IMPROVE SSME POWER BALANCE MODEL

George C. Marshall Space Flight Center
and
The University of Alabama in Huntsville

FINAL REPORT

submitted to

NASA Headquarters
Washington, D.C.

Principal Investigator:

Dr. Gerald R. Karr, Chairman
Department of Mechanical and Aerospace Engineering
University of Alabama in Huntsville

NASA Contract NAS8-36955/D.O. 115

August 1992

(NASA-CR-184464) IMPROVE SSME POWER BALANCE MODEL Final Report
(Alabama Univ.) 83 p

N93-10756

Unclass

63/20 0118094
<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>Background</td>
</tr>
<tr>
<td>2.0</td>
<td>SSME PBM Logic Assessment</td>
</tr>
<tr>
<td>3.0</td>
<td>Reconciliation Model</td>
</tr>
<tr>
<td>4.0</td>
<td>Recommendations</td>
</tr>
<tr>
<td>5.0</td>
<td>References</td>
</tr>
</tbody>
</table>
1.0 BACKGROUND

As detailed in the original Scope of Work for this research effort, the principal investigator was to improve the steady-state power balance model (PBM) of the Space Shuttle Main Engine (SSME) in a three phase effort. A summary of the tasks in each phase is given below.

Phase 1: Construct software to facilitate SSME performance prediction and test data validation.

Phase 2: Review computational logic within the current version of the SSME PBM and implement programming structure.

Phase 3: Develop programming logic to improve the physical consistency of routing routines within the SSME PBM.

After discussions with John Butas of the NASA/MSFC Propulsion Laboratory in January of 1992, the contract Scope of Work for Phase 3 was modified to place greater emphasis on refinement of computational tools initially developed in Phases 1 and 2, and to support evaluation of recently acquired Technology Test Bed (TTB) data. The primary effort during Phase 3 was continued development of software intended to support integration of TTB test data with SSME performance predictions from the steady-state power balance model. In addition, variational analyses of both TTB test data and PBM predictions were performed. Results of these analyses were compared to evaluate the computational integrity of the SSME steady-state power balance model.

A considerable portion of the contract effort was dedicated to development and testing of a formal strategy for reconciling uncertain test data with physically limited computational
prediction. This emphasis was motivated by the availability of an extensive and highly organized SSME performance data base from the Engine 3001 test program, and by serious inconsistencies in power balance model predictions.

A review of SSME steady-state power balance model function is provided in Section 2 of this report. Specific weaknesses in the logical structure of the current PBM version are described with emphasis given to the main routing subroutines BAL and DATRED. Selected results from a variational analysis of PBM predictions are compared to TTB variational study results to assess PBM predictive capability.

The motivation for systematic integration of uncertain test data with computational predictions based on limited physical models is provided in Section 3. The theoretical foundation for the reconciliation strategy developed in this effort is presented, and results of a reconciliation analysis of the SSME high pressure fuel side turbopump subsystem are examined. Specific recommendations are presented in Section 4.
The Space Shuttle Main Engine power balance model is a FORTRAN based software package developed by the Rocketdyne Division of Rockwell International. It is used to predict operating characteristics and performance of the SSME under steady-state conditions. Approximately 800 SSME temperatures, pressures, flow rates, shaft speeds, and other hardware performance parameters are calculated by the power balance model.

The current version of PBM has a number of analysis options. The standard power balance analysis option determines fluid and flow properties throughout the entire engine system assuming nominal hardware performance characteristics. In addition, there is a data reduction analysis option which uses actual test data to define hardware operating characteristics such as efficiencies and flow multipliers of a specific SSME. The base balance option is used to further define engine hardware characteristics by matching nine parameters to data reduction output.

Although conceptually a powerful prediction tool, PBM exhibits a number of significant shortcomings. Documentation of the computational, physical, and functional operation of PBM has not been rigorously maintained and is inadequate. Moreover, recent tests have demonstrated that PBM predictions fail to satisfy fundamental energy balance relations within all engine subsystems [1]. As a result, confidence in PBM predictions has been degraded and software utility diminished.

In order to assess the logical integrity of PBM, five sources
of information were utilized:

1) detailed flowcharts of the main analysis routing routines BAL and DATRED
2) iteration loop diagrams for routines BAL and DATRED
3) detailed translation of routines BAL and DATRED
4) direct source code inspection
5) analysis of variations (ANOVA) comparisons of PBM predictions and TTB data.

An overview of the main routing logic was obtained from sources 1 and 2. The physical and logical consistency of individual lines of code was examined using sources 3 and 4. The integrity of the PBM prediction process was evaluated using comparisons from source 5.

Detailed, computer-generated flowcharts of PBM subroutines BAL and DATRED were obtained from NASA/MSFC/EP52 and examined for logical structure. Copies of these flowcharts were previously presented in the Phase 2 final report. A high level of vertical connectivity (logic feedback) is obvious upon examination of the complex BAL flowchart. Subroutine logic is highly integrated and structured segmentation cannot be achieved without fundamental and costly code modifications beyond the scope of this effort. Subroutine DATRED has a more sequential logic process, however, structured segmentation of the code was not attempted for reasons described below.

