

AN OPTIMIZATION METHODOLOGY FOR CLOSURE DEVELOPMENT OF INTEGRATED MODEL CORRELATIONS

David A. Pimentel*, and
*Los Alamos National Laboratory,
CIC-12, MS P954, Los Alamos, New Mexico 87545, USA
Phone: 505-665-1255, Fax: 505-665-5638
E-mail: davidp@lanl.gov*

Ralph A. Nelson, Jr.
Los Alamos National Laboratory

This paper summarizes the implementation of tools within the Transient Reactor Analysis Code (TRAC-PF1/MOD2), which allows the semiautomated optimization of the reflood constitutive package. The tools included a software package external to TRAC-PF1/MOD2 that used a line-search method to minimize a generic function value given the function's partial derivative vector with respect to a set of closure coefficients used within the reflood model. Within TRAC-PF1/MOD2, the generic function was a normalized penalty function dependent on time-averaged calculated values of vapor temperature, vapor void fraction, wall-to-fluid heat-transfer rate (or wall temperature), and the respective steady-state data. The penalty function was implemented only for a one-dimensional vessel configuration because the available reflood data were taken primarily from postcritical-heat-flux tube experiments. Results suggest that such techniques can provide advantages for future model development work, but extensive expertise is required to utilize such techniques (i.e., the model developer must fully understand both the physics of the process(es) being represented and the computational techniques being employed). The computer may then be used to improve the correlation of computational results with experimental data.

Keywords: ICONE-7, Nuclear Engineering, Nonlinear Optimization, Transient Computer Code

1 INTRODUCTION

Optimization¹ problems have three basic elements:

1. A penalty function (also often called an objective function) that one wishes to minimize or maximize. For instance, in a manufacturing process, one might want to *maximize the profit* or *minimize the cost*. In fitting experimental data to a user-defined model, one might *minimize the total deviation* of observed data from predictions based on the model. Often the user would actually like to optimize several different objectives at

¹ Much of the discussion of optimization is taken from documentation from the Optimization Technology Center at Argonne National Laboratory and Northwestern University.

once. For instance, in a panel design problem, it would be nice to *minimize weight* and *maximize strength* simultaneously. Usually, the different objectives are not compatible; the variables that optimize one objective may be far from optimal for the others. In practice, problems with multiple objectives are reformulated as single-objective problems by either forming a weighted combination of the different objectives or by replacing some of the objectives by constraints.

2. A set of unknowns, or variables, that affect the value of the penalty function. In the manufacturing problem, the variables might include the *amounts of different resources used* or the *time spent on each activity*. In the fitting-the-model-to-data problem, the unknowns are the *parameters* that define the model.
3. A set of constraints that allow the unknowns to take on certain values but exclude others. For the manufacturing problem, it does not make sense to spend a negative amount of time on any activity, so we constrain all the “time” variables to be nonnegative. Constraints are not essential. In fact, the field of unconstrained optimization is a large and important one for which many algorithms and software are available. It has been argued that almost all problems *really do* have constraints. For example, any variable denoting the “number of objects” in a system can only be useful if it is less than the number of elementary particles in the known universe! In practice, however, answers that make good sense in terms of the underlying physical or economic problem can often be obtained without putting constraints on the variables.

The TRAC-PF1/MOD2 code (Spore, et al., 1993) solves the two-field fluid conservation equations and the required closure relationships coupled to structures (both heated and unheated) along with the heat diffusion conservation equation to find

$$\{P(\bar{z}, t), \mathbf{a}(\bar{z}, t), T_1(\bar{z}, t), T_v(\bar{z}, t), v_1(\bar{z}, t), v_v(\bar{z}, t), T_w(\bar{z}, t)\} , \quad (1)$$

where $\bar{z} \equiv (x, y, z)$ position in space. Equation (1) represents the variables for which the conservation equations are solved and that vary in both space and time.

2 GENERAL TWO-FLUID CLOSURE OPTIMIZATION

The general optimization problem regarding the development of closure relationships for a two-fluid code like TRAC-PF1/MOD2 becomes

$$\{P(\bar{Z}, \{\mathbf{c}_{tp}\}), \mathbf{a}(\bar{Z}, \{\mathbf{c}_{tp}\}), T_1(\bar{Z}, \{\mathbf{c}_{tp}\}), T_v(\bar{Z}, \{\mathbf{c}_{tp}\}), v_1(\bar{Z}, \{\mathbf{c}_{tp}\}), v_v(\bar{Z}, \{\mathbf{c}_{tp}\}), T_w(\bar{Z}, \{\mathbf{c}_{tp}\})\} , \quad (2)$$

where the equation set is that required to yield Eq. (1) plus a penalty function, \mathbf{f} , and $\bar{Z} \equiv (x, y, z, t)$. The thermal-hydraulic system variables are $\{P, \mathbf{a}, T_1, T_v, v_1, v_v, \text{ and } T_w\}$, and $\{\mathbf{c}_{tp}\}$ represents the set of closure coefficients. The penalty function is arbitrary, depending upon the end result desired; however, for the sake of discussion, we will define it initially to be

$$\mathbf{f} = \sum_{j=1}^J \sum_{k=1}^{K_j} \mathbf{g}_j (\mathbf{b} - 1)_{jk}^2 , \quad (3)$$

where \mathbf{b} represents the ratio of a code-calculated quantity to its experimental measurement, K_j is defined as the number of data points available from an experiment that correspond to the chosen calculated quantity, and J is the number of calculated quantities to be optimized (i.e., T_w , \mathbf{a} , q'' , ...). Also, \mathbf{g} is a weighting factor that is defined by the analyst, and it is to be used to emphasize or de-emphasize a specific quantity. As an example, one might choose T_w as the only variable to optimize, which would yield

$$\mathbf{f} = \sum_{k=1}^K \left(\frac{T_{w,calc}}{T_{w,data}} - 1 \right)_k^2 \quad (4)$$

as the penalty function where K is defined as the number of thermocouples used in an experiment. This would be the only possible comparison to be made for an old postcritical-heat-flux tube experiment.

