Nordisk
kerne-
sikkerheds-
forskning

Nordisk
kärn-
säkerhets-
forskning

Pohjoismainen
ydin-
turvallisuus-
tutkimus

Nordic
nuclear
safety
research

RAK-1

NKS/RAK-1(95)R1

SUMMARY OF APPROACHES TO DYNAMIC ANALYSIS OF MAN-MACHINE INTERRACTION

E. Hollnagel

Human Reliability Analysis, UK

May, 1995

TABLE OF CONTENTS

Summary of Approaches To Dynamic Analysis Of Man-Machine Interaction

NKS/RAK-1(95)R1
Version 1.1 - January 1995

1. INTRODUCTION

The present report was commissioned as a study under the first phase of the NKS/RAK-1, Sub-project 3. The topic of this sub-project is integrated sequence analysis with emphasis on human-system interaction. The report provides the following:

- a presentation of the principles of dynamic event analysis (joint system simulation),
- a short survey and characterisation of the main existing systems, and
- a recommendation of concepts and techniques in relation to the aims of the NKS/RAK-1 project.

It is intended that the recommendations shall be used as part of the further planning of the sub-project.

The work reported here is an extension of the presentation that was given at the first NKS meeting on August 26, 1994. The work has been carried out by Human Reliability Associates, Ltd., in the period October-November, 1994.

2. PRINCIPLES OF DYNAMIC EVENT ANALYSIS

Complex industrial systems, such as nuclear power plants, must be designed, implemented, operated, and maintained with great care. As part of this there is a need to look at several things:

- Whether the system will **perform as specified** and **comply with the functional requirements**. This is usually done through **testing** (verifying) both that design specifications are achievable and that they are achieved. It is assumed that design specifications follow established guide-lines, for example regarding human factors, task allocation, etc. The analysis/ evaluation can specifically try to determine whether the operators are able to perform the required tasks given the working conditions.

- That the system will **perform reliably** and not fail during the situations considered by the design. This means that there are **no opportunities for failure**. This analysis identifies the events and actions that can be the cause of an incident (initiating events) - both normal and beyond design base accidents. The analysis must look at single and multiple events, as well as the external conditions (common modes) that may contribute to a failure and the possible dependencies between events.

- That the system is able to **respond appropriately** to possible incidents so that (1) uncontrolled release of material and energy is avoided or contained, and (2) a safe state (for example normal operation) is reached as quickly as possible. This is achieved by **predicting** possible consequences of initiating failure events.

To provide these assurances it is necessary to evaluate the **effects of system design**, to analyse **possible initiating events,** and to analyse the system's **failure response potential**. These evaluations and analyses are needed both for fully automated systems and for systems including Man-Machine Interaction (MMI). The analyses identify the various ways a system can respond to an unexpected or untoward event. The cause may be one of the initiating system events, or an external / extraneous cause. The analyses must consider issues of detection, readiness of remedial action, and alternative solutions (success paths). It must also consider the reliability of the failure response potential.

2.1 The Need For Dynamic Analysis

Risk and reliability analyses, such as PSA/PRA and HRA, have traditionally been based on a discrete representation of accident sequences using event trees, cf. Figure 1. Each part - or block - of the event tree may be analysed further using techniques such as fault trees. In particular, the contribution of human erroneous actions can be analysed using well-known HRA methods (Dougherty & Fragola, 1988; Gertman & Blackman, 1994). The human contribution can be related to either the initiating event or to the way in which the event propagates.