Iteration loop logic diagrams for subroutines BAL and DATRED were also presented in the Phase 2 final report. Subroutine BAL contains four multivariate iteration loops for solution of
simultaneous nonlinear relations, and twenty-five univariate iteration loops for solving individual nonlinear relations, all of the Newton-Raphson type. Five deep nesting of iteration sequences is found in BAL with high level multivariate iteration loops traversing virtually the entire routine. Intersection of major iteration sequences inhibits structuring of code logic. Iteration nesting and crossover are not as severe in subroutine DATRED, however, both BAL and DATRED use a segmented solution strategy on restricted subsets of the fluid and flow governing equations. Values of the subset solution variables are iteratively matched with both nested and sequential subset solutions. This type of segmented solution approach with matching is generally less efficient than robust global strategies for solution of nonlinear equations [see, e.g. 2].

In order to facilitate interpretation of PBM logic, a software translator package was constructed in the C programming language. The translator substitutes variable definitions found in PBM documentation in place of the variable names in PBM. The result is a readable document describing PBM function in detail. Translations of subroutines BAL and DATRED were included in the Phase 2 final report. These translations were used to study software logic line by line.

A detailed examination of BAL and DATRED logic indicates that many computations are empirical and/or heuristic. This conclusion is based on comparison of the actual number of SSME flow network controllers with the number of independent variables used by PBM to predict a variety of engine operating conditions. Since the SSME is
a feedback dominated flow network, each control setting can be expected to affect operating characteristics throughout the engine. However, many PBM computations are based on reduced dependencies. This is especially evident in the data reduction routine DATRED where certain densities, temperatures, flow rates, pressures, and hardware characteristics are specified by relations depending on thrust level (or commanded chamber pressure) alone.

In order to assess the fundamental dependencies of PBM computations, a variational study of power balance predictions was performed. PBM analyses of engine number 3001 were performed by John Butas of the MSFC Propulsion Laboratory. Analysis independent parameters were set at values corresponding to control parameter settings for each of the 16 test profiles employed in the TTB Engine 3001 test program. Control parameter definitions and settings for each of the 16 TTB program tests are shown in Appendix B, Table 1.

The Engine 3001 test series was based on a Taguchi type design of experiments [3]. The matrix of control settings displayed in Table 1 was selected based on a fractional factorial test plan to facilitate data utilization. A variational analysis of TTB recorded engine operating conditions at test matrix control settings was also performed. Results of the TTB data variational study were then compared to the PBM analysis of variations. Selected parameter comparisons are displayed in Appendix A, Figures 1 through 4. The computed contribution of each individual control parameter to the variation of the performance variable listed in the title is displayed. The abscissa designations OSP, HSP, ORP,
and HRP correspond to LOX NPSP, FUEL NPSP, LOX REPRS, and FUEL REPRS respectively in the test matrix. The category COMB that appears in each figure represents contributions from control parameter combinations that cannot be individually allocated because the test program was not designed as a full factorial set of experiments.

It is obvious from Figure 1 that, within the TTB test range, low pressure fuel pump (LPFP) speed variation was only weakly affected by F7 orifice size. PBM predicted F7 contributions to LPFP speed variation were significantly greater than test results indicated. Similarly, as shown in Figure 2, TTB data indicated a significant F7 orifice size contribution (17%) to high pressure fuel pump (HPFP) discharge temperature variation that was largely absent from PBM predictions. Comparisons such as these are indicative of potential PBM weakness in assigning component level contributions to performance and in predicting operational contributions of hardware modification.

Large differences (approximately 31%) between observed and predicted power level (% RPL) and mixture ratio (M/R) contributions to coolant control valve (CCV) flow rate are indicated in Figure 3. Significant disparity between predicted and observed controller effects was not isolated to the parameters displayed in Figures 1 through 3. In general there was good agreement between predicted and observed controller contributions to pressure variation such as is shown in Figure 4 for the high pressure fuel pump (HPFP) outlet pressure. Only isolated cases of significant pressure variational differences were observed. More common were large differences
between predicted and observed contributions to temperature, flow rate, and hardware performance characteristics.

The variational analysis comparisons described above suggest that PBM does not adequately model the variation of important SSME performance parameters. This is not particularly surprising since, in many places throughout the code, physical dependencies have been replaced by "hardcoded" empirical relations that lack adequate documentation to assess application and range validity. These comparisons also suggest that the power balance model would be inadequate as a design or anomaly resolution tool. Integrity of PBM predictions can be expected only in nominal engine operating ranges over which code empirical relations were established.