For a single closure coefficient, $\{c_{tp}\} = c_1$, one can argue the solution might be unique because one equation has been added to the original equation set; however, for $tp \geq 2$, the system must be underspecified or underdetermined. This implies the solution is not unique, and the determination of c_n would depend upon $\{c_{tp}\}$ where $tp \leq n - 1$.

2.1 Optimization Techniques

A number of different techniques are available to optimize different problems. Figure 1 is a visual summary of these techniques and should demonstrate the massive size of this field of study. It is not our purpose here to review this technology, only to note it. For a better understanding, the reader is referred to the Optimization Technology Center¹ and to Moré and Wright (1993) for information related to available software.

The basic structure of the problem we wish to optimize has been given above, and additional details are given below. The solution method was chosen to be gradient-based because the initial coefficient estimates were assumed to be close enough to the final solution to allow a gradient-based method to quickly find “a” solution. This solution will represent a local minimum, and care was exercised to limit the solution to a reasonable solution space, thus the problem is a constrained one. Also, because of various data comparisons to be made, the problem has multiobjective functions. These can be included within the single function, as represented by Eq. (3). The reader should realize that this choice of optimization method is not absolute and that there are advantages to nonderivative based methods that might be considered in the future.

The use of a gradient-based optimizer to determine $\{c_{tp}\}$ in Eq. (2) requires the generation of

$$\frac{\partial \mathbf{b}(\bar{\mathbf{Z}})}{\partial c_{tp}} = \left\{ \frac{\partial P}{\partial c_{tp}}, \frac{\partial \mathbf{a}}{\partial c_{tp}}, \frac{\partial T_l}{\partial c_{tp}}, \frac{\partial T_v}{\partial c_{tp}}, \frac{\partial v_l}{\partial c_{tp}}, \frac{\partial v_v}{\partial c_{tp}}, \frac{\partial T_w}{\partial c_{tp}} \right\}(\bar{\mathbf{Z}}) \quad (5)$$

¹ <http://www-c.mcs.anl.gov/home/otc/>

for all t_p . Determination of these partial derivatives has historically been done either analytically and then programmed, or by finite-difference approximations for the variable of interest. Both of these methods provide their own problems and are particularly difficult within the framework of a complex code such as TRAC-PF1/MOD2.

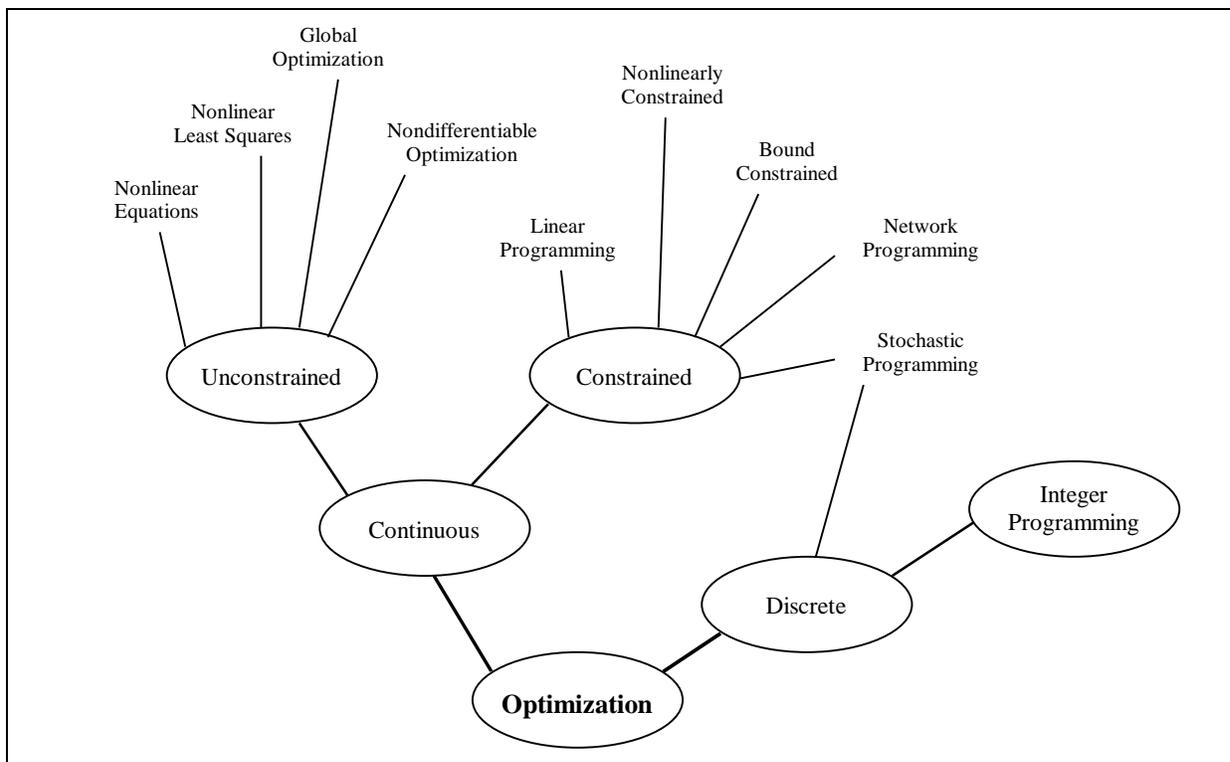


Fig. 1. The “Optimization Tree” shows various optimization techniques' interrelationships.

