LA-UR- 96-2090

CONF-961192--2

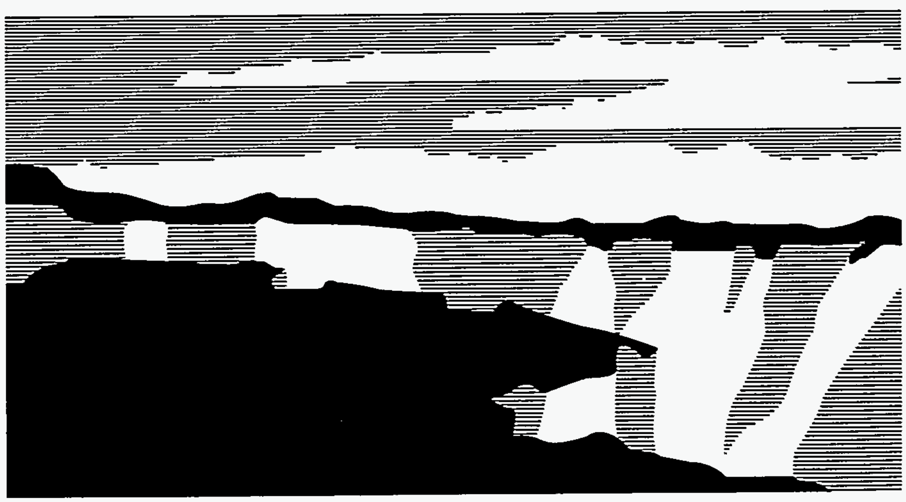
Title: METHODOLOGY, STATUS, AND PLANS FOR DEVELOPMENT AND ASSESSMENT OF THE TRAC CODE

RECEIVED
JUL 19 1996
OSTI

Author(s): B. E. Boyack, R. A. Nelson, S. Jolly-Woodruff

Submitted to: OECD/CSNI-transient Thermal-Hydraulic & Neutronic Codes
Annapolis,MD, November 1996

MASTER



Los Alamos
NATIONAL LABORATORY

Form No. 836 R5
ST 2629 10/91

DISCLAIMER

LA-UR- 96-2090

METHODOLOGY, STATUS, AND PLANS FOR DEVELOPMENT AND ASSESSMENT OF THE TRAC CODE*

by

B. E. Boyack,** R. A. Nelson,** and S. Jolly-Woodruff†

Abstract

The Transient Reactor Analysis Code (TRAC) is a state-of-the-art, best-estimate, transient system analysis computer code for analyzing geometrically complex multidimensional thermal hydraulic systems, primarily nuclear reactor power plants. TRAC is used by government and industry organizations for design and safety analysis, phenomenological studies, operational transient analysis, evaluating emergency operating procedures, simulator support and operator training, and for assessment of data involving basic experiments, separate effects tests, and plant operations. TRAC will calculate one- and three-dimensional (rectilinear and cylindrical coordinates) fluid flow involving gas, liquid, and mixture states. Although TRAC has many capabilities, it also has limitations. Some limitations arise from its implementation, dating from the 1970s. Rapid advances in hardware and software engineering highlight TRAC's inefficiencies; however, other limitations relate to the level of scientific knowledge regarding two-phase flow physics. These limitations will continue until such time as the fundamental understanding of two-phase flows is extended. Presently, several development activities are either in progress or soon to begin that will fundamentally improve TRAC. Foremost among these are reimplementation of the current TRAC data structures in Fortran 90 and the integrated development of closure packages for large-break loss-of-coolant accident applications.

INTRODUCTION

The Transient Reactor Analysis Code (TRAC)[1-4] is an advanced, best-estimate computer program that calculates the transient reactor behavior of a pressurized water reactor (PWR). In the early 1980s, the Nuclear Regulatory Commission (NRC) branched the development of the boiling water reactor (BWR) version of TRAC off the main version of TRAC. All the BWR versions begin with the designation TRAC-B, and the PWR versions begin with the designation TRAC-P. The development of TRAC-B began at Los Alamos, but is currently being developed at the Pennsylvania State University. In September 1995, the NRC announced plans to consolidate the two TRAC versions along

---

* The work reported in this paper was funded by the U. S. Nuclear Regulatory Commission and the US Department of Energy.

** Technology and Safety Division, Los Alamos National Laboratory

† Ogden Environmental and Energy Services

certain features from RAMONA. While acknowledging the consolidation objective, subsequent discussion of TRAC in this paper will focus on TRAC-P.

As used in the remainder of this paper, TRAC refers to the latest PWR version, which has the official NRC designation TRAC-P 5.4.15, which was released in January 1996. Before then TRAC-P was known as TRAC-PF1/MOD2 (MOD2). It is the latest in a series of TRAC codes, including TRAC-PD2/ MOD1, TRAC-PF1, TRAC-PD2, TRAC-P1A, and TRAC-P1, the earliest publicly released version.

Code development must be guided by a vision. For TRAC this vision is built on the foundation of specific code design objectives and targeted applications. The TRAC *design objectives* are as follows: TRAC should (1) accurately model important light water reactor accident phenomena in current-generation and advanced-passive reactors, (2) deliver best-estimate predictions of accident progression, (3) have a practical running time, (4) be portable, maintainable, and extensible, and (5) be adaptable to other reactor types.

The *targeted applications* for TRAC are: (1) reactor safety analyses for both operating and planned reactors, (2) audits of licensee's calculations, (3) analyses of operating reactor events, (4) analyses of accident management strategies, (5) support for test planning and interpretation, (6) support for Probabilistic Risk Assessments (PRAs), (7) design analyses, and (8) nuclear plant training and I&C simulators.

Ultimately, the measure of TRAC or any computational tool is whether the tool fulfills its design objectives and can be used with confidence for its targeted applications. The determination of code adequacy assessment is, of necessity, an ongoing process. However, it is important that there be, at appropriate intervals, a more searching consideration of code adequacy. The OECD/CSNI Workshop on Transient Thermal-Hydraulic and Neutronic Codes Requirements* and associated activities is one such review.

In the remainder of this paper we will present TRAC-related information within the overall context of code adequacy. A code adequacy assessment is divided into two parts (Fig. 1). First, the adequacy of each closure model in the field equations is examined by considering its pedigree, applicability, and fidelity to appropriate fundamental or separate

* Held November 5-8, 1996, in Annapolis, Maryland, United States of America.