Because the power balance model is a highly connected software package with significant iteration looping, it is difficult to access the overall impact of an individual code modification without significant computational testing. Simple code corrections to achieve improvements in isolated parameter prediction can have a far reaching and detrimental affect. Therefore, code maintenance and modification time will be substantially greater than for a highly structured, modular, and well documented performance model.

One of the major functions of the power balance model is to integrate test data with theoretical predictions. The weaknesses of the existing data integration procedure will be discussed in the following section, and a new integration strategy will be introduced.
One of the features of the steady-state power balance model is its ability to integrate test data into the performance prediction process. This is accomplished within the data reduction analysis option. In the data reduction process, test information is incorporated literally into predictions, and hardware operating parameters are adjusted to values consistent with this presumed pristine data.

Unfortunately, experimental data associated with a complex flow system such as the SSME is fraught with uncertainty. Maintaining operation and calibration of sensing and signal conditioning instrumentation is difficult in the severe SSME operating environment. In addition, point measurements in such a complex flow environment often include the effects of highly localized and/or secondary flow phenomena that are not characteristic of system average conditions. Literal incorporation of inaccurate test data can lead to nonphysical predictions of engine operation and erroneous assumptions concerning hardware performance. Since all test data has associated uncertainty, the pristine data assumption is inappropriate in a test information integration strategy.

Performance prediction models based on fundamental flow physics are also limited by theoretical approximations required to achieve tractable solution. For example, PBM computations assume steady-state operation throughout the engine, and provide estimates of average flow conditions using a cross-stream uniform, one-
dimensional flow approximation. In addition, thermodynamic property data for hydrogen and oxygen in SSME operating ranges has accuracy limitations. These type assumptions and limitations necessarily restrict the accuracy of theoretical model predictions and present an additional source of uncertainty for data integration strategies.

The above observations suggest that any systematic procedure for integrating experimental data and theoretical predictions should recognize both data uncertainty and model limitations. The objective of the reconciliation development effort undertaken as part of this study was to construct a logical strategy for integrating uncertain test data with limited theoretical predictions in order to determine most plausible SSME operating conditions.

A heuristic yet logical procedure for achieving systematic data integration was presented in the Phase 1 final report of this study. A refinement of this reconciliation procedure has been developed in Phase 3. The basis of the new method rests on the principle that the mean of experimental observations reflects most probable, but not absolute, engine operating conditions. If measured engine operating properties are assumed to be independent, normally distributed, random variables, then the most probable engine state will maximize the property joint probability density function (pdf) subject to constraints imposed by physical laws. A mathematical expression for this state pdf is given below:
where

\( X_i \) - adjusted value of property \( i \)
\( \sigma_i \) - standard deviation of property \( i \)
\( \mu_i \) - mean of property \( i \)
\( d_i \) - deviation of adjusted property \( i \) from its mean (measured value) \( d_i = X_i - \mu_i \)
\( F \) - joint probability density function of state
\( k \) - number of properties
\( (X_1, \ldots, X_k) \) - state of system

Properties in the relation above include measured flow rates, temperatures, and pressures throughout the engine system.

The state pdf is a maximum when the expression in brackets in the exponent of \( e \) is minimized. In the absence of physical constraints this minimum would occur when all the \( d_i \) are zero, or when the value of each property \( i \) is at its mean. Therefore, experimental property measurements are assumed to correspond to the property means in equation 1, and the \( d_i \) are adjustments from measurement values required to adequately satisfy physical constraints, including mass and energy conservation requirements as well as second law limits.

A robust data reconciliation strategy must also incorporate measurement system uncertainty limits in addition to physical constraints. The problem of determining most plausible engine operating conditions can thus be reduced to a mathematical programming problem of the form:
maximize \( Z = F(d_1, \ldots, d_k) \) by selection of \((d_1, \ldots, d_k)\) assuming constant \((\sigma_1, \ldots, \sigma_k)\) subject to

physical constraints for each engine subsystem \( j \)

\[
\begin{align*}
| \text{mass flow imbalance}_j | & < L_{\text{flow-,}j} \\
| \text{energy imbalance}_j | & < L_{\text{energy-,}j} \\
| \text{entropy production}_j | & > 0
\end{align*}
\]

uncertainty limits for measurements at each node \( i \)

\[
\begin{align*}
| \text{mass flow adjustment}_i | & < U_{m-,i} \\
| \text{pressure adjustment}_i | & < U_{p-,i} \\
| \text{temperature adjustment}_i | & < U_{T-,i}
\end{align*}
\]