Fortunately, software has been developed that allows automatic generation of these derivatives. Some of this software is in its second generation and thus provides a validated configuration. The software selected for use within our work was ADIFOR¹ (Automatic Differentiation of FORTRAN Code). ADIFOR is a tool for the automatic differentiation of FORTRAN 77 programs. Given FORTRAN 77 source code and a user's specification of dependent and independent variables, ADIFOR will generate an augmented derivative code that computes the original result(s) plus the partial derivatives of all specified dependent variables with respect to all of the specified independent variables.

2.2 Technical Issues Associated with Closure Optimization

There are several issues faced in the optimization of closure relationships that are not encountered in typical optimization problems. First, most systems that are optimized are not transient in nature. Their behavior is steady state. Second, the finite-difference formulation used in TRAC-PF1/MOD2 removes the implicit dependence of measurements at a particular location on upstream phenomena.

¹ ADIFOR is a collaborative project between the Mathematics and Computer Science Division at Argonne National Laboratory and the Center for Research on Parallel Computation at Rice University.

Even during “steady-state” experiments, two-phase systems often experience oscillatory behavior; thus, the term steady state is within the Class 2 transient problem discussed by Nelson, et al. (1998) and denoted by

$$\frac{d\bar{q}}{dt} = \frac{\partial \bar{q}}{\partial t} = \frac{d\bar{T}_w}{dt} = \frac{d\bar{a}}{dt} = \frac{d\bar{T}_f}{dt} = \dots = 0, \quad (6)$$

where \bar{q} , \bar{T}_w , \bar{a} , \bar{T}_f , ... represent the time-averaged values of heat flux, wall temperature, void fraction, local phasic temperature, etc. As discussed by Nelson, et al. (1998), satisfying this condition requires that the time-averaging window be sufficiently long so that the average is independent of the length of the window or the point at which the window is started.

In running “steady-state” experiments, data are commonly time-averaged before being reported. In simulations of these experiments, it is common for the code to oscillate. If the code is simulating true physical behavior, time averaging of the code results should be performed to compare them with the experimental data. Time averaging has been developed as part of the optimization process because optimization techniques cannot be expected to satisfactorily perform for an oscillating system.¹ An additional desirable comparison is the frequency content of the averaged information, both experimental and simulated. Whereas it is possible to generate this information within the simulation, its value is limited because of the general lack of experimental information in this regard. It would be useful in the future if this experimental information were gathered whenever possible.

For codes such as TRAC-PF1/MOD2 to provide valid predictions of these “transient” experiments, the transient must be slow enough for the quasi-steady condition to be satisfied. Nelson, et al. (1998) discusses the requirements on this assumption, such that

$$\frac{\partial \bar{q}}{\partial t} = 0. \quad (7)$$

To utilize this type of experiment, short time intervals where Eq. (7) is satisfied must be considered, and the optimizer must rerun this interval repeatedly using the “restart” capability of the code where the coefficients are adjusted in each restart. The length of each restart run must be sufficient to allow the new coefficients to propagate effects within the system, and then a meaningful time-average at the end of the run is produced.

The finite-difference formulation used in TRAC-PF1/MOD2 removes the implicit dependence of measurements at a particular location on upstream phenomena. For example, the vapor superheat downstream of the critical heat flux point is an integral property involving phasic wall heat transfer, interfacial phasic heat transfer, and interfacial phasic drag along the channel. The measurement of the vapor temperature at a particular point within the flow stream indicates the result of this integration process before that point. Whereas the finite-difference method solves this problem, one cannot determine the derivative of the vapor temperature with respect to one of the coefficients of an upstream process. This limits our

¹ The transport time of the flow and the possible time constant of the walls and phases limit our ability to potentially optimize this situation. If these were small enough so that coefficient changes would be reflected correctly each timestep, an optimizer might be able to optimize the coefficients. However because this is not the case and several seconds of transient simulation are required to reflect a new coefficient, one must provide average behavior to the optimizer.

ability to investigate some sensitivities and to optimize some behaviors within the closure relationships.

2.3 Optimization Methodology

Whereas Eqs. (2), (3), and (5) represent the general optimization problem for closure development, it is a problem that we cannot solve. The difficulty lies in the fact that \mathbf{b} does not exist for all the independent parameters reflected in Eq. (1). Typically, because of the lack of hydrodynamic data, the penalty function cannot be constrained to assure that the proper hydrodynamic state exists. For example, using an old post-CHF tube experiment, one would know the pressure, mass flux, heat flux, and a set of tube wall temperatures. This knowledge would allow a TRAC-PF1/MOD2 input model to be built and a penalty function like Eq. (4) to be defined. However, this system is underspecified, and hydrodynamic information is not part of the penalty function; therefore, determination of the “proper hydro-state” cannot be assured.

