Component Configuration Control System: An Application of Logic Programming

Rex (Trav) C. Stratton

Computer Applications and Program Development
Experimental Breeder Reactor II Division
Argonne National Laboratory

George G. Town

Computer Science Department
Central Washington University

CONF-8510161--1

DE85 018455

ABSTRACT

A computer application system is described which provides nuclear reactor power plant operators with an improved decision support system. This system combines traditional computer applications such as graphics display with artificial intelligence methodologies such as reasoning and diagnosis so as to improve plant operability. This paper discusses the issues, and a solution, involved with the system integration of applications developed using traditional and artificial intelligence languages.

INTRODUCTION

Improving nuclear reactor power plant operability is an ever-present concern for the nuclear industry. The definition of plant operability involves a complex interaction of the ideas of reliability, safety, and efficiency. In this paper we present our observations concerning the issues involved and the benefits derived from the implementation of a computer application which combines traditional computer applications with artificial intelligence (AI) methodologies. A system, the Component Configuration Control System (CCCS), is being installed to support nuclear reactor operations at the Experimental Breeder Reactor II (EBR-II), located in Idaho and run by Argonne National Laboratory for the Department Of Energy.

As noted by Lay and Menke[1], currently in the United States, there is very little direct computer control of the reactor in nuclear power plants. The operator controls the reactor through a specified set of complex sequences of switch settings and valve manipulations, that is, a series of component configurations. The knowledge base required of the reactor operator is very extensive. The collection of components can be very large and form an elaborate network with many possible paths which provide a multitude of acceptable and unacceptable process functions [2,3,4]. During operation of the plant, the proper interpretation of the sensor readings requires that the operator have a thorough understanding of component relationships and the associated laws of physics and chemistry.

*Work supported by the U. S. Department of Energy, Reactor Systems, Development and Technology, under Contract W-31-109-ENG-38

In addition to knowing the physical relationships, the operator must be thoroughly familiar with Technical Specifications and administrative constraints. Technical Specifications are legal documents which list the conditions and sequence of component configurations which must be observed when operating the reactor. Administrative constraints are plant policy and are determined by management. Within these constraints there is considerable latitude for the operator to control the plant.

Unanticipated plant parameter excursions which approach Technical Specification boundaries are a major cause for the plant to be shut down which in turn results in less-than-optimal plant operation. A measure of plant efficiency is the plant capacity factor (PCF). It was reported in Nuclear News[5] that in the United States the average annual PCF ranged from 51% to 63%. Many of the unexpected shut-downs could be avoided if the operator had currently valid, pertinent presentation of plant parameters, associated "trajectories" of the parameters, and validated analysis of projected changes in component configurations.

Computer applications used in the reactor control room are nearly always limited to data collection, archiving and graphics display. It is not feasible to redesign the control system to provide more direct computer control. Thus, to improve plant operability the alternative is to provide the operator with a support system which will more effectly and directly support the decision processes. The general requirements are that the system provide:

1. reliable and consistent results (capable of proof of correctness and not subject to emotional stress);

2. flexible, effective operator interaction;

3. performance and responsiveness consistent with plant requirements;

4. effective presentation of current plant status (including current component status);

5. effective presentation of reliably projected plant parameters;

6. validated state (parameter, operational, and performance) readings;

7. validated knowledge base;

8. validated analysis and diagnosis of proposed changes in component status, relative not only to the physical requirements but also with respect to the Technical Specification and administrative Constraints.

The above specifications combine the more traditional application system requirements with requirements more closely associated with AI. Moreover, the component system can be completely described in a data base along with the rules of interaction so that a reasoning system can be used to provide validated results rather than heuristic methodologies. The system presented in this paper is a axiomatic inferential system, it is not an expert system[6].