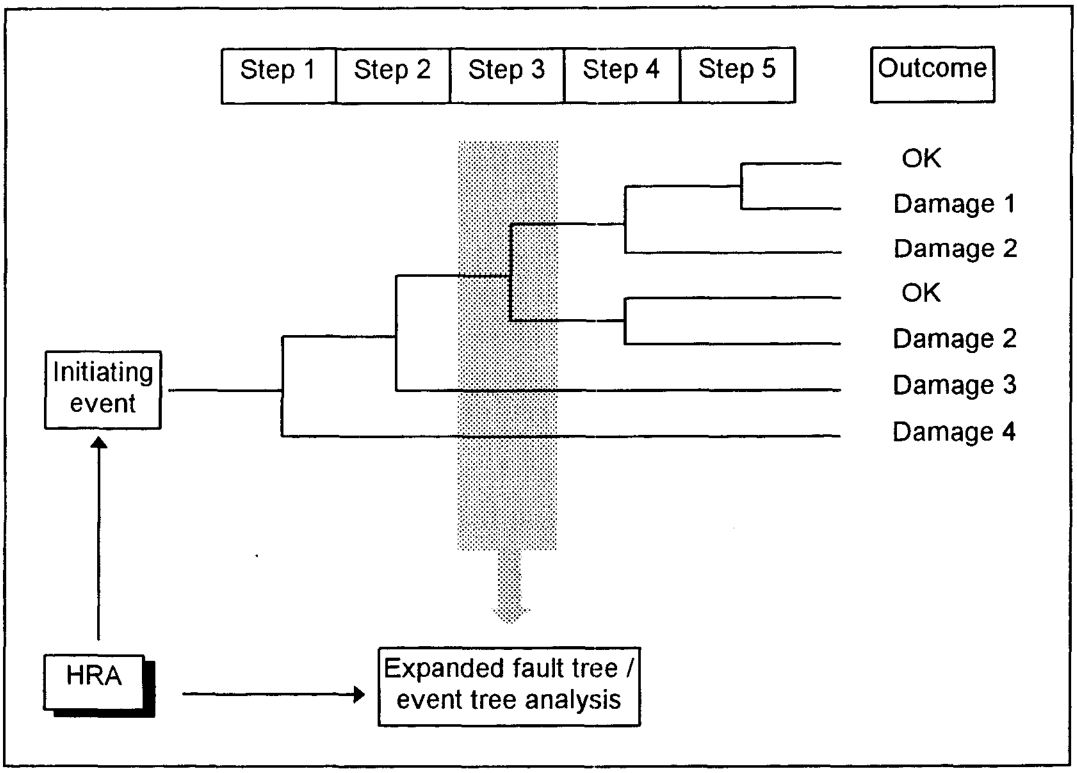

Figure 1: Typical PSA event tree, with links to HRA.

Any analysis of a technical system must be based on a simplified representation of the system and the events that can occur, for purely practical reasons. As long as technical systems were only loosely coupled (Perrow, 1984) it was defensible to base the analysis on a discrete - and static - representation of system states. However, for systems where the coupling between sub-systems and components is tight rather than loose, and where furthermore the interactions are complex rather than linear, there is a need to include consideration of system dynamics in the analysis. This specifically means that it is necessary to base the analysis on a representation that retains the essential features of couplings and interactions, that is, an analysis that looks at sequences rather than single events.

Technical systems that depend on MMI to accomplish their function, such as nuclear power plants, are typically tightly coupled with complex interactions. Specifically, the following four observations are important (Siu, 1990, p. 360):

- Plant operator and plant components are interacting parts of an overall system that responds to upset conditions.
- The actions of operators are governed by their beliefs as to the current state of the plant.
- The operators have memory; their beliefs at any given point in time are influenced (to some degree) by the past sequence of events and by their earlier trains of thought.
- A number of operators (more than one) are involved during an accident.

Man-machine systems are furthermore often very complex in structure with a large number of components and sub-systems. For these reasons the need of a dynamic analysis is particularly strong. The traditional approach has been based on decomposition of human actions, similar to the way in which accident sequences have been decomposed into their constituent events or stages (Hollnagel, 1993). This, however, makes it difficult for the analysis to consider the **context** in which the actions take place, and imposes a disregard of the fact that human actions never are independent of each other.[1] Human action is, however, fundamentally intentional and the intention depends on the perceived context. MMI, furthermore, is the functioning of the **joint system**, which cannot be adequately understood if each part is considered by itself. Altogether this means that risk and reliability analyses of systems that involve man-machine interaction need to go beyond the currently available static approaches.

2.2 Monotonic And Non-Monotonic Analysis

The current analyses in PSA/PRA/HRA are principally **monotonic**. This means that the overall structure of the sequence is defined **before** the analysis starts, and that it is not modified or updated during the analysis. The initial conditions are frozen at the beginning, and the analysis assumes that changes do not occur during the time that is covered by the analysis (cf. Figure 2). This approach, by the way, is also typical of classical decision theory and game theory.

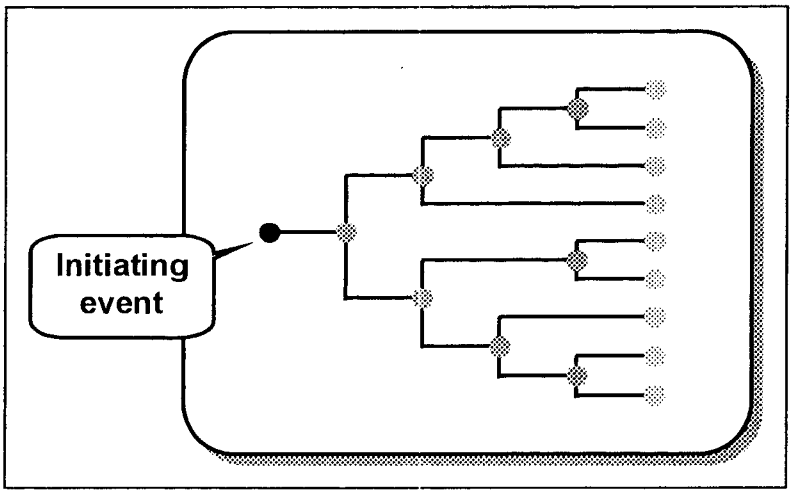

Figure 2: Principles of monotonic event analysis.