2.3.1 Original NRC SET and Code Development Philosophy

To represent the original NRC philosophy, write

$$\{c_{tp}\} = \{\{c_{tp}\}_{SET1}, \{c_{tp}\}_{SET2}, \{c_{tp}\}_{SET3}, \dots\}, \quad (8)$$

where the subscript SET1 represents one Separate Effects Test (SET) defined to produce a measure of a particular closure phenomena. One could use the data from this experiment to define $\{c_{tp}\}_{SET1}$.¹ Through a process of running a number of SETs, one could then arrive at a complete set of closure relationships. The general philosophy represented within Eq. (8) can also be more flexible than that just noted. Some of the SET experiments may be designed to produce specific coefficient sets,² and for the sake of discussion, one might denote these to be $\{c_{tp}\}_{SET1}$ and $\{c_{tp}\}_{SET3}$. Others may then be built off these initial sets. For example, $\{c_{tp}\}_{SET2}$ might be developed based upon $\{c_{tp}\}_{SET1}$. An experiment where the walls are adiabatic followed by one where they are heated would be an example of this sequence. The initial adiabatic experiment might be used to develop the drag coefficients whereas the heated experiment might provide the wall heat-transfer coefficients assuming the drag coefficients are correct.

2.3.2 Closure Development Using Nonlinear Optimization

The goal of the current closure development is to remain as true to the process implied by Eq. (8) as possible realizing that there are significant constraints upon the data available for phenomena to be modeled.

Optimization may also allow greater flexibility in the development of $\{c_{tp}\}_{SET1}$ and $\{c_{tp}\}_{SET2}$ than that just noted above. One can develop the initial sets of quantities as discussed above. These sets can then be used as a starting point for a solution of the combined sets of coefficients and data (i.e., $\{c_{tp}\}_{SET1}$ is not constant during final development). This may produce a better set of coefficients than might otherwise be obtained; however, this point requires further investigation.

¹ One can read $\{c_{tp}\}_{SET1}$ as correlation or closure relationship in this case.

² Independent of the other sets.

2.3.3 Generalized Penalty Function

The optimization process described above requires a set of automated tools for recorrelation of the TRAC-PF1/MOD2 closure coefficients. The method selected involves an iterative implementation of the penalty function defined by Eq. (3). The penalty function is nondimensionalized and normalized to produce

$$\mathbf{f} = \sum_{e=1}^E \sum_{j=1}^J \sum_{k=1}^{K_{ej}} \frac{r \mathbf{g}_j}{K_{ej}} (\hat{\mathbf{b}} - 1)_{ejk}^{\mathbf{x}}, \quad (9)$$

where E is the number of TRAC-PF1/MOD2 input decks to be used for each determination of a penalty function value and its gradient; r is a dynamically determined convergence acceleration parameter that was patterned from Fox (1971), $\mathbf{x} = 2$, and

$$\hat{\mathbf{b}} = \frac{X_{calc}}{X_{data}}, \quad (10)$$

and \mathbf{x} is some quantity that is either calculated or measured. It follows that the gradient of the penalty function in Eq. (10) is defined as

$$\frac{\partial \mathbf{f}}{\partial \mathbf{c}_{tp}} = \sum_{e=1}^E \sum_{j=1}^J \sum_{k=1}^{K_{ej}} \left\{ \frac{r \mathbf{g}_j \mathbf{x}}{K_{ej}} (\hat{\mathbf{b}} - 1)_{ejk}^{\mathbf{x}-1} \left(\frac{\partial \hat{\mathbf{b}}}{\partial \mathbf{c}_{tp}} \right)_{ejk} \right\}, \quad (11)$$

where $\hat{\mathbf{b}}$ is averaged over a specified region of time. These partial derivatives can be calculated within TRAC-PF1/MOD2 by implementing subroutines generated by ADIFOR.

It is important to check that the averaged quantity $\hat{\mathbf{b}}$ is stationary to reflect behavior around a limit attractor or strange attractor. If this is not the case, the characteristics of the behavior must be investigated.

3 OPTIMIZATION IMPLEMENTATION

The implementation of the optimization discussed in Section 2 required a software package to perform the minimization of Eq. (9). The selected Minimizer¹ is a Limited Memory Quasi-Newton Solver using a line search method that implements a direct primal method to determine a search direction (Byrd, et al., 1994) (Zhu, et al., 1994). The Optimizer² is a stand-alone program that performs the optimization of the TRAC-PF1/MOD2 code.

3.1 Optimization Convergence Criteria

The Optimizer has been equipped with a convergence criteria based upon the tunable parameter values (i.e., independent variables of the penalty function), and the Minimizer has intrinsic convergence criteria based upon both user specified values and machine precision.

¹ LBFGS-B version 2.3 written in FORTRAN 77.

² FORTRAN 77 driver to run Minimizer and TRAC.

First, the Optimizer considers the relative change of the tunable parameters with a stopping criterion of

$$\left| \frac{c_{tp,n} - c_{tp,n+1}}{c_{tp,n+1}} \right| \leq 10^{-14}, \quad (12)$$

which, when active, will preempt a restart in the steepest direction for a failed line search (Zhu, et al., 1994). Second, the Minimizer considers the relative change of the penalty function value

$$\frac{(f_n - f_{n+1})}{\max(|f_n|, |f_{n+1}|, 1)} \leq JM_p \begin{cases} \mathbf{J} = \text{user-defined factor} \\ M_p = \text{machine precision} \end{cases}, \quad (13)$$

where the machine precision is typically on the order of 10^{-16} for double precision accuracy. For the TRAC-PF1/MOD2 optimization process, $\mathbf{J} \equiv 10^{12}$. Finally, the Minimizer considers the projected gradient ($\text{proj } g$), which is the projection of the penalty function's gradient vector onto the space tangent to the active bounds. When the infinity norm of this projection meets the criterion

$$\|\text{proj } g\|_{\infty} \leq \mathbf{e}_{grad}, \quad (14)$$

the optimization process is terminated. The test of Eq. (14) will be very difficult to satisfy if \mathbf{e}_{grad} is set to anything smaller than $\sqrt{M_p}$. For the purposes of the TRAC-PF1/MOD2 optimization, \mathbf{e}_{grad} is set to 10^{-5} . The subscript n in Eq. (12) through Eq. (14) refers to a specific Minimizer iteration.