where

\[
\begin{align*}
\text{mass flow imbalance} &= \text{ImbM} = \sum_{\text{inlets}} m + \sum_{\text{outlets}} m \\
\text{energy imbalance} &= \text{ImbE} = \sum_{\text{inlets}} m \left[ h + \frac{m^2}{(2\rho^2A^2)} \right] \\
&\quad - \sum_{\text{outlets}} m \left[ h + \frac{m^2}{(2\rho^2A^2)} \right] \\
&\quad + Q - W \\
\text{entropy production} &= \text{ImbS} = \sum_{\text{inlets}} m \left[ s \right] \\
&\quad - \sum_{\text{outlets}} m \left[ s \right] \\
&\quad + \frac{Q}{T_0}
\end{align*}
\]
The mathematical programming problem stated in formulation 2 above is highly nonlinear. Without loss of generality, the objective function \( Z = F \) can be replaced by the exponent of \( e \) in equation 1. If, in addition, the imbalance relations are approximated as first order truncated Taylor series expansions in the nodal adjustment values \( d \), the mathematical programming problem reduces to the following:
minimize \( Z = \sum_{i=1}^{k} \frac{d_i^2}{2\sigma_i^2} \quad k = \text{number of measurements} \)

subject to

linearized forms of the physical constraints for each engine subsystem \( j \)

\[
| \text{LimbM}(d)_j | < L_{\text{flow}-j} \\
| \text{LimbE}(d)_j | < L_{\text{energy}-j} \\
| \text{LimbS}(d)_j | > 0
\]

measurement uncertainty limits for each node \( i \)

\[
| d_{m-i} | < U_{m-i} \\
| d_{p-i} | < U_{p-i} \\
| d_{r-i} | < U_{r-i}
\]

(3)

where

\[
d = \begin{bmatrix} d_1 \\ \vdots \\ d_k \\ \vdots \\ d_{m-n} \\ d_{p-1} \\ \vdots \\ d_{p-n} \\ d_{T-1} \\ \vdots \\ d_{T-n} \end{bmatrix}
\]

The objective function \( Z \) in formulation 3 above is quadratic in the measurement adjustments \( d \), and the constraints have been linearized in terms of the adjustment variables \( d \). This is the form of a classical quadratic programming problem for which a variety of robust solution strategies exist. The solution of this
problem minimizes the property adjustments required to satisfy physical constraints within measurement system uncertainty bounds, and in a logical sense provides most plausible engine operating conditions subject to restrictions inherent in the linearization of the physical constraints.

The reconciler developed as part of this effort constructs the quadratic programming problem defined in formulation 3 above and implements the complementary pivot method algorithm [4] to obtain the QP problem solution. A hierarchy diagram describing the organization of routines in the reconciler is presented in Appendix A, Figure 5. Documentation describing the function of reconciler routines is given in Appendix C1, and a source code listing is presented in Appendix C2.

In order to perform a reconciliation analysis, four types of input data are required. Thermodynamic property data in operating ranges of interest are necessary. Specific enthalpy, specific entropy, and density as functions of pressure and temperature are required. For SSME analyses, hydrogen, oxygen, and water property information was provided and integrated into the reconciler logic by John Butas of NASA/MSFC/EP52. In addition, experimental data (or computational simulation results) are required to provide a baseline for adjustment calculations. The TTB Engine 3001 test program has provided extensive high quality experimental data for reconciliation analyses. PBM predictions have provided a simulation baseline for initial reconciler testing. The third type of input required for reconciliation analysis is uncertainty estimates quantifying model limitations (physical constraint bounds
L in formulation 3) as well as test data confidence bands (uncertainty bounds \( U \) in formulation 3). Finally, system definition information must be constructed to specify engine configuration and to properly associate nodal property data. A detailed description of input data requirements is provided in Appendix C1 documentation and a listing of the reconciler input data file format is provided in Appendix B, Table 2.

Reconciler performance has been verified on a number of test problems. Recently, reconciler logic was tested using results of a PBM simulation of the HPFTP system at 109% RPL to provide baseline measurements. A schematic of the HPFTP system with analysis nodes identified is displayed in Appendix A, Figure 6. The analysis configuration was composed of 14 nodes, 5 mass flow circuits, and 4 energy volumes. Mass circuit and energy volume definition nodes for this analysis are specified in Appendix B, Table 3. The energy volumes include the fuel preburner, high pressure fuel turbopump, fuel side turn around duct, and fuel side hot gas manifold. Coarse measurement system uncertainty estimates were utilized in the HPFTP test case analysis because more precise uncertainty information was unavailable. These estimates are provided in Appendix B, Table 4.

Mass, energy, entropy, and availability imbalances both before and after reconciliation are displayed in Appendix B, Tables 5 and 6 respectively. A significant reduction in subsystem energy imbalances after reconciliation is the most striking result observed in comparing Tables 5 and 6. A 99% energy imbalance reduction in the fuel preburner and turbopump subsystems was
obtained during the reconciliation process while mass balance and entropy production requirements were maintained. System properties both before and after reconciliation are presented in Appendix B, Table 7. The adjustments required to achieve reconciliation (solution to the quadratic programming problem outlined in formulation 3) are also presented in Table 7. Significant reductions in PBM predicted hot gas temperatures throughout the system are suggested. These temperature reductions remain within specified measurement uncertainty bounds, yet provide substantial improvement in the initially large energy imbalances.