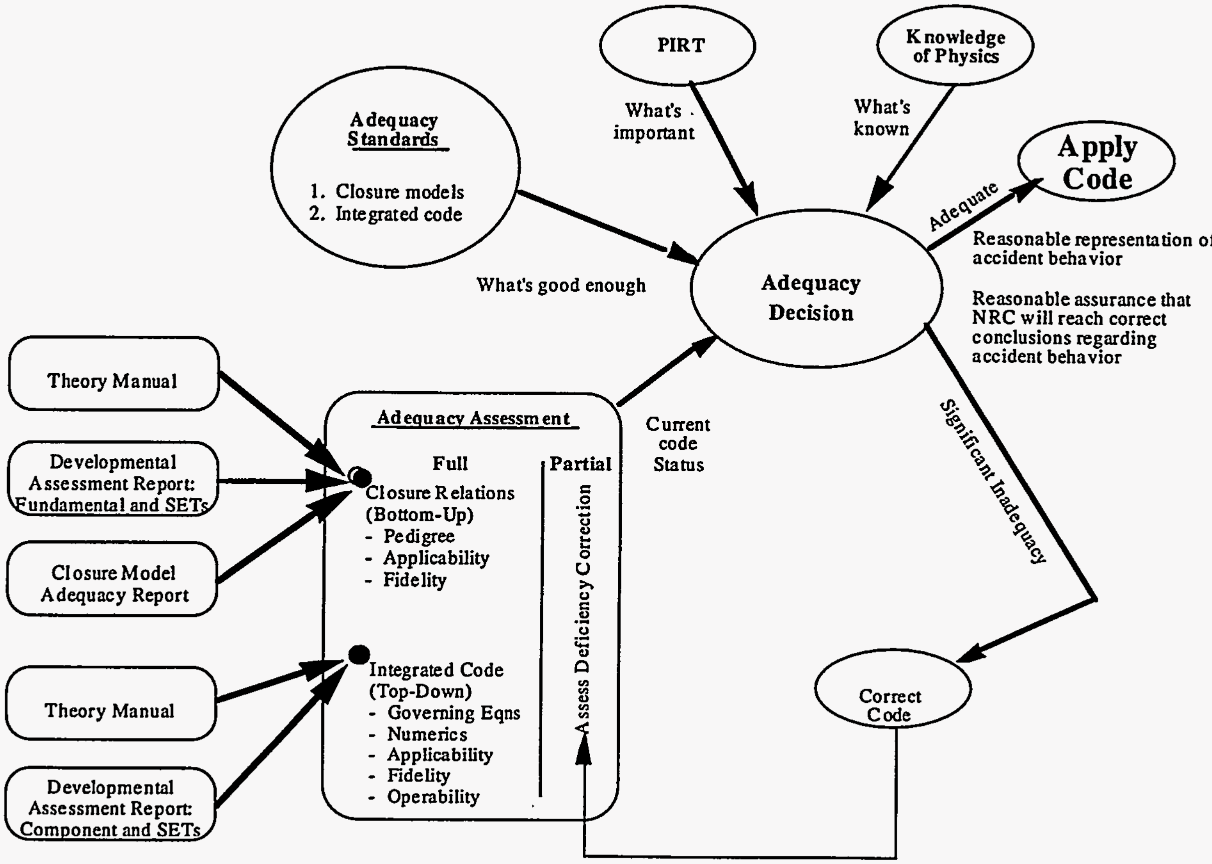


Fig. 1. Adequacy assessment overview.

effect test (SET) data. This part of the assessment effort is called the "bottom-up" review because it focuses on the fundamental building blocks of the code (e.g., closure relationships for interfacial heat and mass transfer). Adjunct features of the pedigree element of the adequacy standard are related to the physical basis of the closure model, assumptions and limitations attributed to the model, and details of the adequacy characterization at the time the model was developed. Adjunct features of the application element are related to whether the model, as implemented in the code, is consistent with the pedigree, or whether use over a broader range of conditions has been demonstrated. Adjunct features of the fidelity element are related to the existence and completeness of validation efforts (comparison to data), benchmarking efforts (comparison to other standards, e.g., a closed-form solution or results of another code), or some combination of the two.

Second, the adequacy of the integrated code is evaluated by examining the field equations, numerics, applicability, fidelity to the component or integral effect test (IET) data, and operability. This part of the assessment effort is called the "top-down" review because it focuses on the integrated code. An adjunct feature of the field equation element of the adequacy standard is that the equations are accepted by the scientific community. Adjunct features of the numeric solution element include convergence, stability, and property conservation. Adjunct features of the application element are related to whether the integrated code is capable of modeling the key plant systems and components. Model noding issues also are addressed as an element of applicability. Adjunct features of the fidelity element are related to the existence and completeness of validation efforts using applicable IET data. Adjunct features of the operability element are related to code robustness and run time (e.g., does the code run successfully to completion for the required scenarios in an acceptable time interval?).

Several key perspectives must be considered during a code adequacy assessment effort. These perspectives provide insights regarding the relationship of the elements of code adequacy assessment to each other and to the whole of the assessment. These perspectives support the process of adequacy assessment of thermal-hydraulic (T/H) analysis codes by addressing the question "How good is good enough"? Three key perspectives relate to (1) knowledge of physical processes, (2) the relative importance of physical processes, and (3) adequacy standards. These concepts can only be covered briefly here, but they are discussed in more detail elsewhere.[5]

The current level of scientific knowledge regarding T/H processes that occur in nuclear power plants during accident sequences varies. The physics of some physical processes are well understood, whereas the physics of other physical processes are partially or poorly understood. The associated perspective is that a computer code cannot be expected to model precisely phenomena that are not yet fully understood by the scientific community.

Some processes and phenomena are more important than others and have a dominant influence on the course of an accident; therefore, it is important that the relative importance of systems, components, processes, and phenomena be assessed. Code models that are necessary to simulate highly ranked phenomena accurately must satisfy the appropriate adequacy standards fully; code models having less impact on the predicted course of the transient are held to a lesser standard. There are several recent examples of phenomena identification and ranking (PIRT) efforts.[6,7]

Finally, The standard for technical adequacy of the individual closure models is that (1) the model pedigree is known, documented, and acceptable; (2) the model is used appropriately (the application of the model is acceptable); and (3) the prediction of the phenomena being modeled is acceptable because the model predicts the appropriate data with acceptable fidelity or accuracy. The standard for technical adequacy of the total code is that (1) the field equations represent the key processes and phenomena, (2) the numeric solution approximates the equation set (field and closure) with acceptable accuracy, (3) the code is used appropriately (the application of the integrated code is acceptable), (4) the prediction of the performance of key systems, components, processes, and phenomena is acceptable because the model predicts the appropriate IET data with acceptable fidelity, and (5) the operability of the code is acceptable. Fidelity of code-calculated results to data is the best measure of "How good is good enough"? Judgments are based on the application of a standardized and consistent set of criteria that has been applied previously in the assessment of NRC-sponsored codes originated from fundamental tests, separate effect tests (SETs), component tests, or IETs that have been developed.[8]

METHODOLOGY

TRAC will calculate one-dimensional (1D) and three-dimensional (3D) (rectilinear and cylindrical coordinates) fluid flow involving gas, liquid, and mixture states. Two fluids are modeled with six equations to capture nonhomogeneous, nonequilibrium behavior. The field equations solved by TRAC are the combined-gas mass, liquid motion, combined-gas motion, total energy, combined-gas energy, noncondensable-gas mass, and liquid solute concentration equations.[1] The associated dependent variables are the liquid and gas velocities, liquid and gas temperatures, void fraction, pressure, noncondensable partial pressure, and solute concentration.

TRAC has a flow-regime-dependent constitutive equation package. Closure relationships are required for the interfacial area, interfacial mass transfer rate, interfacial drag coefficient, liquid wall-drag coefficient, combined-gas wall drag coefficient, liquid interfacial heat-transfer coefficient, combined-gas interfacial heat-transfer coefficient, liquid-to-gas sensible heat-transfer coefficient, wall-to-liquid heat transfer, and wall-to-combined gas heat-transfer coefficient. A separate mass equation is added for a noncondensable gas, and a separate equation is added for tracking solutes in the liquid phase.

A key modeling challenge of general purpose T/H systems analysis codes such as TRAC is illustrated in Fig. 2 for the liquid mass and combined-gas mass equations. Individual constitutive models must be provided at the two-fluid interface for closure of the two-fluid model for these equations. The interface-to-liquid heat-transfer coefficient for each flow regime that might be encountered, e.g., bubbly-slug, churn, annular-mist, stratified, plug, and reflood, must be provided. In a similar manner, constitutive models must be provided for the interface-to gas heat-transfer coefficient and the interfacial area for the same flow regimes. Closure relationships must also be provided at the wall, as shown in Fig. 2.

The code sorts the problem of single-phase vs two-phase fluid at a very high level in determining the equation set to be solved. If the fluid is single-phase liquid or vapor, all of the interfacial processes are eliminated, and the code considers only the interactions with the walls and the transport of a single-phase fluid. For the case of single-phase liquid, the code sets the vapor velocity to that of a bubble; and for the case of single-phase vapor, the code sets the liquid velocity to that of a droplet. The code used this prescription to prevent accelerating the appearing phase from zero velocity when the fluid first becomes two-phase.

TRAC is completely modular by component. The components in a calculation are specified through input data; available components allow the user to model virtually any PWR design or experimental configuration. Thus, TRAC has great versatility in its range of applications. This feature also allows component modules to be improved, modified, or added without disturbing the remainder of the code. TRAC component modules currently include accumulators, breaks and fills, generalized heat structures, pipes, pressurizers, pumps, steam generators, tees, turbines, valves, and vessels with associated internals (downcomer, lower plenum, core, upper plenum, etc.).