It is not difficult to see that a monotonic analysis is insufficient to account for a dynamic system, such as a man-machine system. In fact, PSA/PRA/HRA investigate long sequences of events or incidents where significant changes in the system take place. It stands to reason that these changes should be reflected in the way the analysis is made. A monotonic analysis is inadequate for MMI, because:

[1] Most methods try to compensate for the lack of context by introducing Performance Shaping Factors of various kinds. These are, however, applied *post hoc* rather than *ante hoc*, and are generally a poor substitute for context.

- human actions may change the **configuration** of the system, hence the propagation pathway,
- humans respond to the **current** situation,
- the sequence of human responses is not fixed, but will be **adapted** to the conditions,
- humans do not only react, but also do things **proactively** in anticipation of future events.

To address these issues it is necessary to reproduce the event tree after each step of the analysis, to reflect that changes that have taken place. This principle is illustrated in Figure 3. Clearly, if such an undertaking is to be done manually it is only feasible if the event tree is very small. In all other cases it is necessary to consider the use of computerised support, either to improve the event tree method or by introducing a joint system simulation.

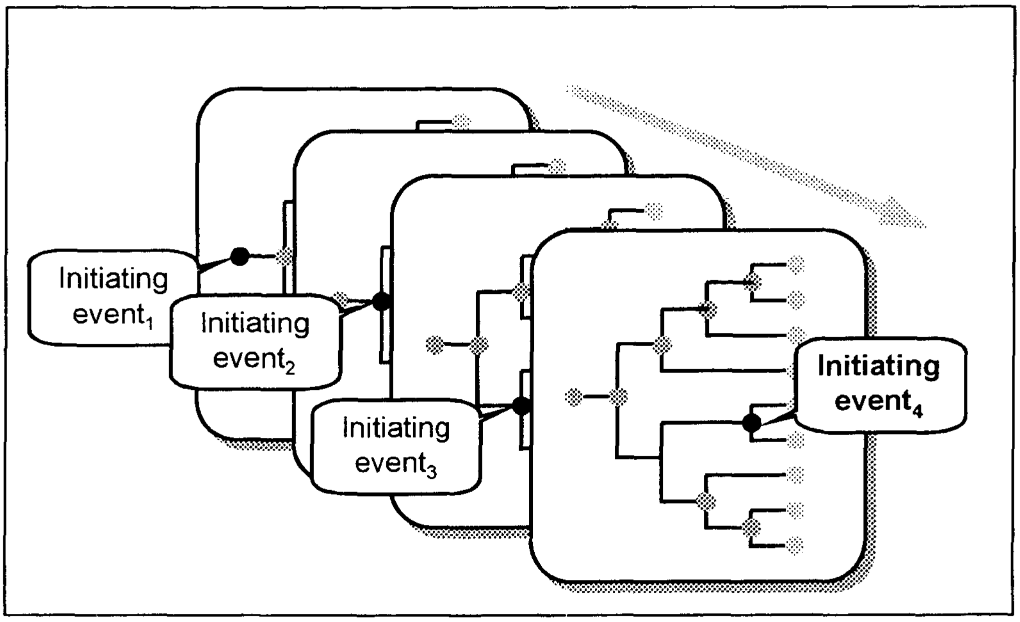

Figure 3: Principles of non-monotonic event analysis.

The event tree can be improved by making it dynamic. Conventional event tree methods illustrate the static aspects of the system and the interactions, and cannot explicitly account for changes in process variables or for human performance. Since the event tree approach has been accepted as an industrial *de facto* standard, it makes sense to consider how the event tree can be improved by various techniques such as state transitions and temporal conditions (Siu, 1994). These extensions of the event tree do, however, not consider the operator and the process as separate entities. Rather, the possible operator reactions are factored into the event tree. Although this is more sophisticated than the current approaches, it is still seen as a step-wise improvement rather than as a new solution.

Non-monotonic analyses can be of two kinds: semi-dynamic analyses and fully dynamic analyses. In a semi-dynamic analysis the updating of the system state is done **off-line** in discrete steps, either manually or with the help of simulations or calculations. As in the monotonic analysis, a detailed event tree is prepared in advance. But for each step of the analysis, the event tree is modified if necessary, based on the analyst's assessment of the results from the previous step. This corresponds to a discrete, off-line simulation, hence does not require a dynamic model. The event tree must be redrawn a number of times, but only for the parts that have not yet been reached.

In a fully dynamic analysis, the updating of the system state is done **on-line**, using coupled simulations or calculations. (The simulations themselves may, however, be either continuous or discrete.) In this case detailed **initial conditions** for all parts of the system are prepared before analysis, but there is no need to develop an event tree. Specific events (branch points) that may occur are also specified, together with their triggering conditions. System performance is then simulated, by alternating between process simulation and operator simulation. An actual event tree (performance path) may be constructed after the analysis, although this would correspond more to a time-line than to an event tree.