Heuristic data integration procedures do not provide the level of confidence in prediction that is required in a long term engine development program. Efficient and reliable use of experimental observation to improve performance prediction and safety requires a systematic test data integration strategy. The reconciliation strategy outlined above is a logical procedure for improving rocket engine performance prediction. The mathematical foundation is well defined and computations are physically sound within approximation limits. In addition, the base procedure is completely general, with material and system configuration provided by modular data file inputs. Initial test results have been quite successful and strongly support continued development and use of this mathematical programming approach for large system test data reconciliation. This technique can also be utilized to evaluate test data integrity and isolate measurement system problems.
A list of recommendations based on results of this research effort is presented below.

1. Local modifications to the power balance model should be thoroughly investigated before implementation due to the high level of logic connectivity. If PBM is to be used extensively as a prediction tool, a catalog of parameter influence coefficients should be developed to efficiently assess the system wide impact of specific code changes.

2. Without extensive documentation describing imbedded empiricisms within PBM logic, the power balance model should be considered a high order gains model containing the experience base archive. PBM should not be considered a cornerstone theoretical prediction tool without major modification.

3. Development of mathematical programming approaches to test data reconciliation should continue in order to provide a consistent and logical basis for improving performance prediction, a platform for logically resolving data inconsistencies, and a means of assessing data and measurement system integrity.

4. A fundamentally sound theoretical model of engine performance should be developed.

5. Uncertainty analysis should be incorporated in any rocket engine performance evaluation and prediction program.

6. An integrated rocket engine performance prediction platform should be developed that modularizes fundamental theoretical computations and provides a standardized interface for
efficient parametric integration of test data.

7. Frictional resistance relations should be added to the quadratic reconciler in order to provide more consistent pressure loss relations.
5.0 REFERENCES


APPENDIX A

FIGURES
Figure 1. LPFP Speed
Figure 2. HPFP Disch Temp
Figure 3. CCV Flow Rate
Figure 4. HPFP Disch Pressure
Figure 5. Quadratic Reconciler Hierarchy Diagram

MAIN

------------------------FILEIO.DAT

------------------------RECON

| PDAT* | TDAT* | UDAT* | VDAT* |

------------------------CPIVOT

| ODAT* | ODATV* |

------------------------PROP

| PRPMIX |

------------------------PRPSAT

------------------------ITERP2

| ITERP1 |

* - indicates data file designated in FILEIO.DAT
Figure 6. HPFTP system with reconciliation analysis nodes indicated.

[Diagram showing the HPFTP system with labeled nodes and connections indicating flow and supply paths.]
APPENDIX B