3.2 Optimization Algorithm

The following algorithm outlines the method of optimizing TRAC-PF1/MOD2 with respect to specified closure parameters:

1. Start: choose initial values for r and $\{c_{tp}\}$ (realize that $\{c_{tp}\}$ will be divided into two subsets, one set that remains constant and a much smaller set that will be re-correlated). Also, the Optimizer requires knowledge of the type of independent variables' bounds and the test numbers corresponding to the desired TRAC-PF1/MOD2 input decks.
2. If the convergence criterion of Eqs. (12) through (14) is true, then quit the algorithm.
3. Run TRAC-PF1/MOD2 using the user-specified input decks to calculate the penalty function and its gradient with respect to $\{c_{tp}\}_{re-correlate}$.
4. If TRAC-PF1/MOD2 fails to generate \mathbf{f} and $\frac{\partial \mathbf{f}}{\partial c_{tp}}$, then the current values of $\{c_{tp}\}_{re-correlate}$ are inappropriate for the current TRAC-PF1/MOD2 input. Reset

$\{c_{tp}\}_{reccorrelate} = \frac{\{c_{tp}\}_{reccorrelate,current} + \{c_{tp}\}_{reccorrelate,previous}}{2}$ and get the first user-specified input deck. Return to step 3.

5. If another input deck is specified for TRAC-PF1/MOD2, get it and return to step 3.

6. Generate new $\{c_{tp}\}_{reccorrelate}$ values¹ based upon the current values of \mathbf{f} and $\frac{\partial \mathbf{f}}{\partial c_{tp}}$.

7. If the norm of the gradient, $\left\| \frac{\partial \mathbf{f}}{\partial c_{tp}} \right\| \leq \mathbf{z}$, then $r \Leftarrow \Omega r$ where Ω and \mathbf{z} are user-defined parameters with typical values of 10 and 10^{-2} respectively.²

8. Return to step 2.

3.3 Interfacial Heat-Transfer Optimization

Using ADIFOR, the product of the interfacial heat-transfer coefficient and the interfacial area was differentiated with respect to several closure parameters and the vapor temperature.

Within the subroutine generated by ADIFOR, the dependence of vapor temperature upon the tunable parameters was not visible, so the partial derivative in Eq. (12),

$\left. \frac{\partial \hat{\mathbf{b}}}{\partial c_{tp}} \right|_{j=1} = \frac{1}{T_{vap,data}} \frac{\partial T_{vap,calc}}{\partial c_{tp}}$, was determined using

$$\frac{\partial T_{vap,calc}}{\partial c_{tp}} = \frac{\partial \dot{q}_{vap,calc}}{\partial c_{tp}} \left(\frac{\partial \dot{q}_{vap,calc}}{\partial T_{vap,calc}} \right)^{-1} \quad (15)$$

where c_{tp} is an arbitrary closure parameter and

$$\dot{q}_{vap,calc} = (hA)_{vap,calc} (T_{vap,calc} - T_{sat}) \quad (16)$$

The ADIFOR-generated routine was used to calculate all of the derivatives on the right-hand side of Eq. (15).

3.4 Interfacial-Drag Optimization

Once again, using ADIFOR, the interfacial-drag coefficient, C_i , was differentiated with respect to several closure parameters within TRAC-PF1/MOD2.

¹ Performed by the Minimizer.

² This step was deactivated with $\Omega < 0$, and r had a constant value of one for the optimization of TRAC-PF1/MOD2 documented here.

Within the subroutine generated by ADIFOR, the dependence of the drag coefficient upon the void fraction was not visible; therefore, the partial derivative $\frac{\partial \mathbf{a}_{calc}}{\partial C_i}$ was determined from second-order polynomial regressions of the time-averaged values of \mathbf{a}_{calc} and C_i at each hydro-cell-centered elevation. These polynomials were calculated using \mathbf{a}_{calc} and C_i at five axial positions with the current cell as a center. Using the ADIFOR generated gradient vector in conjunction with the regressions' derivatives allowed the partial derivative in Eq. (11) $\left. \frac{\partial \hat{\mathbf{b}}}{\partial c_{tp}} \right|_{j=2} = \frac{1}{\mathbf{a}_{data}} \frac{\partial \mathbf{a}_{calc}}{\partial c_{tp}}$ to be determined using

$$\frac{\partial \mathbf{a}_{calc}}{\partial c_{tp}} = \frac{\partial \mathbf{a}_{calc}}{\partial C_i} \frac{\partial C_i}{\partial c_{tp}}. \quad (17)$$

3.5 Wall Heat-Transfer Optimization

Using ADIFOR, the total wall heat flux was differentiated with respect to several closure parameters and the heated wall temperature.