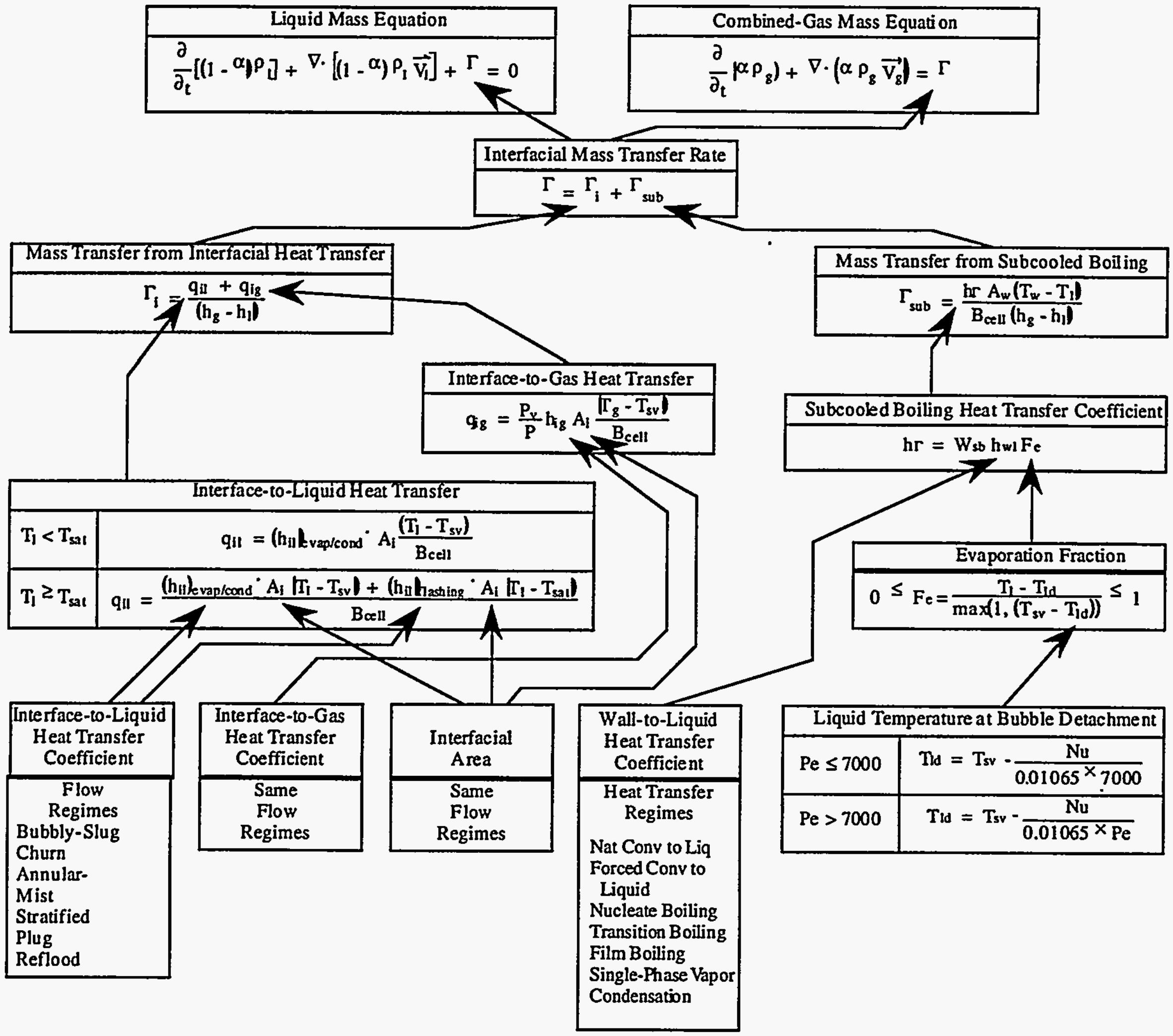


Fig. 2. Example relating flow-regime-dependent constitutive relations to field equations.

TRAC has additional models for nuclear reactor and other energy systems, including point-reactor kinetics with generalized reactivity feedback; general trip, control-system, and component-action models; and a comprehensive heat-transfer capability with 2-dimensional heat conduction and radiation. Each of these models adds to both the generality and the complexity of the overall code.

TRAC also is modular by function; that is, the major aspects of the calculations are performed in separate modules. For example, the basic 1D hydrodynamics solution algorithm, the wall temperature field solution algorithm, heat transfer coefficient selection, and other functions are performed in separate sets of routines that are accessed by all component modules. This modularity allows the code to be upgraded readily as improved correlations and test information become available.

Various semi-implicit finite difference schemes have been used for solving problems in fluid flow. In many problems of interest, however, the stability limit on time-step size (less than the mesh size divided by the material velocity) associated with this class of methods if far smaller than is necessary for reasonable accuracy. In such cases the standard approach for cutting computational costs is to eliminate this material Courant limit with a fully implicit difference method, or in multidimensional problems, employ an alternating direction implicit scheme. The SETs method[1,9] was designed to propagate information needed for stability with minimal implicit coupling between spatial nodes. This method has been implemented in TRAC for both 1D and 3D calculations. Information about pressure wave propagation is provided with a basic step, which is simply a semi-implicit equation set. A stabilizing step is then added to provide the necessary flow of information about the density, energy, and momentum being transported across cell boundaries. The SETS method is especially valuable when applied to the full two-fluid model for two-phase flow. For this model, the stabilizer equations add less than 20% to the computational cost per cell per step of the basic equation set. A fully implicit method multiplies this cost by a factor of six. Adaptations of this method are now used in several other T/H codes.

MODELING CAPABILITIES

Most physical phenomena that are important in large- and small-break loss-of-coolant accident (SBLOCA), and non-LOCA analyses can be treated by TRAC. The phenomena include the following: emergency core coolant (ECC) downcomer penetration and bypass, including the effects of countercurrent flow hot walls; lower-plenum refill with entrainment and phase separation effects; bottom reflood and falling film quench fronts; multidimensional flow patterns in the core, downcomer, and plenum regions; pool formation and countercurrent flow at the upper-core support plate (USCP) region; pool formation in the upper plenum; steam binding; average and hot rod cladding temperature histories; alternate ECC injection systems, including hot-leg and upper-head injection; and direct injection of subcooled ECC water, without the requirement for artificial mixing zones.

Models are provided for critical flow (choking) using the improved critical flow model; metal/water reaction; wall friction losses; natural circulation flows; horizontally stratified flows, including horizontal countercurrent flow driven by void fraction gradients down the pipe; vertical stratification modeling in the vessel component and in the interphase mass transfer (condensation) to better calculate pressurizer refill and the general refilling of any vertically oriented component; increased range in the water properties to permit the code to calculate fluid conditions beyond the critical point (pressures in excess of 22.12 MPa) and closer to the freezing point; noncondensable gas tracking, including the injection of the noncondensable gas from the accumulators and the effects of the noncondensable gas on the interfacial condensation; liquid solute (boron) tracking, which can be coupled to the reactivity feedback calculation; point reactor kinetics with a generalized representation of the reactivity feedback associated with the core average fuel temperature, the core average coolant temperature, the core average void fraction, and the core average boron concentration.

TRAC also has a balance of plant modeling capability; a Plenum component consisting of a single hydraulic cell with an essentially unlimited number of connections to simplify 1D connections; mixed 1D and 3D calculations or fully 1D calculations; fast computational speed for 1D and 3D problems when the transient is reasonably slow, as SBLOCA and some non-LOCA transients; very general trip, control system, and component action (such as feedwater pump flow characteristics) modeling capability; the ability to use trips and controls in the steady-state calculation; user convenience features, including free format input with capability to use comment cards or fields; forward and reverse additive friction factors for the hydraulics, capability to choose to input Darcy K factors for the additive friction, capability to choose to input cell centered elevations instead of the old gravity parameters at cell interfaces, and sophisticated input checking; consistent generation of steady-state conditions for initializing transients so that the same T/H models and numerics are used in both the steady state and the transient; general orientation and magnitude of the VESSEL component for gravitational acceleration vector; and a generalized heat-structure component to allow the user to connect two hydro cells, resulting in increased accuracy for the modeling of steam generators, internal vessel structures, etc.