TABLES
Table 1. TTB Engine 3001 test program matrix of control variable settings

<table>
<thead>
<tr>
<th>TEST PT</th>
<th>POWER LEVEL</th>
<th>MR</th>
<th>F7</th>
<th>LO2 NPSP</th>
<th>LH2 NPSP</th>
<th>GO2 REPRS</th>
<th>GH2 REPRS</th>
<th>TEST #</th>
<th>TIME SLICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100.00</td>
<td>5.85</td>
<td>1.70</td>
<td>20.00</td>
<td>6.00</td>
<td>1.10</td>
<td>0.20</td>
<td>TTB-25</td>
<td>109-119</td>
</tr>
<tr>
<td>2</td>
<td>104.00</td>
<td>5.85</td>
<td>1.70</td>
<td>20.00</td>
<td>24.50</td>
<td>2.40</td>
<td>1.20</td>
<td>TTB-25</td>
<td>134-139</td>
</tr>
<tr>
<td>3</td>
<td>100.00</td>
<td>5.85</td>
<td>1.70</td>
<td>120.00</td>
<td>24.50</td>
<td>2.40</td>
<td>0.20</td>
<td>TTB-25</td>
<td>30-35</td>
</tr>
<tr>
<td>4</td>
<td>104.00</td>
<td>5.85</td>
<td>1.70</td>
<td>120.00</td>
<td>6.00</td>
<td>1.10</td>
<td>1.20</td>
<td>TTB-25</td>
<td>49-59</td>
</tr>
<tr>
<td>5</td>
<td>100.00</td>
<td>5.85</td>
<td>1.90</td>
<td>20.00</td>
<td>24.50</td>
<td>1.10</td>
<td>1.20</td>
<td>TTB-28</td>
<td>114-119</td>
</tr>
<tr>
<td>6</td>
<td>104.00</td>
<td>5.85</td>
<td>1.90</td>
<td>20.00</td>
<td>6.00</td>
<td>2.40</td>
<td>0.20</td>
<td>TTB-28</td>
<td>84-89</td>
</tr>
<tr>
<td>7</td>
<td>100.00</td>
<td>5.85</td>
<td>1.90</td>
<td>120.00</td>
<td>6.00</td>
<td>2.40</td>
<td>1.20</td>
<td>TTB-28</td>
<td>30-35</td>
</tr>
<tr>
<td>8</td>
<td>104.00</td>
<td>5.85</td>
<td>1.90</td>
<td>120.00</td>
<td>24.50</td>
<td>1.10</td>
<td>0.20</td>
<td>TTB-27</td>
<td>36-41</td>
</tr>
<tr>
<td>9</td>
<td>100.00</td>
<td>6.15</td>
<td>1.70</td>
<td>20.00</td>
<td>24.50</td>
<td>2.40</td>
<td>1.20</td>
<td>TTB-25</td>
<td>154-159</td>
</tr>
<tr>
<td>10</td>
<td>104.00</td>
<td>6.15</td>
<td>1.70</td>
<td>20.00</td>
<td>6.00</td>
<td>1.10</td>
<td>0.20</td>
<td>TTB-25</td>
<td>194-199</td>
</tr>
<tr>
<td>11</td>
<td>100.00</td>
<td>6.15</td>
<td>1.70</td>
<td>120.00</td>
<td>6.00</td>
<td>1.10</td>
<td>1.20</td>
<td>TTB-23</td>
<td>69-74</td>
</tr>
<tr>
<td>12</td>
<td>104.00</td>
<td>6.15</td>
<td>1.70</td>
<td>120.00</td>
<td>24.50</td>
<td>2.40</td>
<td>0.20</td>
<td>TTB-23</td>
<td>40-45</td>
</tr>
<tr>
<td>13</td>
<td>100.00</td>
<td>6.15</td>
<td>1.90</td>
<td>20.00</td>
<td>6.00</td>
<td>2.40</td>
<td>0.20</td>
<td>TTB-26</td>
<td>118-122</td>
</tr>
<tr>
<td>14</td>
<td>104.00</td>
<td>6.15</td>
<td>1.90</td>
<td>20.00</td>
<td>24.50</td>
<td>1.10</td>
<td>1.20</td>
<td>TTB-26</td>
<td>144-154</td>
</tr>
<tr>
<td>15</td>
<td>100.00</td>
<td>6.15</td>
<td>1.90</td>
<td>120.00</td>
<td>24.50</td>
<td>1.10</td>
<td>0.20</td>
<td>TTB-26</td>
<td>30-35</td>
</tr>
<tr>
<td>16</td>
<td>104.00</td>
<td>6.15</td>
<td>1.90</td>
<td>120.00</td>
<td>6.00</td>
<td>2.40</td>
<td>1.20</td>
<td>TTB-26</td>
<td>59-64</td>
</tr>
</tbody>
</table>

**DEFINITIONS**

- **Power Level**: percent of SSME rated power level
- **MR**: mixture ratio (mass O2)/(mass H2)
- **F7**: F7 orifice area (square inches)
- **LOX NPSP**: liquid oxygen net positive suction pressure (psi)
- **Fuel NPSP**: liquid hydrogen net positive suction pressure (psi)
- **LOX REPRS**: oxygen repressurization flow rate (lb/s)
- **Fuel REPRS**: hydrogen repressurization flow rate (lb/s)

**Figure Designation**

- % RPL
- M/R
- F7
- OSP
- HSP
- ORP
- HRP
Table 2. Reconciler input data file formats

FILEIO.DAT – designates I/O data filenames

'input property data filename'
'input test data filename'
'input uncertainty estimates filename'
'input volume definition data filename'
'output (standard format) filename'
'output (test data input format) filename'

TDAT = 'input test data filename'

NDESC NTTB
DESC(1) ......... DESC(NDESC)
TTB(1) .......... TTB(NTTB)

UDAT = 'input uncertainty estimates filename'

IPRPD ITTBD IROWA JCOLA ITORDX ITORDY DPF DTF
UP(1) .......... UP(NTNOD)
UT(1) .......... UT(NTNOD)
UW(1) .......... UW(NTNOD)
UWMFC(1) ....... UWMFC(NTMFC)
UEVOL(1) ....... UEVOL(NTVOL)
USVOL(1) ....... USVOL(NTVOL)

(continued next page)
**Table 2. Reconciler input data file formats**

VDAT = 'input volume definition data filename'