The objective of this component of the overall penalty function was to allow the user to select the system variable to optimize (q''_{wall} or T_{wall}) depending upon the available experimental data. For the q''_{wall} option, the ADIFOR routine was capable of calculating the partial derivative in Eq. (12), $\left. \frac{\partial \hat{\mathbf{b}}}{\partial c_{tp}} \right|_{j=3} = \frac{1}{q''_{wall,data}} \frac{\partial q''_{wall,calc}}{\partial c_{tp}}$, directly; however, the chain rule had to be implemented for the T_{wall} option,

$$\left. \frac{\partial \hat{\mathbf{b}}}{\partial c_{tp}} \right|_{j=3} = \frac{1}{T_{wall,data}} \frac{\partial T_{wall,calc}}{\partial c_{tp}} = \frac{1}{T_{wall,data}} \frac{\partial T_{wall,calc}}{\partial q''_{wall,calc}} \frac{\partial q''_{wall,calc}}{\partial c_{tp}}. \quad (18)$$

4 RESULTS OF OPTIMIZATION STUDIES

The optimization task associated with development of a reflood model was incomplete because of software and hardware limitations. However, some encouraging results shown here demonstrate the potential of the optimization methodology.

Figures 2 and 3 show an optimization study that was undertaken with Winfrith (Swinnerton, et al., 1988, and Swinnerton, et al., 1990) Experiment 405 corresponding to the optimization work done on the interfacial drag correlations. Nelson et al., (1998) indicated the rough/wavy interfacial-drag model produced too little drag. Figure 2 indicates the behavior of TRAC-PF1/MOD2 for the repeated runs (TRAC-PF1/MOD2 iterations) where the optimizer adjusted the correlation multipliers, f_{rw} and f_{sm} , on the rough/wavy and smooth regime drag correlations simultaneously. As f_{rw} is raised, the penalty function decreases as expected. Only a slight increase of f_{sm} occurred. Figure 3 shows the behavior of the gradient (i.e., both partial derivatives) of the penalty function. Note that the gradient's magnitude is significantly reduced to an apparent asymptote indicating a local minimum of the penalty function, \mathbf{f} .

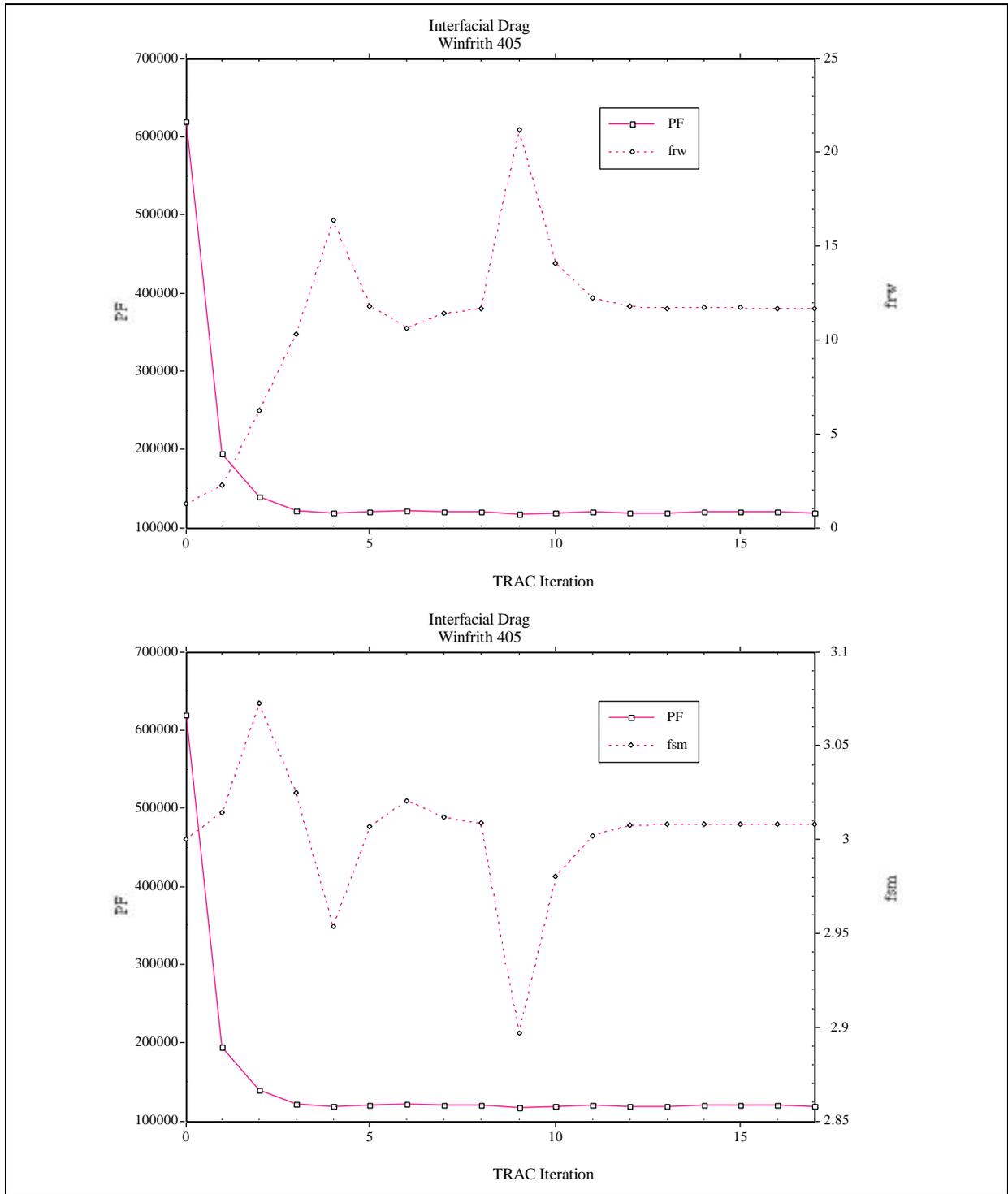


Fig. 2. The penalty function and coefficients **frw** and **fsm** are shown as functions of each iteration of TRAC using the post-CHF upflow data and TRAC input deck for Winfrith Experiment 405.