STATUS

TRAC will run on a Cray supercomputer or on Unix workstations (currently Sun SPARCstation, HP 9000, and IBM RISC 6000). TRAC requires a minimum of 32 MB RAM and 100 MB disk storage for practical applications on a workstation. A source code is provided, and Fortran 77 and ANSI standard C compilers are required for installation.

TRAC is configured with a main driver routine and 575 subroutines. The size of the source code is approximately 104,500 lines of which 70,000 are Fortran statements, 30,000 are comment lines; and 4,500 are pre-compile directives such as "include" statements and coding for platform dependencies.

At present, TRAC's grind effort is 10,000–20,000 floating point operations per fluid cell per cycle. This number includes the conduction solution for the heat structures. The range in the grind effort is associated with several factors, including the complexity of the closure models being exercised in a given calculation, the number of 3D nodes in the particular model, etc.

A graphical user interface (GUI) for TRAC has been developed at Knolls Atomic Power Laboratory (KAPL).[10] This X Window base GUI, named TOOKUIL, supports the design and analysis process, acting as a preprocessor, runtime editor, help system, and postprocessor to TRAC. The preprocessor is an icon-based interface that allows the user to create a TRAC model. When the model is complete, the run time editor provides the capability to execute and monitor TRAC runs on the workstation or supercomputer. After runs are made, the output processor allows the user to extract and format data from the TRAC graphics file. Users may become functional in creating, running, and interpreting results from TRAC without having to know Unix commands and the detailed format of any of the data files. This reduces model development, debug time, and increases quality control.

At stages in its development, the various TRAC releases have been assessed against a broad spectrum of fundamental, separate effect, integral effect test, and plant data. It is not possible to provide a complete list of the assessments in this paper; however, a sampling of the facilities for which TRAC assessments have been performed is provided without citation in Table I. These assessments are not repeated for each code version. In fact, we have conducted relatively few assessment efforts with recent code versions. Therefore, we offer the cautionary note that the previous assessment history will not fully apply to the present code versions. We do acknowledge that too few fundamental assessments have been performed throughout the TRAC development effort. Early in TRAC's development history, this was primarily due to the lack of the needed fundamental data. In later years (1989-1995), the code was placed in a maintenance mode and little fundamental closure model development or assessment was pursued. Even today, however, we are concerned that there are serious deficiencies in the fundamental data base, especially data related modeling processes at the liquid-vapor interface in our two-fluid models. We note that a significant development and assessment effort for the TRAC constitutive package as it applies to the AP600 LBLOCA is currently underway. This development and assessment effort is discussed in a subsequent section of this paper.

LIMITATIONS

Although TRAC has these many capabilities and features, it also has limitations. In fact, it is the very limitations in TRAC and other T/H codes of its generation that are the focus of present OECD/CSNI Workshop on Transient Thermal-Hydraulic and Neutronic Code Requirements. As described in the background and purpose statement of the workshop notice: "The T/H codes that are currently being used were developed to study LOCAs in the 1970's. Over time, improvements have been made to the codes in a somewhat ad hoc basis to include new capabilities and to analyze technical issues that some of these codes were not specifically designed to handle. Although these codes are being used to assess reactor safety issues and we are confident of the results obtained using them, these codes no longer provide the best estimate to T/H phenomena." The background and purpose statement continues: "In addition, the computer technology is changing at an every accelerating rate and it is necessary to almost continually modify the cods in order to keep up with the advances. Past efforts to convert the existing codes to new computer environments did not make the codes more robust or reliable because of outdated coding and numerical methods inherent in the fundamental structure of some codes. Also, PRA

TABLE I

LIST OF FACILITIES/PLANTS/DATA THAT HAVE BEEN USED FOR TRAC ASSESSMENT

Fundamental	**Separate Effect**	**Integral**
Analytical Solution for Steady-State Conduction	19-Tube Once-Through Steam Generator Test	Davis-Besse Loss-of-Feedwater Event
Analytical Solution for U-Tube	ATLE tests	Ginna Steam Generator Tube Rupture event
Analytical Solutions for Stratified Flow	B&W Annular Flow Distribution (AFD) Experiments	Jose Cabrera Plant Inadvertant Pressurizer Spray Event
Bennett Tube Experiments	B&W Mark 22 Assembly Facility	Loop Blowdown Investigation Test Facility (LOBI)
Berkeley Reflood Test	CISE Pressurizer Flooding Facility	Loss-of-Fluid Test Facility (LOFT)
Condensation Test Facility	Cylindrical Core Test Facility (CCTF)	Multi-Loop Integral System Test (MIST)
CREARE Counter-Current Flow Experiments	Edwards Blowdown Experiment	Primarkreislaufe facility (PKL)
Dartmouth College Air-Water Counter-Current Flow Tests	FLECHT Forced-Flooding Experiments	Rig of Safety Assessment (ROSA)
Direct Contact Condensation Experiments	Marviken Tests	Ringhals 2 Inadvertant Steam Line Isolation Valve Closure Event
Northwestern University Perforated Counter-Current Flow Limitation Tests	NEPTUNUS Pressurizer Test Facility	Ringhals 4 Loss of Grid Event
Safety Valve ATWS Separate Effect Experiment	Savannah River Laboratory A-Tank Single-Assembly Flow Tests	Savannah River Plant 1985 L-Area Process Flow Test Series
Whatley Bladder Valve Experiments	Savannah River SPRIHTE and FA Rig Experiments	Savannah River Plant L-Area DC Tests
	Slab Core Test Facility (SCTF)	Semiscale Facility
	Upper Plenum Test Facility (UPTF)	Vandellos II Plant Load Rejection Transient

requirements and the need to analyze beyond design basis accident (DBA) events impose new requirements on the codes compared with those that were used previously for DBA and would require code validation in new regimes and much faster codes."

We acknowledge that each of these statements applies, in some measure, to the present TRAC code. Development of the TRAC code series began in the 1970s. The architecture of the code was designed to efficiently utilize the best computational platforms of the time, but that same architecture is the root cause of some of the present deficiencies of the code. Among the most important is the use of a container array and "pointers." The container array was important in the original construct of the code because it facilitated the general modeling capability of the code, e.g., a small experimental facility for one application and a current-generation nuclear power plant for the next application. This innovative structure, so important for the early computational platforms with small-capacity, high-speed central processing units, now acts as a barrier to efficient computation on current computational platforms.

At present, TRAC executes at approximately 6 million floating point operations per second (MFLOPs) on a Cray Y-MP. Typical rates are 70 MFLOPS for other complex scientific application codes. We have found that the container array approach obstructs compiler optimization and is one factor in the code running slower than needed for some applications, e.g., PRA analyses and simulators. Maintainability is also affected as the container array and pointers make the code difficult to learn and understand. Plans for addressing the container array deficiency are well advanced, as discussed in the next section.

A second important limitation is associated with the evolution of programming languages, the long-term development of the code, and the involvement of ~20 developers over the years. The present code utilizes Fortran 77, which has led to overly complex protocols due to Fortran 77 limitations. Some of the coding is old and illogical, and there are multiple maintenance points. Extensive effort is required to implement changes. Plans for addressing issues related to the programming language, old and illogical coding, and multiple maintenance points are well advanced as discussed in the next section.

As shown in Fig. 1, a full adequacy assessment consists of conducting reviews of both the code closure relations (bottom-up review) and the integrated code (top-down review). The code limitations previously discussed in this section, namely issues related to the container array, pointers, programming language, and old code, are not explicitly shown in Fig. 1. However, the code architecture and programming language form the fundamental code structure in which the T/H models are implemented. To the extent that these issues relate to the adequacy assessment envisioned in Fig. 1, they do so in the area of operability. We have previously mentioned the impact of the container array on run time. Similarly, the use of nonstandard programming practices to compensate for the limitations of Fortran 77 also results in computational overhead that increases run time.

There are other limitations that directly affect code adequacy, as shown in Fig. 1. TRAC is currently being used to support the NRC's large-break (LB) LOCA certification review for the AP600 reactor. Although the initial peak cladding temperature responses predicted by WCOBRA/TRAC and TRAC were similar, submittals by the vendor based upon more recent WCOBRA/TRAC calculations are markedly different.[5] This has called the adequacy of the TRAC blowdown rewet and reflood models into question. As discussed in the next section, a development activity is presently under way to address this issue.