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IENV</td>
<td>IPCTTH</td>
<td>MAXSTG</td>
<td>NHG</td>
<td>NTMFC</td>
<td>NTNOD</td>
<td>NTVol</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IA(1)</td>
<td>...</td>
<td>IA(NTNOD)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IP(1)</td>
<td>...</td>
<td>IP(NTNOD)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IT(1)</td>
<td>...</td>
<td>IT(NTNOD)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IW(1)</td>
<td>...</td>
<td>IW(NTNOD)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAT(1)</td>
<td>...</td>
<td>MAT(NTNOD)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIO(1)</td>
<td>...</td>
<td>MIO(NTMFC)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NIO(1)</td>
<td>...</td>
<td>NIO(NTVol)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODIR(1,1)</td>
<td>...</td>
<td>MODIR(1,MIO(1))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IMFCN(1,1)</td>
<td>...</td>
<td>IMFCN(1,NTMFC)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MODIR(NTMFC,1)</td>
<td>...</td>
<td>MODIR(NTMFC,MIO(NTMFC))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IMFCN(NTMFC,1)</td>
<td>...</td>
<td>IMFCN(NTMFC,MIO(NTMFC))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IODIR(1,1)</td>
<td>...</td>
<td>IODIR(1,NIO(1))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IVOLN(1,1)</td>
<td>...</td>
<td>IVOLN(1,NTVol)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IODIR(NTVol,1)</td>
<td>...</td>
<td>IODIR(NTVol,NIO(NTVol))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IVOLN(NTVol,1)</td>
<td>...</td>
<td>IVOLN(NTVol,NIO(NTVol))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NODHG(1)</td>
<td>...</td>
<td>NODHG(NHG)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NH2HG(1)</td>
<td>...</td>
<td>NH2HG(NHG)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO2HG(1)</td>
<td>...</td>
<td>NO2HG(NHG)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICEFF(1)</td>
<td>...</td>
<td>ICEFF(NHG)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IH2HG(1,1)</td>
<td>...</td>
<td>IH2HG(1,NH2HG(1))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IO2HG(1,1)</td>
<td>...</td>
<td>IO2HG(1,NO2HG(1))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IH2HG(NHG,1)</td>
<td>...</td>
<td>IH2HG(NHG,NH2HG(NHG))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IO2HG(NHG,1)</td>
<td>...</td>
<td>IO2HG(NHG,NO2HG(NHG))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 3. HPFTP analysis number 1 circuit definitions

<table>
<thead>
<tr>
<th>Energy Volume Definitions</th>
<th>Mass Circuit Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume #</td>
<td>Boundary Nodes</td>
</tr>
<tr>
<td>1</td>
<td>1 2 3 4</td>
</tr>
<tr>
<td>2</td>
<td>4 5 6 7</td>
</tr>
<tr>
<td>3</td>
<td>6 7 8 9</td>
</tr>
<tr>
<td>4</td>
<td>9 10 11 12</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 4. Uncertainty estimates for HPFTP analysis number 1

### Nodal Property Uncertainty

<table>
<thead>
<tr>
<th>Node #</th>
<th>Pressure (psi)</th>
<th>Temperature (degF)</th>
<th>Flow Rate (lb/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.772</td>
<td>0.0208</td>
<td>0.00796</td>
</tr>
<tr>
<td>2</td>
<td>0.605</td>
<td>0.1</td>
<td>0.00831</td>
</tr>
<tr>
<td>3</td>
<td>0.659</td>
<td>0.00972</td>
<td>0.0001</td>
</tr>
<tr>
<td>4</td>
<td>200.0</td>
<td>100.0</td>
<td>1.6</td>
</tr>
<tr>
<td>5</td>
<td>0.0272</td>
<td>0.00424</td>
<td>0.0162</td>
</tr>
<tr>
<td>6</td>
<td>200.0</td>
<td>20.0</td>
<td>1.7</td>
</tr>
<tr>
<td>7</td>
<td>200.0</td>
<td>100.0</td>
<td>1.6</td>
</tr>
<tr>
<td>8</td>
<td>200.0</td>
<td>20.0</td>
<td>1.6</td>
</tr>
<tr>
<td>9</td>
<td>200.0</td>
<td>100.0</td>
<td>1.7</td>
</tr>
<tr>
<td>10</td>
<td>200.0</td>
<td>100.0</td>
<td>1.7</td>
</tr>
<tr>
<td>11</td>
<td>0.359</td>
<td>0.0436</td>
<td>0.00175</td>
</tr>
<tr>
<td>12</td>
<td>200.0</td>
<td>200.0</td>
<td>0.00175</td>
</tr>
<tr>
<td>13</td>
<td>200.0</td>
<td>20.0</td>
<td>0.14</td>
</tr>
<tr>
<td>14</td>
<td>200.0</td>
<td>20.0</td>
<td>0.3</td>
</tr>
</tbody>
</table>

### Mass Imbalance Limits

<table>
<thead>
<tr>
<th>Circuit #</th>
<th>Flow Imbalance Limit (lb/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.001</td>
</tr>
<tr>
<td>2</td>
<td>0.001</td>
</tr>
<tr>
<td>3</td>
<td>0.001</td>
</tr>
<tr>
<td>4</td>
<td>0.001</td>
</tr>
<tr>
<td>5</td>
<td>0.001</td>
</tr>
</tbody>
</table>

### Volume Imbalance Limits

<table>
<thead>
<tr>
<th>Volume #</th>
<th>Energy Imbalance Limit (Btu/sec)</th>
<th>Entropy Imbalance Limit (Btu/degR-sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>71.0</td>
<td>100.0</td>
</tr>
<tr>
<td>2</td>
<td>43.0</td>
<td>100.0</td>
</tr>
<tr>
<td>3</td>
<td>2.6</td>
<td>100.0</td>
</tr>
<tr>
<td>4</td>
<td>18.0</td>
<td>100.0</td>
</tr>
</tbody>
</table>
Table 5. High pressure fuel turbopump system imbalances at 109% RPL prior to reconciliation