THE SYSTEM

A system, the Man Machine Control System (MMCS), which addresses the concerns and specifications listed above, is being developed and tested at EBR-II[7]. This paper discusses a subsystem of the MMCS, the Component Configuration Control System (CCCS)[8], which provides assistance in the form of an analysis of proposed changes to the states of components in the plant and/or plant functional requirements and administrative constraints. The requirements and constraints are determined by system goals derived from the mode of operation. The CCCS has been designed to be generally applicable to other nuclear reactor power plants. Extensive use of computer graphics for both input and output provides for the human factors interface requirements and insures the input of reliable and consistent data. It was decided to use a Prolog implementation[10], where feasible, for the reasoning portion of the application since a very successful prototype of that portion was developed using the language [11].

A CONCEPT OF STATE

State is an abstraction and can be defined in general as a condition of existence relative to a defined set of circumstances. We define state on two levels of abstraction, parametric and symbolic. Parametric level is the lowest and is defined by parametric states. The symbolic level is divided into operational and performance states. States in the symbolic level are derived from parametric states. Both levels of state refer to the condition of existence of a physical system.

Parametric states are defined by physics units (dimensions) such as pressure, temperature, neutron flux, volts, mass, energy, and time. As an example, one might say that the energy state of an object is x ergs or that it's heat state is y calories. The parametric state is expressed as a numeric value attached to a physics dimension. Singular or multiple parametric states are used to derive operational and performance states.

The operational state expresses the condition or readiness of operation of an object. The operational state of a heat exchanger might be "on". This state is interpreted to mean that the heat exchanger is active and is available to provide the process of thermal reduction. However, this state makes no reference as to how well the heat exchanger is performing the process. The operational state is either assumed or derived using parametrical states.

The issue of performance is characterized by the performance state. The performance state describes the condition of performance of an object. The performance state of a heat exchanger might be "100%" which means that the heat exchanger is providing the process of thermal reduction and is performing at 100% of the rated design capacity. This state is determined by comparing the real-time parametric state of an object to the design parametric state of the object.

Operational and performance states can be summarized as follows; the operation state implies configuration (potential behavior) of an object and the performance state implies real-time behavior of an object. Presently the CCCS performs analysis of operational states. In the future the CCCS will be expanded to include parametric and performance state analysis.

DEFINITION OF THE CCCS

This section discusses the design functions of the CCCS. These functions provide for the system requirements as defined in the technical specification. The design functions are impact analysis(IA), analysis explanation(AE), and alternate solution determination(ASD).

IMPACT ANALYSIS(IA):

IA determines the impact of a selected component configuration with respect to a specified plant or system mode. The selected configuration is chosen in response to either an operations requirement (shift from 50% capacity to 75% capacity) or a maintenance requirement (replace the seal on the #2 feedwater pump). The system mode defines the process goal and subgoals that must be satisfied by the system given the selected configuration. The configuration is analyzed with respect to functional capability and administrative constraints, real-time or simulated plant state, and singular or multiple primary goals.

The selected configuration is first analyzed to determine its functionality and administrative constraint limitations in association with the process goals. Functionality defines the configuration's ability to provide process functions. Administrative constraints express the limitations imposed on configurations by the Technical Specifications (as defined in the FSAR) and plant administrative policy. It should be noted that a configuration can provide the necessary functions required by the mode goal and yet fail the analysis due to constraint violations.

Analyzing the configuration against these criteria allows the system to determine impact differently in emergency and non-emergency situations. In non-emergency situations a configuration must satisfy the functionality requirements without violating the full administrative constraint set. However, in the emergency condition the administrative constraint set is dynamically reduced with respect to the severity of the emergency and thereby shifting the emphasis of the analysis toward configuration functionality.

The IA function also provides analysis with respect to either real-time or simulated plant states. Real-time plant state analysis imposes additional state constraints. These constraints are a function of the real-time operational capability of the components and red-tag disposition. Real-time operational capability is characterized by the operational state of the component. These states are defined as operational, maintenance, and failed. Red-tags are danger tags placed on components that specify the state in which the component must remain and therefore limit the state space available to the component. Simulated plant state analysis imposes no state limitations on the components. If analysis is performed using simulated plant states then the complete set of component design states are available for operation.