With the rapid advancement of computer platforms, the analyst-machine interface is rapidly becoming an important limiting factor. The TOOKUIL GUI[9] previously discussed is one element of the TRAC-related effort to address this limitation in the important areas of model creation, run time management, and output extraction and formatting. However, we are still limited in our ability to process the voluminous data generated by TRAC. As discussed in the next section, a development activity is presently under way to address this issue.

DEVELOPMENT ACTIVITIES AND PLANS

The limitations discussed in the previous section have resulted in the NRC sponsoring several development activities. In addition, plans are well advanced for a significant TRAC modernization effort jointly sponsored by the NRC and the U. S. Department of Energy (DOE). Finally, at the request of the NRC, plans have been developed for consolidation of TRAC-P, TRAC-B, and the multidimensional kinetics modeling capabilities of the RAMONA code. Initiation of the consolidation effort is currently delayed but it will be reported for completeness.

We first report on a recently completed activity. We have just completed a developmental assessment plan for TRAC focused on the AP600 LBLOCA application.[11] This effort has defined a developmental assessment plan for TRAC to support its application to the AP600 LBLOCA transient. As a part of this effort, we reviewed the AP600 and its safety systems, testing done in support of the design certification, and a calculation of an AP600 LBLOCA transient. We used the AP600 LBLOCA PIRT,[7] which rates the importance of processes and phenomena to the LBLOCA transient. We identified the code models corresponding to the processes and phenomena in the PIRT and combined the PIRT priorities with the adequacy of the code models to generate the developmental assessment priorities. Based on these assessment priorities and on the fluid conditions existing during the various phases of the transient, we identified separate-effects tests that can be used for developmental assessment. The nature of the PIRT leads to a concentration on separate-effects tests, and these tests seldom lead to comprehensive testing of the overall code performance. Therefore, we also identified integral tests for inclusion in the developmental assessment plan to check the overall quality of the code and to support enhancements to the robust nature of the code (the ability of the code to perform calculations without code failures). The resultant developmental assessment matrix is summarized in Table II.

We next report on three development activities currently in progress. The first is an adequacy assessment of TRAC closure and special models.[5] In effect, we are nearing completion of the bottom-up review described in Fig. 1. As previously described, the pedigree (physical basis, assumptions and limitations, and original adequacy characterization), applicability (consistency with pedigree or other demonstrations of applicability), and fidelity (validation or comparison to data) and benchmarking (comparisons to other correlations) are evaluated. An example of the detailed information tabulated for each closure model is provided in Fig. 3 for the bubbly flow interfacial area model in TRAC. Summary findings are tabulated for each closure model, e.g., interfacial area. Information from the adequacy assessment effort, when coupled with the conclusions of the AP600 LBLOCA PIRT,[7] laid the foundation for decisions regarding the needed model development and developmental assessment.

The adequacy standards for pedigree are that the model pedigree is known, documented, and acceptable. The adequacy standard for applicability is that the model application is acceptable. The adequacy standard for fidelity is that the model predicts the appropriate data with acceptable accuracy. The term "acceptable" is invoked repeatedly and this implies judgment based upon documented information. The most concrete measure is fidelity. For fidelity assessments we use standardized fidelity criteria that characterize

TABLE II

TRAC DEVELOPMENTAL ASSESSMENT MATRIX FOR THE AP600 LBLOCA

LBLOCA Phase	Separate-Effects Tests	Integral Tests
Blowdown	GE Level Swell 1004-3 5801-15 INEL post-CHF data T-Junction Test Facility THETIS Boildown test T2L095 Winfrith steady-state post-CHF data	LOFT LP-02-6 LOFT L2-3 Semiscale S-06-3
Refill	GE Level Swell 1004-3 5801-15 INEL post-CHF data THETIS Boildown test T2L095 UPTF 5B, 6, 21A, 21B, 25B, 21D, and 27A Winfrith steady-state post-CHF data	LOFT LP-02-6 LOFT L2-3 Semiscale S-06-3
Reflood	CCTF Runs 14, 54, ... FLECHT-SEASET 31504 31701 33436 INEL post-CHF data Lehigh SCTF (total of 8 tests between CCTF and SCTF) Winfrith steady-state post-CHF data	LOFT LP-02-6 LOFT L2-3 Semiscale S-06-3

the agreement as excellent, reasonable, minimal, or insufficient. Reasonable agreement is the minimum standard for adequacy.

The second activity is the integrated TRAC development of LBLOCA closure packages for AP600 applications. Some background is provided for this effort. In the late 1980s, the NRC undertook to have its contractors improve the documented basis for the T/H systems analysis codes. The TRAC-PF1/MOD1 (MOD1) Correlations and Models Document[12] was prepared to provide detailed descriptions of the various constitutive

ADEQUACY ASSESSMENT — BUBBLY SLUG FLOW INTERFACIAL AREA	
Pedigree	Physical basis: Ishii and Mishima (Ref. 5.88) assumed an idealized flow pattern in bubbly slug flow and developed an equation for the interfacial area concentration based on the geometrically idealized shapes. Over repeated lengths L, two distinct regions are assumed to exist within L. 1) a region occupied by a liquid-bubble mixture. 2) a region occupied by a vapor slug and surrounding liquid. The slugs convert to cap bubbles if the channel diameter exceeds a critical diameter. From geometrical arguments, the area of cap bubbles is greater than slugs. Assumptions and limitations: Interfacial area models are mostly based on steady-state and fully developed flow data. In addition, almost all data are obtained from adiabatic air-water experiments, at or near atmospheric pressure. 1) The bubble portion of bubbly slug flow can be represented as a population of spherical bubbles that are characterized by the Sauter mean diameter, D_b. D_b is evaluated using a simple expression by Ishii (Ref. 5.31). Bubble size and shape probability distributions are not considered. The upper and lower limits for bubble diameter are given by $0.1 \text{ mm} \leq D_b \leq 0.9 D_H$. 2) Slugs or cap bubbles form depending on the diameter of the flow channel. Slugs form when the channel diameter is less than a critical diameter. Slugs form if $a > 0.3$ and the mass flux is <2700 kg/m^2-s (see flow map review). Cap bubbles form if the $D > 50L_o$ where L_o is the Laplace coefficient (MOD2/Eq. 4-13) . The idealized cap bubble of Ishii and Mishima (Ref. 5.88) assumes a wake angle $\approx 55^o$. Original adequacy characterization: 1) Ishii's expression for the Sauter mean diameter, D_b, was stated to be an approximate arithmetic average of minimum and maximum bubble diameters observed experimentally. 2) Kataoka and Ishii (Ref. 5.12) state that slug bubbles cannot be sustained for channels with a diameter much larger than $40L_o$. The TRAC specification that cap bubbles form for $D > 50L_o$ is similar. Ishii and Mishima (Ref. 5.88) state that the observed wake angles range from 46 to 55°. Specification for slug to cap bubble transition is consistent with the data of Grace et al. (Ref. 5.13).
Applicability	Consistent with pedigree: Yes, except when quasi-steady and local equilibrium assumptions are violated. Otherwise demonstrated: A mathematical treatment of the interfacial area for bubbly flow (spherical bubbles, no vapor slugs) is shown to be equivalent to the code's model for like conditions.
Fidelity	Validation: 1) Model assessment studies were conducted using the data of Shilimkan and Stepanek (Ref. 5.14), Kasturi and Stepanek (Ref. 5.15), and DeJesus and Kawaji (Ref. 5.16). Each experiment was for upflow in a long vertical tube. Tube internal diameters varied from 0.6 to 2.54-cm i.d. With respect to the data of DeJesus and Kawaji, TRAC-PF1/MOD2 overpredicts the interfacial area concentration in the bubbly slug regime (MOD2/Sec. 4.1.11, Fig. 4-30). After back-calculating the Sauter mean diameter from the data, it was concluded that the available interfacial area data are not directly applicable for reactor safety analysis because the experimental setup does not allow a breakup mechanism into dispersed bubbles at the measured flow rates. The comparison with the data of Kasturi and Stepanek and Shilimkan and Stepanek is reported to have exhibited similar patterns during assessment. 2) None explicitly cited in MOD2 Theory Manual. Benchmarking: None explicitly cited in MOD2 Theory Manual.