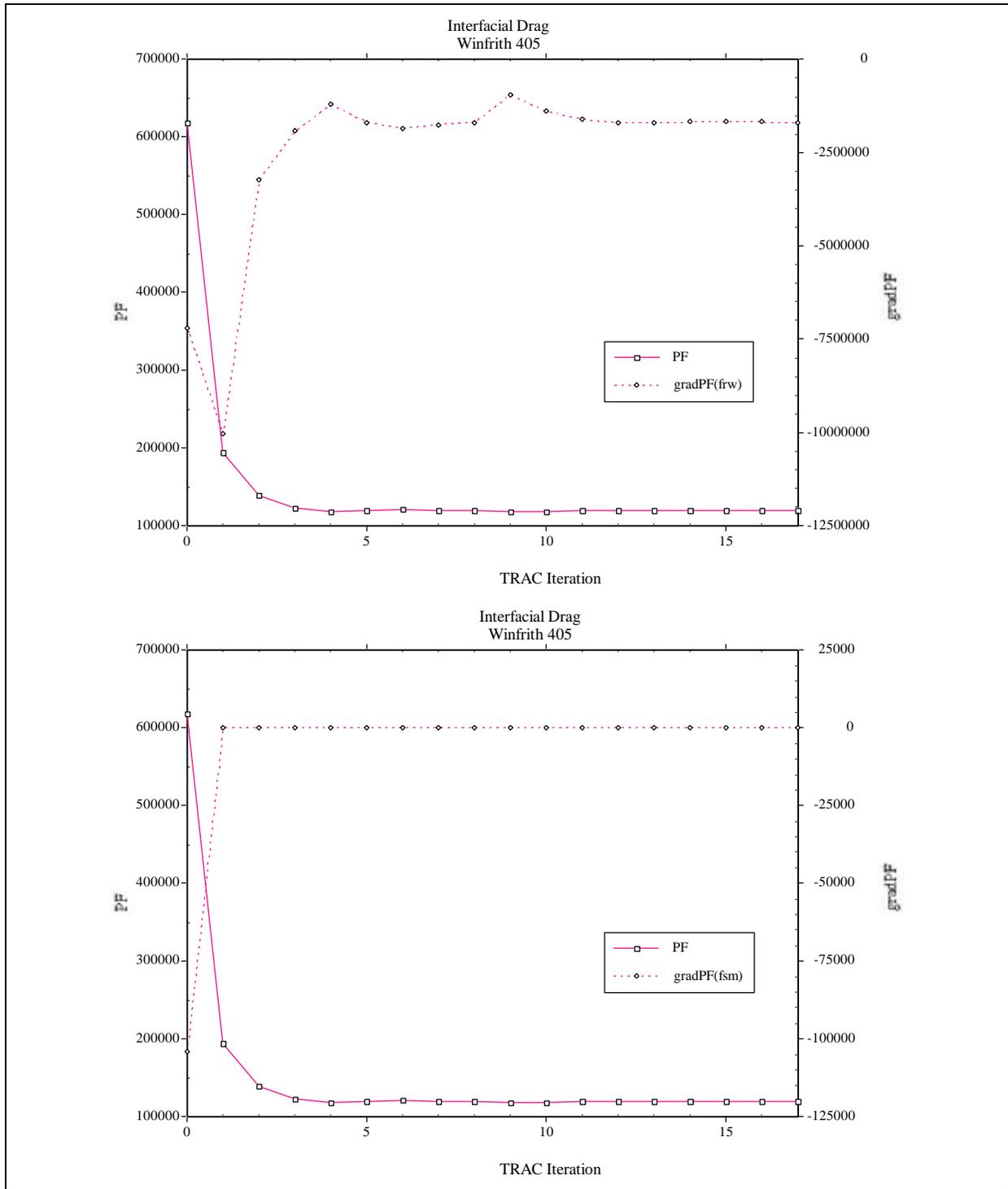


Fig. 3. The partial derivatives (gradPF) of the penalty function with respect to the coefficients, **frw** and **fsm**, are shown as functions of each iteration of TRAC using the post-CHF upflow data and TRAC input deck for Winfrith Experiment 405.