Essentially, IA compares a selected configuration to mode requirements. The selected configuration is derived from operational and/or maintenance goals. The mode requirements, which define the goal processes, are produced by the system/plant design. The selected configuration specifies a desire and the mode requirements specify physical/functional capability. Plant operation is classified in a natural set of modes, eg. operation, maintenance refueling, and testing. These modes are further divided into submodes, sub-submodes, and etc. In some cases, modes share processes and exhibit a certain amount of dependence. IA requires that a mode requirement be assigned to a selected configuration prior to impact analysis. Realizing

that there is a natural set of mode descriptions and that these modes can have integrated processes, IA allows the selected configuration to be assigned to multiple modes. This provides an analysis with respect to multiple goals with specific attention paid to process intergration.

ANALYSIS EXPLANATION (AE):

It is human nature to doubt and question. The natural subsequent action when interfacing with a computing system that exhibits the capability to reason is to question the system's logic associated with its conclusions. The CCCS provides an explanation as to the rules, facts, and logic associated with the impact conclusions it derives. The explanation is presented hierachically and interactively. The system initially gives a general explanation as to the derivation of its conclusion. A more detailed explanation is given as a result of requests from the human element. In this interactive way, the human element can query the machine element as to its conclusion derivation.

The following will illustrate the analysis explanation. Suppose we have a process in a system that provides a pressure differential and the desired mode goal is "cooling". The pressure differential process consists of a suction valve, turbine, and a discharge valve. The states of these components are suction value closed, turbine off, and discharge open. Since these components are in the selected configuration, in their present states, and the assigned mode requires the pressure differential then the CCCS will conclude a negative impact. When asked to explain, the CCCS will respond with "Cannot be in cooling mode due to loss of flow", see Figure 1. Subsequent query by the human element will determine that the impact was derived as a result of no pressure differential process (turbine off) and an incomplete path(closed suction valve).

ALTERNATE SOLUTION DETERMINATION (ASD):

If the selected configuration for the required mode has an adverse impact, the CCCS will provide potential solutions to the human element. ASD is an interactive feature and is initiated by the human element. ASD derives the potential solutions by evaluating the mode inferential expression with respect to the selected configuration as defined by the mode requirements. The human iteratively queries the system until an acceptable solution is derived or a new strategy is desired.

ASPECTS OF ARTIFICIAL INTELLIGENCE

Knowledge in an AI system can be modeled as heuristic knowledge and/or axioms. Heuristic knowledge is the understanding of a phenomenon gained by experience and is usually manifested as rules of thumb. Heuristic knowledge does not have an explicit proof as to its existence and application to the phenomenon. AI systems that utilize heuristic knowledge are designated as expert systems [ref course]. Axiomatic knowledge is derived from the structural and behavioral characteristics of a phenomenon and provide an explicit proof as to its existence and application [dietmeyer]. AI systems that utilize axioms and logic will be designated as axiomatic inferential system. The CCCS is an axiomatic inferential system, the CCCS is not an expert system.