Fig. 3. Example of TRAC closure model adequacy assessment detailed information.

models used for closure of the field equations. This documentation was reviewed by several groups, specifically the Advisory Committee on Reactor Safeguards (ACRS) and the Technical Program Group (TPG) engaged in developing the Code Scaling, Applicability, and Uncertainty (CSAU)[13] evaluation methodology. The following summarizes the key issues at that time.

> The major criticism of TRAC based on [ACRS] review of the Q/A [TRAC-PF1/MOD1 Correlations and Models Document] document, was that many of its basic physical models were not based on a sufficient set of <u>basic</u> data. Since TRAC appears to achieve a reasonable representation of experimental data, LANL must have accomplished this by 'tuning' these basic models to integral system test data, rather than using basic data to obtain the necessary two-fluid constitutive relations. Therefore, use of the code beyond its integral; system data base could lead to large uncertainties in the results.[14]

Although not stated in the above,[14] a related concern of the reviewers was that a significant number of the closure relationships in MOD1 were of an *ad hoc* nature. These concerns were verified by Los Alamos which, for example, reported " . . . the vertical flow map was basically invented to fulfill a need, no original reference exists for this map. . . . This map was originally based on physical intuition . . ."[12]

ACRS, TPG, and NRC criticisms played a significant role in the development of the next major code version, MOD2. The TRAC code-development team adopted an approach that will henceforth be identified as the "absolute" pedigree approach; this approach required that only closure models with an acceptable pedigree could be entered into the code. Further, these models could only be incorporated in their pedigreed or unmodified form.* Where it was deemed necessary, models affecting the blowdown and refill phases of a LBLOCA transient were implemented in MOD2, which satisfied this constraint.

---

* It must be remembered that even the “absolute pedigree” approach must be adapted somewhat due to discontinuities that will sometimes exist from correlation to correlation. The technique more frequently used in this case is interpolation from one correlation to the other over some region.

MOD2 also included a new reflood model that used a modification of the "absolute" pedigree approach.[15-17] The logic behind the generation of this modified approach is summarized as follows:

> Whenever possible, correlations known to apply to a given regime for a particular closure quantity were used. Frequently, however, the original correlation could not be applied directly but had to be modified. For those cases, we tried to use the 'kernel' or 'functional' dependence of the original correlation and modify only its magnitude by use of a multiplier. When no correlations were available for given regimes, we tried to define known bounding regimes and use a weighting function between the known regimes to represent the unknown quantities.[15]

This approach will henceforth be identified as the "conditional" pedigree approach. The approach was built primarily upon the use of basic data from simple tube experiments. We know that limits exist on the current MOD2 reflood model, i.e., the coefficients were modified based primarily on only single-tube data. Extension of the work to fuel rod bundles was terminated when MOD2 was placed into a maintenance mode.

The NRC is currently sponsoring work at Los Alamos to provide a code of demonstrated adequacy for AP600 LBLOCA confirmatory analyses. Of the various approaches possible for closure package development, we wish to discuss only the two approaches previously identified: the absolute pedigree and conditional pedigree approaches. Selection of these two approaches for further examination arises from a consideration of the interconnected inputs related to modeling concepts, constitutive equations sets, and data sets (Fig. 4).

For the absolute pedigree approach, constitutive equations are selected on the basis of pedigree, applicability, and fidelity to basic data. The pedigreed constitutive equations are introduced into the code in their absolute (pure) form; no modifications are permitted. A basic premise underlies the absolute pedigree approach, namely, that the selected constitutive equations contain all the necessary phenomenological information for the modeled phenomena. This includes the various phenomenological couplings that may not have been measured in the experiments that produced the data sets used for creating

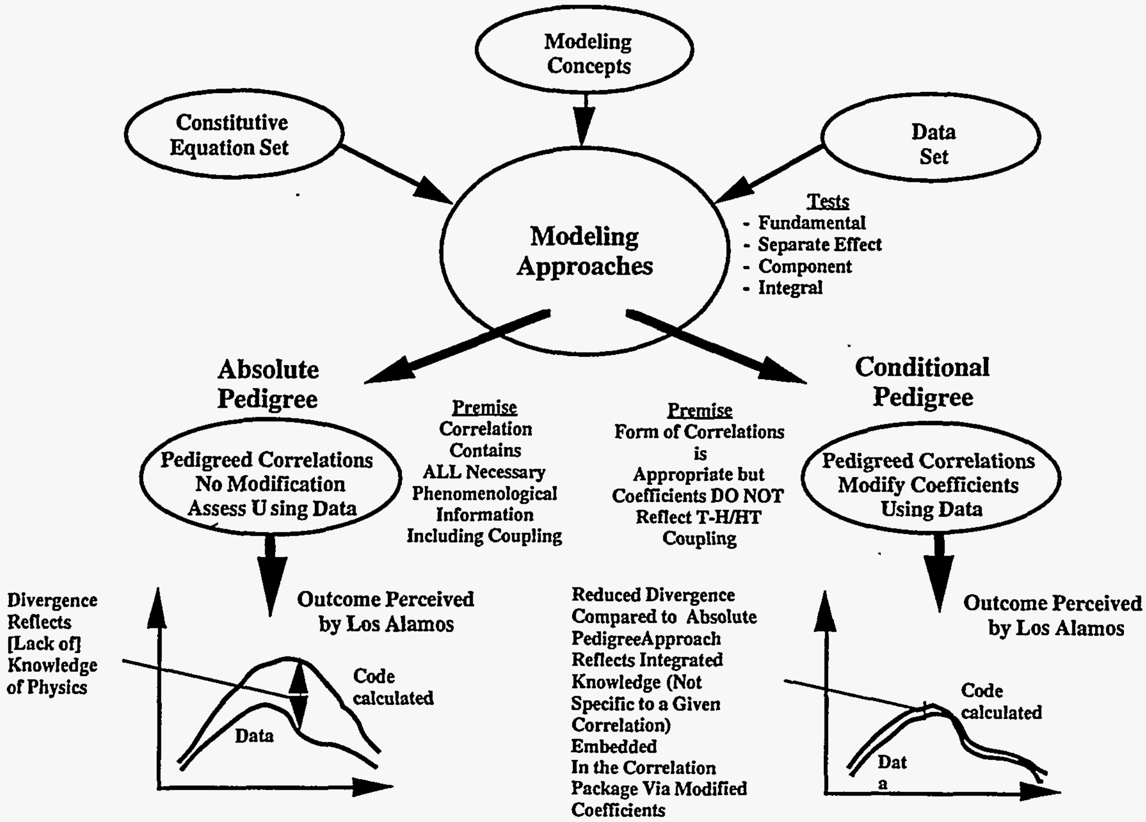


Fig. 4. Approaches to integrated closure package and accident phase modeling.

the constitutive equation. For example, it is assumed that within the absolute pedigree that the wall and interfacial heat transfer and wall and interfacial drag have been properly coupled in the development of the constitutive equation. When this assumption is valid, and given correct implementation of each constitutive equation in the code, a positive outcome of this approach is that divergence between the code-calculated results and data provides a direct indication of the degree to which the physics are understood and captured in the constitutive relationships.

Los Alamos has concluded that the absolute pedigree approach (Fig. 4) will have an undesirable outcome, namely that the difference between the code-calculated results and data for key parameters will be unacceptably large. As TRAC-PF1/MOD2 evolved from TRAC-PF1/MOD1, the absolute pedigree approach was followed. In the process, extensive information embedded in TRAC-PF1/MOD1 constitutive packages was lost as the code was broadly assessed against multiple integral test programs over many years.