Another way of characterising the two types of analysis is by noting that a monotonic analysis represents an **open loop** approach, while a non-monotonic analysis represents a **closed loop** approach. The monotonic analysis must specify everything in advances, and cannot take intermediate results into account. The non-monotonic analysis, and in particular the joint system simulation, makes use of the feedback from each step of the analysis to adjust the system descriptions.

2.3 Joint System Simulation

A joint system simulation is a dynamic (non-monotonic) analysis that makes use of two simulations coupled together: a simulation of the technical system (the process) and a simulation of the operator or operators controlling the process. A joint system simulation simulates the development of the interaction between the two systems, given the initial conditions and the system characteristics. It can be used to analyse both the effects of design, possible initiating events, and failure response potential. It can, in particular, be used to determine:

- whether the joint system can accomplish its functions,
- whether an initiating event will lead to an incident,
- how the MMI will react to an initiating event, hence what the response potential is.

The due consideration of these factors requires a robust architecture that can be used to represent and investigate the dynamics of the man-machine interaction. Considering the experience from a number of previous projects, in particular the development of the System Response Generator (Hollnagel & Cacciabue, 1992; Hollnagel, Cacciabue & Rouhet, 1992) a generic architecture can be proposed as shown in Figure 4. There are four main components plus the log, which is a facility for recording the outcome of an analysis.

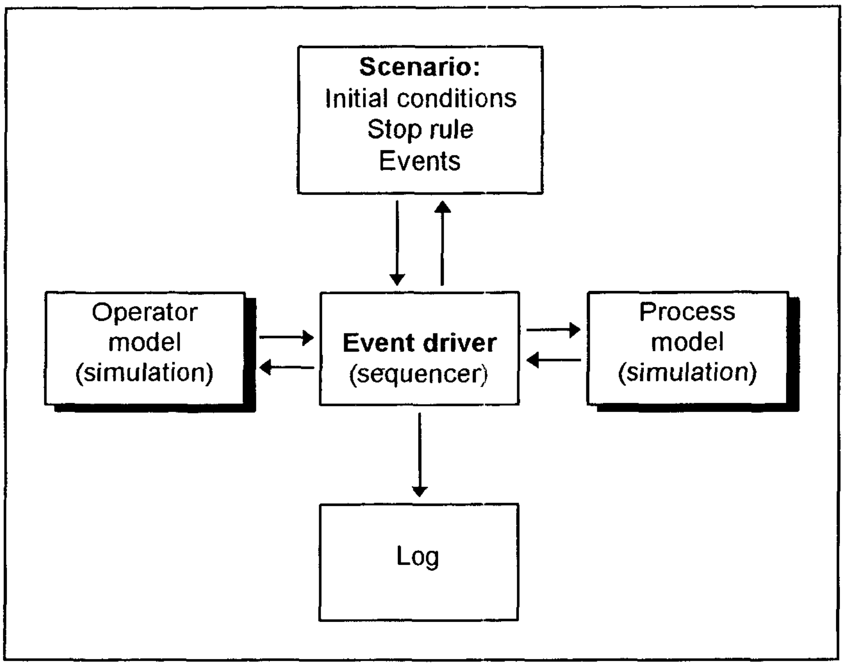

Figure 4: Generic architecture of a joint system simulation.

- The **scenario** describes the situation being investigated. It could, for instance, be an accident sequence. The scenario must define the initial conditions for the two models, the events that can occur during the scenario (for example failures, external events), and the stop rule or the conditions for ending the run.

- The **process model** is a simulation of the process being considered, for example a simulation of the relevant parts of a nuclear power plant. The process models reproduce the dynamics of the process as well as the responses to external events, component failures, and operator actions. The process model must also include the interfaces and control mechanisms through which the operator can interact with it.

- The **operator model** is a simulation of the operator, typically as a cognitive model. The operator model determines how the simulated operator interprets changes in the process and how responses are produced. The operator model should also produce reactions to external events and - ideally - the communication and interaction between operators in a team.

- The **event driver** is a "mechanism" or procedure that handles the **interaction** between the operator and the process, that is between the two simulations. The event driver controls the progress through the scenario, interacts with the two simulators, and produces the log of the analysis.

The generic architecture shown in Figure 4 does not show any details of the four main components. Clearly, for any given application the amount of detail provided by the two simulators is essential. Whereas process simulation is a well-developed technique, insofar as special languages have been developed for it, operator simulation is still something of an art. In the case of a process simulation there is little uncertainty of what should be simulated and how.

In the case of operator simulation there is considerably freedom, and the choice may depend on individual preferences.