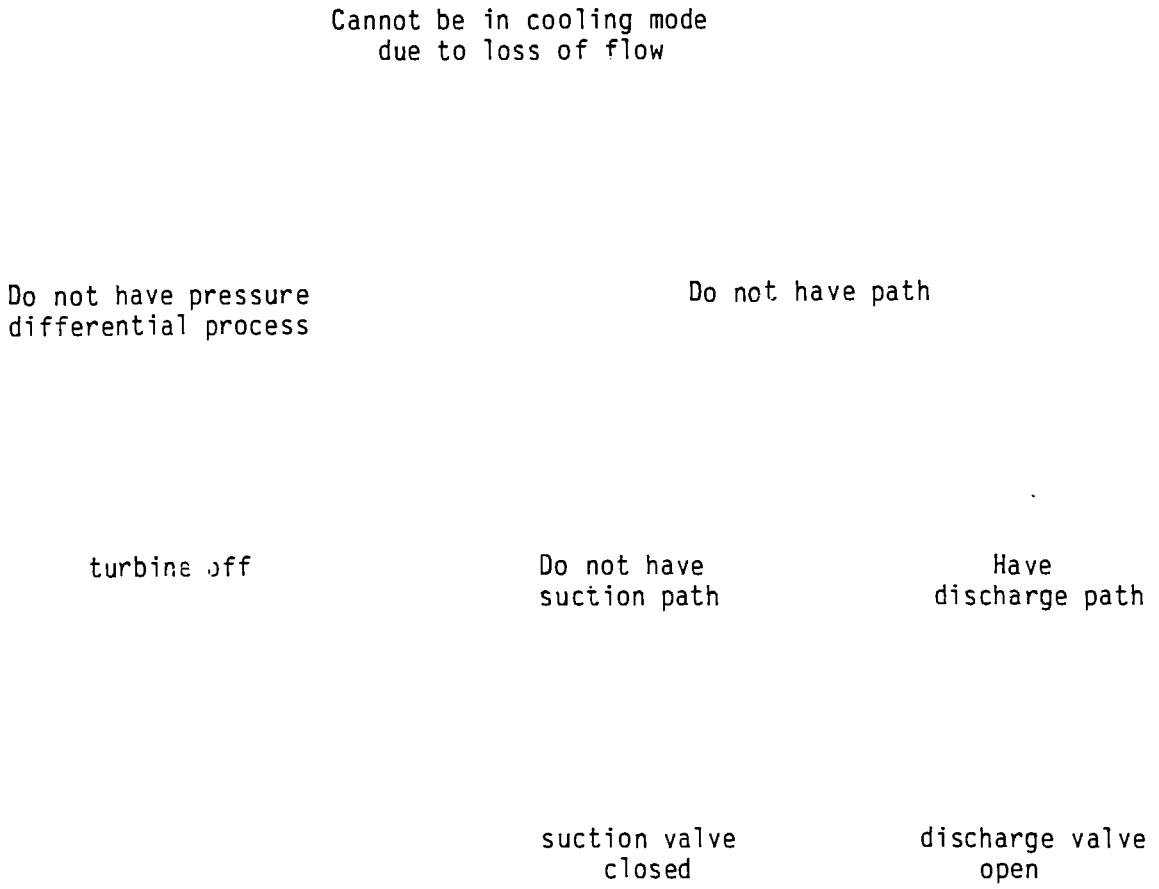

Figure 1. An Explanation Model

In order for the CCCS to perform its function, specific knowledge concerning the plant must be known. This knowledge is categorized as structural, behavioral, constraint, and real-time knowledge as discussed in the paragraphs above. This knowledge is modeled as rules and facts. The rules are expressed in Boolean logic and represent the equations of state from which the mode configurations are derived. Both the rules and facts are implemented as Horn Clauses [12]. Impact is reasoned about using resolution as provided by Prolog.

DISCUSSION

THE ISSUE OF TRANSPORTABILITY:

The CCCS is composed of two parts; a general purpose analyzer and a data base. The analyzer provides the control and reasoning constituents on the CCCS. This part of the system is completely transportable and can be implemented on any complex electro- mechanical system. The data base is specific to the electro-mechanical system it defines. Therefore, a facility desiring to implement the CCCS would transport the analyzer and build their own facility specific data base. We stress facility because the CCCS concept is not limited to just nuclear power plants, the concept can be applied to any electro-mechanical system. In fact we believe that any system built of objects that possess functional attributes and exhibit logical relationships can be analysed for functional-relational aspects using the CCCS.

THE ISSUE OF FUTURE EXPANSION:

The long term expansion plans are to develop the complete MMCS. The MMCS is a control system that performs the complete functions of control and operates at the inference level in a non-deterministic manner [13]. However, the immediate expansion plans are to provide performance state analysis, automatic reconfiguration determination, optimization of the criteria for ASD, and proof of correctness. Performance state analysis is discussed in the section A CONCEPT OF STATE.

