant protection and safeguards systems are available should plant parameters exceed reactor protection operational limits.

In the emergency operational state, plant parameters have exceeded the limits for reactor protection. The plant state has shifted from a state that is operator-paced to one that is largely machine-paced. Consequently, the operator's role has shifted from controlling the plant state to responding to the changing plant state. Two levels of operator response (i.e., reactor trip response and safety injection response) are included in the emergency operational state. In responding to a reactor trip, the operator augments plant protection systems in their role of stabilizing plant conditions and, depending on the cause of the reactor trip, initiates recovery operation. In responding to a safety injection condition, the operator augments plant safeguards systems in protecting plant safety, assesses the plant condition, and initiates recovery actions while continuously ensuring that plant safety is maintained. Operator actions in each level are guided by operating procedures tailored to each operational state. The procedures form a network that guides operator actions in each plant operational state.

E.2 Framework for emergency operations

Emergency transients (i.e., transients and accidents that cause plant parameters to exceed reactor protection limits) challenge the accident management capabilities of the operator. The emergency response guidelines provide the operator with a well-defined framework for emergency operations. The operator's role and special needs are addressed through providing a network of predefined symptom-based strategies for systematically responding to any developing emergency transient.

The network of emergency procedures is listed in Figure 119 and Figure 120. The relationship among the critical safety functions is illustrated in Figure 121. The concepts involved in emergency operations are shown in Figure 122. Finally, the structure of the emergency procedures that implement these concepts is shown in Figure 123.
Figure 119 Optimal recovery guidelines
**Figure 120** Critical safety function restoration guidelines

- **F-0** Critical Safety Function Status Trees
  - **F-0.1** Subcriticality
  - **F-0.2** Core Cooling
  - **F-0.3** Heat Sink
  - **F-0.4** Integrity
  - **F-0.5** Containment
  - **F-0.6** Inventory

- **FR-S.1** Response to Nuclear Power Generation/ATWS
- **FR-S.2** Response to Loss of Core Shutdown

- **FR-C.1** Response to Inadequate Core Cooling
- **FR-C.2** Response to Degraded Core Cooling
- **FR-C.3** Response to Saturated Core Cooling

- **FR-H.1** Response to Loss of Secondary Heat Sink
- **FR-H.2** Response to Steam Generator Overpressure
- **FR-H.3** Response to Steam Generator High Level
- **FR-H.4** Response to Loss of Normal Steam Release Capabilities
- **FR-H.5** Response to Steam Generator Low Level

- **FR-P.1** Response to Imminent Pressurized Thermal Shock Condition
- **FR-P.2** Response to Anticipated Pressurized Thermal Shock Condition

- **FR-Z.1** Response to High Containment Pressure
- **FR-Z.2** Response to Containment Flooding
- **FR-Z.3** Response to High Containment Radiation Level

- **FR-L.1** Response to High Pressurizer Level
- **FR-L.2** Response to Low Pressurizer Level
- **FR-L.3** Response to Voids in Reactor Vessel
Figure 121 Relationships among critical safety functions

Figure 122 Emergency operations concepts
Figure 123 Structure of the emergency procedures for optimal safety function recovery
Curriculum Vitae

Dana Kelly was born on January 24, 1959 in Spartanburg, SC in the United States of America.

After finishing an A.B. in Physics in 1983 at Cornell University in Ithaca, NY in the USA, he studied Nuclear Science and Engineering at Idaho State University in Pocatello, ID in the USA, and Operations Research at The George Washington University in Washington, DC. From 2008-2011 he carried out the research for his PhD project at the Eindhoven University of Technology, The Netherlands, the results of which are presented in this dissertation. Since 1988 he has been employed at Idaho National Laboratory, where he is a Distinguished Staff Scientist.