To prevent the operator model from becoming too variegated, it is useful to observe the simple principle that it should support a comprehensive classification scheme that can maintain the coherence between the various elements of the man-machine system. In other words, the operator model should primarily serve the needs of the joint system simulation. If the model is based on a complicated psychological theory, the result may become too complex for the specific purpose, that is, the model may include functions that are not necessary for the joint system simulation as such.

For practical reasons, the generic architecture must also be supplemented by: (1) a comprehensive field evaluation of the working conditions and of the system control possibilities and (2) a method for application at the engineering level. The field evaluation is necessary to develop and describe the actual manifestations of operator behaviour and system performance that must be reproduced by the simulations, that is, the **data** that support the models. The application method enables the user to combine the different models and data structured according the objective of the analysis, that is, it defines the **method** in use. A distinction can here be made between **retrospective** studies of root cause analysis and **predictive** evaluations of possible outcomes of accidents, including human erroneous behaviour and system failures. In this case, different levels of complexity are envisaged, as far as the models of the system and of the human behaviour are concerned. However, only the predictive / predictive application will be considered here.

3. SURVEY OF REPRESENTATIVE SYSTEMS

The purpose of the survey that follows, is to characterise the representative proposals of joint system simulation that are available in the public domain. The descriptions should enable the reader to get a rough idea about what each particular system does. Detailed operational or technical information will not be provided in this survey, due to limitations of time and space. Since this work is carried out in the context of NKS/RAK-1, an additional purpose is to characterise each of the surveyed systems concerning how well it is suited for integrated sequence analysis and how well or how easy it is to interface it with a PSA/PRA. Altogether, this leads to the following descriptors:

- Acronym / name.
- Intended application.
- Components.
- Process model.
- Operator model.
- State of development.
- Developer.
- Contact person.

The survey will only consider systems that are genuine joint system simulations, that is, systems that clearly include a model of the process and a model of the operator. It is, however, also necessary to mention the original DYLAM (Dynamic Logical Analytical Methodology; Cacciabue & Amendola, 1986), because it has been very influential for the field as a whole. It represents the solution of the dynamic event tree, where the propagation through the tree is determined by a dynamic simulation of events, including probabilistic failures of components. DYLAM has been developed at the JRC Ispra since about 1984, and has been described in several reports and conference presentations. In the current versions of DYLAM, the event trees have been enhanced with an operator model based on the step-ladder decision model.

A more recent system, which has taken these ideas a step further, is DETAM (Dynamic Event Tree Analysis Method; Acosta & Siu, 1994). DETAM treats plant process variables deterministically, but allow for stochastic variations in both hardware states and crew states. The latter are represented in relation to, e.g. diagnosis and planned actions. Although DETAM does take the possible effect of operator actions into account, it is done by way of the event tree rather than by means of a joint system simulation. DETAM is therefore not include in the survey.

The following survey presents the systems in alphabetical order.

3.1 CAMEO - Cognitive and Action Modelling of an Erring Operator

Developer: Mitsubishi Atomic Power Industries - MAPI (since 1992).

Intended application: To establish a framework with which an integrated view on human error inducing mechanisms can be obtained in engineering terms. Also as a task analysis tool.

Components: Knowledge-base, memory, information processing (perception, attention, decision making, action). CAMEO is coupled to a process simulator.

Process model: Target system modeller; a flexible modelling scheme implemented through a high-level language (G2). Demonstrations have been made using a schematic water supply plant.

Operator model: Working memory, long-term memory, perception/recognition module, decision making module, action module, attention/resource controller.

State of development: CAMEO has been developed as a prototype and has been demonstrated for the water supply system.

Contact person: Yushi Fujita, MAPI, 4-1, 2-Chome, Minato-ku, Tokyo 105, Japan.

3.2 CES - Cognitive Environment Simulation

Developer: Westinghouse R&D Center (1986 - 1990).

Intended application: To enhance the measurement of human contribution to risk in PRA studies. To simulate the processes that determine situation assessment and intention formation (errors of commission).

Components: Dynamic plant simulator, virtual; display, CES. The coupling between CES and the plant simulator is only automated in the direction from the simulator to CES.

Process model: CES is developed specifically for nuclear power plant applications. The process model is represented by snapshots, but CES can in principle be linked to a dynamic process simulation.

Operator model: CES contains a knowledge base and a set of processing mechanisms (monitoring, explanation building, response management).

State of development: The CES was developed with funding from the NRC. The development was completed in 1990, although some additional studies have been made.

Contact person: Emilie Roth, Westinghouse Research and Development Center, 1310 Beulah Road, Pittsburgh, PA 15235, USA.

3.3 COSIMO - Cognitive simulation Model

Developer: JRC Ispra (since 1987).

Intended application: COSIMO was developed as an extended human model for DYLAM, but is usually referred to in its own right. The purpose of COSIMO is to simulate how operators manage complex environments/processes.

Components: "Low" level cognition (information processing functions) controlled by "meta" cognition (rule frames knowledge frames).