The results obtained during a Los Alamos conducted investigation of various state-of-the-art models lead us to the conclusion that the conditional pedigree approach is the correct approach. For the conditional pedigree approach, constitutive equations are selected in the same manner as for the absolute pedigree approach. It is likely, in fact, that the same constitutive relationships would be used for either approach. The pedigreed constitutive equations are introduced into the code, but with a single, important difference. Modifications are permitted to a single part of the constitutive relationships, namely the coefficients.* These are adjusted so that reasonable code-data comparisons are obtained for a selected set of basic data and data from scaled integral experiments (Fig. 4). Thus, additional phenomenological information becomes embedded in the constitutive set as the relationship coefficients are adjusted to improve the code-data comparisons. A basic premise underlies the conditional pedigree approach, namely, that the form or kernel of the constitutive equation is appropriate but that all necessary phenomenological couplings have not been included in the constitutive equations for the previously stated reasons. This development activity, as presently planned, will provide results for both the absolute and conditional pedigree approaches.

The integrated development of closure packages will utilize nonlinear optimization techniques to "recorrelate" the model coefficients. Nonlinear optimization techniques are well established and have been used as part of complex system design for a number of years. The effort with integrated closure models will use the computer itself to accomplish the coupling and recorrelation within the closure packages.

Visualization and plotting tool for TRAC, X-TRAC-View (XTV), uses the platform-independent X Windowing System to create its GUI. XTV was originally designed to aid in visualizing complex phenomena that result from LOCAs or other similar incidents, where line plots of critical variables do not easily indicate all of the interactions within a component and between components. XTV has been expanded to include line plot capabilities, and is scheduled to eventually replace EXCON and TRAP, the current TRAC plotting features.

XTV allows the user to view up to 18 2D representations of components simultaneously. These visuals can be either static at a given time interval, or animated throughout time. Three dimensional components can be viewed in either Cartsian or cylindrical

* This might be thought of as a recorrelation of model within the frame work of the code.

coordinates with any one of the three axes fixed at a particular value. For the multidimensional Vessel component, the additional capability exists to optionally plot either liquid or vapor flow vectors, as well as wall temperatures, in addition to any other scalar value at each cell. By placing adjoining components in neighboring viewpanes, one can visualize how the two components interact with respect to a certain variable; conversely, two different variables can be examined for the same component.

XTV is currently being expanded to improve its online plot capabilities, as well as being able to perform calculations on any of the variable arrays. Additionally, capabilities to visualize any and all of the data generated in TRAC are being added. Its inherently modular data structure allows calculated values to be added as if they were produced in TRAC, which should also help XTV to function as an interactive controller for TRAC, allowing visualization as the results become available, a feature planned for implementation in late 1996.

We next report on the TRAC modernization effort. Both the NRC and DOE are sponsoring elements of this activity. Our overall objective is to provide a computationally efficient, portable, standard code in Fortran 90. We also seek significant improvements in extensibility by providing data structures required for new methods and models and maintainability. The specific goals of the TRAC modernization effort are as follows:

- Apply modern software engineering principles,
- Achieve full portability to all single-processor Unix-basd platforms,
- Significantly improve the maintainability of the code,
- Achieve a factor of 10 improvement in run time on current single-processor platforms,
- Improve code operability and robustness
- Position the code for parallelization,
- Separate the input/output and computational engine, and
- Provide full functionality at all times during the modernization effort.

The modernization plan consists of three stages. The first stage is to reimplement the current data structures in Fortran 90 for portability without impacting the computational routines. We will also enhance information hiding between different data structures. We will take advantage of the current modular code design and object-oriented data structures and transform the code rather than begin anew. Throughout the reimplementation effort, we will maintain an operational code relative to an appropriate test matrix. At the completion of the first modernization step, the following success metrics will apply. The run-time improvement will be quantified, special coding associated with multiple computational platforms will be eliminated with a concomitant improvement in portability, the container array will be eliminated, and the modularity of the code will be increased. A brief synopsis of each of the tasks within the first stage effort is provided in Table III. We anticipate initiation of this effort about August 1, 1996.

The second stage of the modernization plan is to develop new data structures to support improved computational efficiency, maintainability, and extensibility without impacting the computational routines. The third stage of the modernization plan reorganizes the computational flow and reimplements the computational routines to take advantage of the new data and new features of Fortran 90, such as array syntax. At the present time, there is no commitment from either the NRC or DOE to continue with the second and third stages of the modernization plan.

Finally, we report on the TRAC consolidation effort. As previously discussed, we are not presently pursuing this task at the direction of the NRC. However, we have been informed that the NRC may pursue this effort at a later time; therefore, a brief summary of the effort is provided here.

The NRC has developed several system transient codes, each for a slightly different mission. TRAC-P was developed at Los Alamos to analyze LBLOCAs and system transients in PWRs. A version of this code was used to develop TRAC-B, for analysis of LOCAs and system transients in BWRs. The RAMONA code, with 3D neutronics capability for BWRs, was purchased by the NRC from Scandpower and modified by adding capabilities to perform calculations for BWR stability and anticipated transient without scram. TRAC-P is being maintained at Los Alamos, TRAC-B at Pennsylvania State University, and RAMONA at Brookhaven National Laboratory. Maintenance of these three codes cost the NRC a considerable amount of funding each year. Consolidation of TRAC-P and TRAC-B, including the capabilities of RAMONA, will be cost beneficial.

TABLE III
FIRST STEP TRAC MODERNIZATION ACTIVITIES

ID	*Subtask Title* and Task Description
20	*Dynamic system arrays*: Use Fortran 90 built-in dynamic memory management facility to dynamically allocate system-level arrays.
21	*Convert comdecks:* Convert existing common blocks, to F90 MODULES as appropriate in order to support dynamic memory allocation and ease code modification.
22	*Test object concepts:* Use the Heat Structure (HS) component to design the concept for implementing TRAC's component data structure in Fortran 90.
23	*Vessel data structure:* Modify the current inverted/equivalenced Vessel array data structure.
24	*Upgrade FIND:* Provide a universal, flexible, abstract, and efficient interface among TRAC's various data structures.
25	*Replace 1D hydro data base:* Replace 1D hydro component and boundary condition data bases and interfaces to driver routines.
26	*Replace plenum data base:* Replace the zero-dimension hydro Plenum component data base and interfaces in a manner that minimizes changes to lower-level core routines.
27	*Complete HS data structure:* Complete the modernization of the HS data structure and its interfaces with other modernized components to achieve full functionality.
28	*Control system/1D hydro communication:* Remove hard-wired knowledge of 1D component data structure from control system.
29	*Control system/HS communication:* Remove hard-wired knowledge of heat slab (HS component data structure from control system.
30	*Object-oriented Control system:* Replace current control system coding that is dispersed throughout all the various components with modular coding. Re-implement control-system data structure in a standard and portable fashion.
31	*PIPROD with FIND capability:* Replace current hard-wired communication between 1-D hydro components and Heat Structures in PIPROD with use of FIND.
32	*Closure relations data structure:* Develop and implement any changes to closure relationships required to accommodate the new data structures.
33	*Neutronics and power:* Re-implement reactor power and neutronics capabilities with the new F90 data structures.
34	*Remaining non-std/port constructs:* Automate the detection of any remaining non-standard and/or non-portable constructs in TRAC.
35	*Steady-state initialization:* Implement the new hydraulic path steady-state (HPSS) initialization capabilities to be consistent with the new F90 data structures.
36	*Constrained steady state:* Implement constrained steady state capabilities to be consistent with the new F90 data structures.
38	*Radiation heat transfer:* Replace data base and interface (driver routines) associated with radiation heat transfer.
39	*Generalize output:* Provide generalized output interface using F90. Implement an array management methodology such that information that characterizes each array is embedded in the code using standard Fortran 90 features.
40	*English units:* Standardize units processing within the code.
41	*Integrated testing 1D code*
43	*Special models* : Replace data base and interface (driver routines) associated with special models, e.g. TURB.
44	*Vessel full capabilities:* Complete modernization of Vessel coding.
46	*Integrated testing 3D code*

We have proposed the following major tasks for combining the codes.

1. Modify software development procedures to meet the intent of American National Standards Institute, Inc., standards and NRC requirements using a cost-effective graded approach.

2. Write a Software Requirements Description (SRD) document that forms the overall basis for the consolidated code and associated GUIs.