Automatic reconfiguration determination would provide a target configuration subsequent to the occurrence of an off normal event. Given a situation where the plant is operating in mode X and component Z fails, the system would automatically determine a new configuration that satisfies the highest achievable goal objective. Goal objective would be determined from mode models. Procedure and configuration requirements would be determined from generalized and specific component control models based on function, path, boundary and constraints.

As discussed in the section Alternate Solution Determination, the purpose of ASD is to determine alternate configurations when the selected configuration imposes an adverse impact. ASD performs this function by determining the difference between the selected configuration and a mode configuration from the configuration set for the specified mode. The selection order of mode configurations is preestablished. Optimization would allow for the order to be dynamically determined as a function of real- time operating constraints and human guidance.

The validation or proof of correctness of computer programs is not a solved problem in general. Some limited results have been achieved with small programs and there are some general results available [14]. Logic programs are more amenable to validation by nature of their form and means of execution. However, one of the principal problems is the meticulous specification of the input data space, the associated data space of the program output, and the functional relationship of the input to the output. It is planned to develop an appropriate set of problem specifications which, in turn, will lead to the validation of the IA programs.

REFERENCES:

[1] R. K. Lay and J. L. Menke, "Reactor Operations: The Role for Computers", IEEE Transactions on Power Apparatus and Systems, Vol PAS-102, No. 11, November 183, pg. 3564-70.

[2] S. E. Seeman, R. W. Colley, and R. C. Stratton, "Optimization of the Man-Machine Interface for LMFBRs", Nuclear Safety, Vol. 24, No. 4, July-August 183, pg. 506-512.

[3] R. C. Stratton and S. E. Seeman, "A Computerized Operational Work Control System," ANS Transactions, TANSA039-1076, 1983, pg. 304.

[4] R. C. Stratton, "Automated Reasoning Application To Design Validation and Sneak Function Analysis", 1st Symposium on Space Nuclear Power Systems, January 8-13, 184, pg. sr-6.

[5] E. Michael Blake, "INEL Goes For A Piece Of The New Breeder Action", Nuclear News, Vol. 27, No. 13, October 184, pg. 50-52.

[6] L. Wos et al, "An Overview Of Automated Reasoning and Related Fields", Journal of Automated Reasoning, Vol. 1, No. 1, 185, pg. 6-13.

[7] L. R. Monson, R. C. Stratton, G. H. Chisholm, R. W. Lindsay, "Computer Application for Reactor Control and Diagnosis", 11th Bienneal Conference on Reactor Operating Experience, TANSAO 44 (Suppl. 1) (183), pg. 6.

[8] R. C. Stratton and L. R. Monson, "Component Configuration Control System at EBR-II", Proceedings of Joint ANS/ASME Conference on Design, Construction, and Operation of Nuclear Plants, TANSAO 46 (Suppl. 1) (184), pg. 46.

[9] R. W. Lindsay, G. H. Chisholm, and R. C. Stratton, "Potential Safety Enhancements To Nuclear Plant Control: Proof Testing at EBR-II", Proceedings of the American Power Conference, April 184.

[10] Quintus Corporation Prolog product description.

[11] R. C. Stratton and G. G. Town, Impact Analysis prototype notes and program, Jan. 1895.

[12] R. Kowalski, "Logic For Problem Solving", Elsevier North Holland, 182, Amsterdam.

[13] R. C. Stratton and G. G. Town, "Development of Logic Programming Methodologies to Reduce Demands on Computing Speed and Enhance System Diagnosis and Control" SDI/PDO-003 Proposal RAES-004, May 185.

[14] L. Wos et al, "Automated Reasoning Introduction and Applications", Printice- Hall Inc., pg. 314-370, 185.

DISCLAIMER