Process model: COSIMO can be interfaced to various process models, but requires a detailed specification of the interface.

Operator model: Conceptually, COSIMO consists of a knowledge base and a working memory. The former contains both declarative and procedural knowledge. The latter manages the flow of events and processes, linking to the knowledge base via "calling conditions". The knowledge base uses the heuristics of similarity matching and frequency gambling as described by J. Reason. Computationally, COSIMO is implemented by a blackboard architecture.

State of development: COSIMO has been developed as a prototype and demonstrated. It has also been used in a field application. The development is apparently no longer continuing.

Contact person: Pietro C. Cacciabue, ISEI, JRC Ispra (Va), I-21020 Varese, Italy.

3.4 MIDAS - Man-machine Integration Design and Analysis System

Developer: Joint Army - NASA programme (since 1984).

Intended application: Predictive model of human performance combined with a CAD tool for MMI design

Components: Model of operating environment, equipment, and operator (perception, attention, memory, cognition). User interface for inputs and outputs, i.e., links between user model and system.

Process model: Physical component models, e.g. aerodynamic & guidance models.

Operator model: Human performance representation (competence, capabilities), physical (anthropometric) representation, perception and attention, updatable world representation, activity representation, scheduler, task loading model.

State of development: MIDAS has been developed into a software tool that runs on a single machine (e.g. SG IRIS). It has successfully been demonstrated, and applied to a number of real-life studies. Development is ongoing. The literature describing MIDAS is, however, limited, due to the nature of the project.

Contact person: Kevin M. Corker, NASA ARC, Moffet Field, CA 94035.

3.5 OASYS - Operability Assessment System

Developer: BBN for Air Force Systems Command, Wright Patterson AFB, OH, USA.

Intended application: OASYS is a software system to support the investigation of man-system allocation, automation-crew composition, and human-computer integration issues. It is intended to be used throughout the system life-cycle.

Components: OASYS contains a number of software tools for (1) system and workstation design, (2) system and workstation simulation, (3) operability experiment design, (4) experiment data collection, (5) documentation, and (6) database access.

Process model: The process simulation is assembled from a number of "primitives" that represent events in the target system which can be observed by the operator. OASYS is also designed to be interfaced with a real simulator.

Operator model: The operator model is a performance model consisting of semi-autonomous centres (holons) in a hierarchy. It is implemented by an actor programming model.

State of development: OASYS appears to be a conceptual design, and the state of implementation is unknown. Literature about OASYS is restricted, due to the military funding.

Contact person: Michael J. Young, Armstrong Laboratory, Wright Patterson AFB, OH 45433-6563, USA.

3.6 SRG - System Response Generator

Developer: CRI A/S, APSYS, JRC Ispra (1991-1994).

Intended application: Identify MMI problem areas, evaluate effect of specific design changes, evaluate joint system performance.

Components: Control mechanisms to perform joint simulation, interfaces to simulators. SRG also contains a low level operator model.

Process model: The SRG has been developed as a professional software tool and can therefore easily be interfaced with various process simulators. Demonstrations have been made using two different types of simulators from the aerospace domain.

Operator model: The operator model is a low level cognitive model, based on production rules. The SRG, however, does not have an integrated operator model but can be interfaced to any available model.

State of development: The SRG was completed as a software tool in 1993. The emphasis was on the control of the joint simulations, rather than on the simulations themselves. The SRG development is has not been continued.

Contact person: Pietro C. Cacciabue, ISEI, JRC Ispra (Va), I-21020 Varese, Italy. Erik Hollnagel, HRA Ltd., UK.

3.7 SYBORG - Simulation System for Behaviour of the Operating Group

Developer: Central Research Institute of Electric Power Industry - CRIEPI (since 1993).

Intended application: To simulate the behaviour of a team coping with an abnormal event. To evaluate effectiveness of human error countermeasures.

Components: Plant model, MMI model, team behaviour model (several cognitive models + human-human interaction model).

Process model: Simplified nuclear power plant, as a dedicated simulator.

Operator model: Individual operator model contains: (1) attention model, (2) thinking model, (3) action model, (4) utterance model, (5) memory models (STM, MTM, LTM). The operation team model contains a number of individual; operator models, plus a human-human interface model. The link to the process model is through an MMI model.

State of development: The basic team model has been developed. SYBORG is funded by a long-term research programme, going beyond 1997.

Contact person: Ken'ichi Takano, CRIEPI, 11-1, Iwato Kita 2-Chome, Komae-shi, Tokyo 210, Japan.

4. SUMMARY AND RECOMMENDATIONS

As the preceding survey shows, the available examples of joint system simulations are quite varied and have been developed for many purposes. It is, however, encouraging to note that despite the relative newness of the field, most of the cases have been developed with a practical aim in mind, rather than as pure research tools.