3. Identify models that need improving, or identify where new models must be developed, considering various PIRTs for BWRs and PWRs and code adequacy requirements. Recommend a developmental assessment matrix based on the PIRTs and code adequacy requirements.

4. Examine current models and recommend those that should be included in the consolidated code. Select specific existing models from both TRAC-P and TRAC-B for the consolidated code. The intent of this phase of consolidation is to retain TRAC-P models that are judged adequate for both PWR and BWR applications and to add TRAC-B models as necessary for BWR applications. Modify the SRD to include discussions of specific models.

5. Modernize the software architecture of TRAC-P for improved portability, maintainability, and extensibility. Verify the modified architecture, including data structures and interfaces, with no changes in functional TRAC-P models. Note: this effort is precisely the stage 1 modernization effort previously discussed.

6. Modify the modernized TRAC-P to include selected existing BWR models, correlations, and functions from TRAC-B. This forms the consolidated code TRAC. Verify the operation of the individual BWR models in TRAC. Some of this work can be done in parallel with modernization of TRAC-P, e.g., models not affecting data structures.

7. Examine the available 3D neutronics models suitable for TRAC, and select the best model and implementation details. Integrate these into TRAC, combining the best features from each. Modify to capture RAMONA features. Verify.

8. Extend GUIs—TOOKUIL from KAPL and XTV from Los Alamos—for BWR models in TRAC. Integrate XTV with TOOKUIL. Add an input deck converter for TRAC-B to TOOKUIL.

9. Combine the standard verification and validation test matrices for both TRAC-B and TRAC-P into a single master test matrix for TRAC. Modify the matrix as necessary to meet testing requirements that are based on PIRT and code adequacy.

10. Verify TRAC, after full integration of all BWR models, against the master test matrix and resolve differences in test results between TRAC and either TRAC-B or TRAC-P.

11. Integrate the code documentation for TRAC-B and TRAC-P into a master set for TRAC, contemporaneously with programming, in electronic and paper editions.

CONCLUDING REMARKS

The TRAC code is presently applicable to many facilities and transients. Through its many versions, it has been broadly assessed against a broad set of separate effect and integral effect data. Its closure models have a documented pedigree. However, the applicability of numerous closure models is more limited. This is because the data from which the closure models were developed frequently cover only a fraction of the conditions encountered during calculated accident scenarios in nuclear power plants. Some of the strengths of TRAC are its generalized modeling capabilities, multidimensional Vessel component, point and multidimensional kinetics models, and the two-fluid model. TRAC approaches fulfillment of its design objectives in that it accurately models most important light water reactor accident phenomena in current-generation and advanced-passive reactors, delivers best-estimate predictions of accident progression, and has proven adaptable to some other reactor types.

With the passage of time and the advancement in computational platforms and languages, the deficiencies in TRAC are becoming more serious. These limitations are most directly associated with the data structure and code architecture. The TRAC data structure and architecture date from the 1970s. Although they were advanced for their time, they now stand as liabilities when measured against current data structures and architectures.

These limitations most adversely impact run time, portability, maintainability, and extensibility. Fortunately, the start of TRAC modernization efforts is imminent. We believe that these activities, when completed, will result in improved run time, portability, maintainability, and extensibility. TRAC will then have an improved capability for its targeted applications, namely, reactor safety analyses for both operating and planned reactors, audits of licensee's calculations, analyses of operating reactor events, analyses of accident management strategies, support for test planning and interpretation, support for PRAs, design analyses, and nuclear plant training simulators.

REFERENCES

1. J. W. Spore, S. J. Jolly-Woodruff, T. K. Knight, J-C. Lin, R. A. Nelson, K. O. Pasamehmetoglu, R. G. Steinke, and C. Unal, *TRAC-PF1/MOD2 Code Manual, Volume 1, Theory Manual,* NUREG/CR-5673, LA-12031-M, draft document (July 21, 1993).

2. N. M. Schnurr, R. G. Steinke, V. Martinez, and J. W. Spore, *TRAC-PF1/MOD2 Code Manual, Volume 2, User's Guide,* Los Alamos National Laboratory document NUREG/CR-5673, LA-12031-M (July 1992).

3. L. A. Guffee, R. G. Steinke, and J. W. Spore, *TRAC-PF1/MOD2 Code Manual, Volume 3, Programmer's Guide*, NUREG/CR-5673, LA-12031-M, draft document (November 1990).

4. J. C. Lin, V. Martinez, and J. W. Spore, *TRAC-PF1/MOD2 Code Manual, Volume 4 Developmental Assessment Manual,* NUREG/CR-5673, LA-12031-M, draft document (August 1993).

5. B. E. Boyack, "TRAC-PF1/MOD2 Adequacy Assessment: Closure and Special Models," Los Alamos National Laboratory Draft document (December 5, 1995).

6. C. D. Fletcher, G. E. Wilson, C. B. Davis, and T. J. Boucher, "Interim Phenomena Identification and Ranking Tables for Westinghouse AP600 Small Break Loss-of-Coolant Accident, Main Steam Line Break, and Steam Generator Tube Rupture Scenarios," Idaho National Engineering Laboratory document INEL-94/0061, Rev. 1 (July 1995).

7. B. E. Boyack, "AP600 Large-Break Loss-of-Coolant Accident Phenomena Identification and Ranking Tabulation," Los Alamos National Laboratory document LA-UR-95-2718 (1995).

8. P. S. Damerell and J. W. Simons, Eds., "2D/3D Program Work Summary Report," prepared jointly by Japan Atomic Energy Research Institute, Gesellschaft füer Reaktorsicherheit, Siemens AG, UB KWU, US Nuclear Commission, Los Alamos National Laboratory, and MPR Associates, Inc., US Nuclear Regulatory Commission document NUREG/IA-0126, GRS document GRS-100, MPR Associates, Inc., document MPR-1345 (June 1993).

9. J. H. Mahaffy, "A Stability-Enhancing Two-Step Method for Fluid Flow Calculations," *Journal of Computational Physics* **46**, 329–341 (1982).

10. A. E. Kelly, C. K. Harkins, and R. J. Smith, "Development of a Graphical User Interface for the TRAC Plant/Safety Analysis Code," Proceedings of the 7th International Meeting on Nuclear Reactor Thermal-Hydraulics, NURETH-7, NUREG/CP-0142, Vol. 3 (1995) 2003–2010.

11. T. D. Knight, "AP600 Large-Break Loss-of-Coolant Accident Developmental Assessment Plan for TRAC-PF1/MOD2," Los Alamos National Laboratory draft document (May 23, 1996).

12. D. R. Liles, et al., "TRAC-PF1/MOD2 Correlations and Models," Los Alamos National Laboratory Report NUREG/CR-5069 and LA-11208-MS (December 1988).

13. Technical Program Group, "Quantifying Reactor Safety Margins: Application of Code Scaling, Applicability, and Uncertainty Evaluation Methodology to a Large-Break Loss-of-Coolant Accident," EG&G Idaho, Inc. report NUREG/CR-5249 and EGG-2552 (October 1989).

14. L. M. Shotkin, "ACRS Meeting at LANL on TRAC-PWR," US NRC Reactor and Plant Systems Branch memorandum to B. W. Sheron, February 4, 1988.

15. R. Nelson and C. Unal, "A Phenomenological Model of the Thermal Hydraulics of Convective Boiling During the Quenching of Hot Rod Bundles, Part I: Thermal Hydraulic Model," *Nuclear Engineering and Design* **136**, 277–298 (1992).

16. C. Unal and R. Nelson, "A Phenomenological Model of the Thermal Hydraulics of Convective Boiling During the Quenching of Hot Rod Bundles, Part II: Assessment of the Model with Steady-State and Transient Post-CHF Data," *Nuclear Engineering and Design* **136**, 299–318 (1992).

17. C. Unal, E. Haytcher, and R. Nelson, "A Phenomenological Model of the Thermal Hydraulics of Convective Boiling During the Quenching of Hot Rod Bundles, Part III: Model Assessment Using Winfrith Steady-State, Post-CHF, Void-Fraction and Heat-Transfer Measurements and Berkeley Transient-Reflood Test Data," *Nuclear Engineering and Design* **140**, 211–227 (1993).