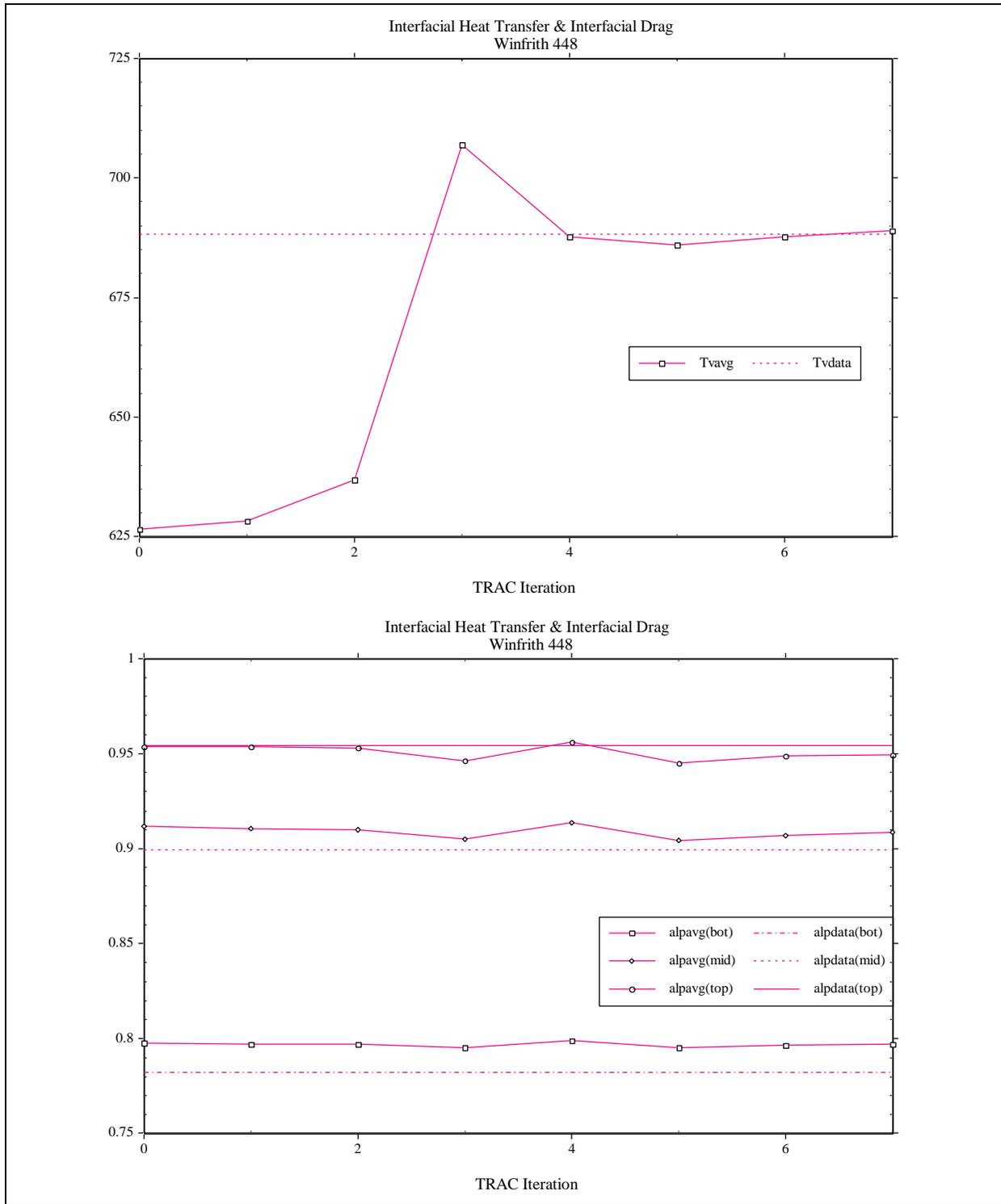


Fig. 4. The vapor temperature and void fractions corresponding to the experimental measurement elevation(s) is shown as a function of each iteration of TRAC as the coefficients **fgam2** and **ffd** are tuned using the post-CHF upflow data and TRAC input deck for Winfrith Experiment 448.

Winfrith experiment 448 was used to investigate the optimization methodology's effectiveness when both the interfacial drag and interfacial heat-transfer correlations are modified. Figure 4 shows the effect upon the calculated vapor temperature when multiplier $fgam2$ is increased from its initial value of ~ 30 to ~ 59 , and the multiplier ffd is decreased from its initial value of ~ 0.5 to ~ 0.25 . The results of this combined optimization can be seen in Figure 4 where the prediction of the vapor superheat is improved, whereas a minimal change is seen in the void fraction because its original prediction was reasonable.

REFERENCES

- Byrd, R. H., Nocedal, J., and Schnabel, R. G.: *SIAM J. Scientific Computing*, Vol. 16:5, pp. 1190–1208 (1994).
- Fox, R. L.: "Optimization Methods for Engineering Design," Addison-Wesley Pub. Co., Massachusetts (1971).
- More, J. J. and Wright, S. J.: "Optimization Software Guide," SIAM Pub. (1993).
- Nelson, R. A., et al.: "Reflood Completion Report–Volume I–A Phenomenological Thermal-Hydraulic Model of Hot Rod Bundles Experiencing Simultaneous Bottom and Top Quenching and an Optimization Methodology for Closure Development," Appendix A, Los Alamos National Laboratory document LA-UR-98-3043 (April 1998).
- Spore, et al.: "TRAC-PF1/MOD2: An Advanced Best-Estimate Computer Program for Pressurized Water Reactor Thermal-Hydraulic Analysis," Los Alamos National Laboratory report LA-12031-M, NUREG/CR-5673 (1993).
- Swinerton, D., Hood, M. L., and Pearson, K. G.: Winfrith United Kingdom Atomic Energy Authority report AEEW-R 2267 (1988).
- Swinerton, D., Savage, R. A., and Pearson K. G.: AEA Thermal Reactor Services, Physics and Thermal Hydraulic Div. report AEA-TRS-1045, Winfrith, United Kingdom Atomic Energy report AEEW-R 2503 (1990).
- Zhu, C., et al.: "L-BFGS-B: FORTRAN Subroutines for Large Scale Bound Constrained Optimization," Northwestern University document NAM-11 (1994).