To summarise the survey, the seven systems are characterised in Table 1 below, using slightly different descriptors.. The three last columns of Table 1 describe the degree of relevance to PSA/PRA/HRA, how flexible the system is, that is, how easy it will be to apply it to another application, and how mature it is, that is, how far it has come in its development.

Table 1: Summary of surveyed joint system simulations.

Name	Purpose	PSA/PRA/HRA relevance	Flexibility	Maturity
CAMEO	Analysis of human error mechanisms	Medium	Low	Low
CSE	Operator modelling for PSA, focusing on commissions.	High	Low	High
COSIMO	Simulation of operator cognition and management of complexity	Medium	Low	Medium
MIDAS	Predictive model for MMI design, emphasis on ergonomics	Medium	Medium	High
OASYS	MMI design support tool covering whole life-cycle	Low	Low (?)	Low (?)
SRG	General tool to support joint system simulation	Medium	High	High
SYBORG	Analysis of team communication and performance	Low	Low	Low

As this summary shows, none of the systems are in a completely ready state to be used for dynamic sequence analysis. Some of the systems are relevant for PSA, and one has been built with PSA/HRA in mind; others have an acceptable degree of flexibility, although this is no indication of the amount of effort it actually will take to reconfigure them; and some are fully developed systems that are safely beyond the prototype stage. Unfortunately, there is not one of them that possess all the virtues at the same time.

The survey nevertheless shows that the principles of joint system simulation are quite well developed. A closer scrutiny of the systems reveals that they all comply in principle with the generic architecture described above. This is, perhaps, not very surprising, since it is a fairly obvious way of approaching the problem.

4.1 Recommendations For NKS/RAK-1

The use of joint system simulation for integrated sequence analysis offers some obvious advantages. Primarily, it is an effective way to overcome the fundamental limitation of static, manual analyses. A joint system simulation does not require the elaboration of an explicit event tree, but uses instead a specification of initial conditions and likely events, described, for example by their triggering conditions. This means that the ensuing analysis is not limited by the possibilities that have been included in the event tree, although it is limited by other things, cf. below. A joint system simulation can be used not only for sequence analysis, but will also have applications for design evaluation, training, etc. It may also serve as a vehicle for a second generation HRA.

These advantages do not come without any costs, unfortunately. In particular, the quality of the output from a joint system simulation depends on the quality of the constituent models. (However, there are no methods that are not limited by this condition.) Developing a joint system simulation may require a substantial amount of work in specifying the knowledge needed by the two models and the interface between them. This initial cost may, however, easily be outweighed against the savings in making the analysis and the ease by which multiple analyses can be made, for example by making small modifications to the scenarios. In simulation technical terms it may be an issue that the simulation is based on discrete time steps rather than being fully continuous. The main obstacle for that is the difficulty in synchronising the operator model with a real-time process model. Finally, it must be realised that the developments of techniques for joint system simulation still are at an early stage, and that the work mainly has taken place within two domains: nuclear power plants and aviation.

On the basis of these conditions, the following recommendations can - cautiously - be made.

First, integrated sequence analyses should be dynamic rather than static. The shortcomings of static analysis methods are serious, even for the analysis of single events. When sequences are considered, and in particular when MMI is included, it is necessary that the analyses are dynamic.

Second, the use of joint system simulations is preferable to the use of dynamic event trees. Although dynamic event trees probably are simpler to introduce, they will in the long run limit the analysis. The reason is that a dynamic event tree does not clearly separate process events and operator events, since they both are represented as nodes in the event tree. This means both that operator events are restricted to a few simple categories, and that the event tree may be difficult to modify.

Third, scenarios should be considered as an alternative to event trees. The difference between the two is that a scenario describes conditions while an event tree describes the instantiation of the conditions. Although the graphical representation of an event tree is immensely valuable as an aid to understanding, it also has some serious practical limitations. A scenario description, on the other hand, can represent many potential event trees. It is only when the scenario description is realised that a specific event tree - or rather an event sequence - comes to life. A scenario description is also more meaningful for considering MMI than an event tree is. Cognitive models generally relate to a context rather than to an event sequence..

Fourth, efforts should be spent on defining the principles of modelling rather than of running and controlling the joint system simulation. The generic architecture for this technique is well developed and understood, and can easily be implemented. (For instance, both MIDAS and the SRG are documented with detailed software specifications - although they are not publicly available.) The problems are in developing the models and in defining interfaces, for example the knowledge or assumptions that must be represented in the system.

Finally, it is necessary to develop robust principles for operator modelling (or cognitive modelling). Even if it turns out to be impossible within the project to implement an actual operator model, the detailed description of the principles may enable a "manual" simulation, for example in a Wizard-of-Oz technique[2]. That will make it possible to apply the principles of joint system simulations without necessarily implementing a complete system.

[2] The Wizard-of-Oz technique refers to a situation where a specific set of functions are performed by a person rather than by e.g. a computer or a knowledge-based system. It requires that specific instructions are given, but avoids the efforts needed to implement a system. It is therefore a quick and efficient way of determining whether a principle will work. The meaning of the term is that a significant effect is achieved by simple means, almost by "cheating". The origin of the term is the film The Wizard Of Oz (1939).

5. REFERENCES

Acosta, C. & Siu, N. (1993). Dynamic event trees in accident sequence analysis: Application to steam generator tube rupture. *Reliability Engineering and System Safety*, *41*, 135-154.

Dougherty, E. M. Jr., & Fragola, J. R. (1988). *Human reliability analysis. A systems engineering approach with nuclear power plant applications.* New York: John Wiley & Sons.

Cacciabue, P. C., Amendola, A. & Cojazzi, G. (1986). Dynamic logical analytical methodology versus fault tree: The case of the auxiliary feedwater system of a nuclear power plant. *Nuclear Technology*, *74(2)*, 195.

Gertman, D. I. & Blackman, H. S. (1994). *Human reliability & safety analysis data handbook.* New York: John Wiley & Sons, Inc.

Hollnagel E., (1993). *Reliability of Cognition: Foundations of Human Reliability Analysis.* Academic Press, London.

Perrow, C. (1984). *Normal accidents: Living with High-Risk Technologies.* New York: Basic Books.

Siu, N. (1994). Risk assessment for dynamic systems: An overview. *Reliability Engineering and System Safety*, *43*, 43-73.

5.1 Selected Literature

CAMEO

Fujita, Y., Yanagisawa, I., Itoh, J., Yamane, N., Nakata, K. & Kubota, R. (1993). *Modelling operator with task analysis in mind.* ANS Top. Meeting on NPP Instrumentation, Control and Man Machine Interface Technologies, Illinois.

Fujita, Y., Sakuta, H. & Yanagisawa, I. (1993). *Human reliability analysis using simulated human model.* 2nd International Conference on Probabilistic Safety Assessment Methods, San Diego, CA, March 21-25, 1994.

CSE

Roth E. M., People Jr. H. E., Woods D. D., (1991). Cognitive Environment Simulation: a Tool for Modelling Operator Cognitive Performance during Emergencies. In Apostolakis, G. E. (Ed.), *Proc. of the Int. Conf. on Probabilistic Safety Assessment and Management (PSAM)* 4-7, February 1991, Beverly Hills, California. New York, NY: Elsevier.

Woods, D. D. Roth, E. M., Pople, Jr. H. E. (1987). *Cognitive Environment Simulation: An Artificial Intelligence System for Human Performance Assessment.* NUREG/CR-4862.

Woods, D. D., Roth, E. M. & Pople, H. Jr. (1988) Modeling human intention formation for human reliability assessment. In G. E. Apostolakis, P. Kafka, & G. Mancini (Eds.), *Accident* *sequence modelling: Human actions, system response, intelligent decision support.* London: Elsevier Applied Science.

COSIMO

Cacciabue, P. C., Decortis, F., Drozdowicz, B., Masson, M. & Nordvik, J. P. (1992). COSIMO: A Cognitive Simulation Model of Human Decision Making and Behaviour in Accident Management of Complex Plants. *IEEE Transaction on Systems, Man and Cybernetics, IEEE-SMC,* **22** (5), 1058-1074

MIDAS

Corker, K. M. & Smith, B. R. (1993). *An architecture and model for cognitive engineering simulation analysis*: Application to advanced aviation automation. AIAA Computing in Aerospace 9 Conference, October 21, 1993, San Diego, CA.

Statler, I., C. & Corker, K. (1993). *Activities at the NASA Ames Research Center on Aeronautical Human Factors.* Proceedings of Fourth International Conference on Human-Machine Interaction and Artificial Intelligence in Aerospace, 28-30 September 1993, Toulouse, France.

OASYS

Young, M. J. (1993). *Human performance models as semi-autonomous agents.* 4th Annual Conference on AI, Simulation, and Planning in High Autonomy Systems, IEEE Computer Society Press.

SRG

Hollnagel, E. & Cacciabue, P. C. (1992). *Reliability Assessment Of Interactive Systems With The System Response Analyser.* European Safety and Reliability Conference '92, Copenhagen, 10-12 June, 1992.

Hollnagel, E., Cacciabue P. C. & Rouhet J.-C. (1992). *The use of an integrated system simulation for risk analysis and reliability assessment.* 7th International Symposium on Loss Prevention, 4-8 May, 1992, Taormina, Italy.

SYBORG

Sasou, K., Yoshimura, S., Takano, K., Iwai, S. & Fujimoto, J. (1993). *Conceptual design of simulation model for team behaviour.* In E. Hollnagel & M. Lind (Eds.), Designing for simplicity. Proceedings of 4th European Conference on Cognitive Science Approaches to process Control, August 25-27, Hillerød, Denmark.