**HP_FUEL_SIDE_ANALYSIS_1**

**THRUST = 109.0% RPL**

<table>
<thead>
<tr>
<th>SUBSYSTEM</th>
<th>DW(LB/S)</th>
<th>DE(BTU/S)</th>
<th>DE(HP)</th>
<th>DS(BTU/R-S)</th>
<th>DAV(BTU/S)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPB_SUBSYSTEM</td>
<td>.014</td>
<td>-7111.634</td>
<td>-10061.800</td>
<td>-664.165</td>
<td>344895.60</td>
</tr>
<tr>
<td>HPFTP_SUBSYSTEM</td>
<td>.000</td>
<td>-4277.324</td>
<td>-6051.713</td>
<td>-174.517</td>
<td>88216.79</td>
</tr>
<tr>
<td>TAD_SUBSYSTEM</td>
<td>.100</td>
<td>-2563.647</td>
<td>-3627.140</td>
<td>-12.175</td>
<td>3889.41</td>
</tr>
<tr>
<td>HGM_SUBSYSTEM</td>
<td>.000</td>
<td>183.112</td>
<td>259.074</td>
<td>-5.406</td>
<td>3048.38</td>
</tr>
</tbody>
</table>

Table 6. High pressure fuel turbopump system imbalances at 109% RPL after reconciliation

**HP_FUEL_SIDE_ANALYSIS_1**

**THRUST = 109.0% RPL**

<table>
<thead>
<tr>
<th>SUBSYSTEM</th>
<th>DW(LB/S)</th>
<th>DE(BTU/S)</th>
<th>DE(HP)</th>
<th>DS(BTU/R-S)</th>
<th>DAV(BTU/S)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPB_SUBSYSTEM</td>
<td>.002</td>
<td>-59.775</td>
<td>-84.571</td>
<td>-660.372</td>
<td>349937.30</td>
</tr>
<tr>
<td>HPFTP_SUBSYSTEM</td>
<td>-.003</td>
<td>-17.536</td>
<td>-24.811</td>
<td>-165.596</td>
<td>87748.05</td>
</tr>
<tr>
<td>TAD_SUBSYSTEM</td>
<td>.004</td>
<td>-58.564</td>
<td>-82.858</td>
<td>-6.975</td>
<td>3638.13</td>
</tr>
<tr>
<td>HGM_SUBSYSTEM</td>
<td>.024</td>
<td>-227.164</td>
<td>-321.400</td>
<td>-.467</td>
<td>20.30</td>
</tr>
</tbody>
</table>
Table 7. HPFTP System Reconciliation at 109% RPL

**HPFTP ANALYSIS NO 1**

**ORIGINAL NODE DATA**

<table>
<thead>
<tr>
<th>NODE</th>
<th>PRESSURE</th>
<th>TEMPERATURE</th>
<th>FLOWRATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7723.00</td>
<td>208.30</td>
<td>79.63</td>
</tr>
<tr>
<td>2</td>
<td>6049.00</td>
<td>278.40</td>
<td>83.07</td>
</tr>
<tr>
<td>3</td>
<td>6585.00</td>
<td>97.16</td>
<td>1.01</td>
</tr>
<tr>
<td>4</td>
<td>5498.00</td>
<td>1929.00</td>
<td>163.70</td>
</tr>
<tr>
<td>5</td>
<td>271.70</td>
<td>42.40</td>
<td>162.30</td>
</tr>
<tr>
<td>6</td>
<td>6739.00</td>
<td>96.71</td>
<td>159.20</td>
</tr>
<tr>
<td>7</td>
<td>3718.00</td>
<td>1737.00</td>
<td>166.80</td>
</tr>
<tr>
<td>8</td>
<td>6739.00</td>
<td>96.71</td>
<td>157.80</td>
</tr>
<tr>
<td>9</td>
<td>3691.00</td>
<td>1720.00</td>
<td>168.10</td>
</tr>
<tr>
<td>10</td>
<td>3585.00</td>
<td>1715.00</td>
<td>168.10</td>
</tr>
<tr>
<td>11</td>
<td>3587.00</td>
<td>436.30</td>
<td>17.51</td>
</tr>
<tr>
<td>12</td>
<td>3574.00</td>
<td>460.90</td>
<td>17.51</td>
</tr>
<tr>
<td>13</td>
<td>6739.00</td>
<td>96.71</td>
<td>1.38</td>
</tr>
<tr>
<td>14</td>
<td>6739.00</td>
<td>96.71</td>
<td>3.05</td>
</tr>
</tbody>
</table>

**RECONCILED NODE DATA**

<table>
<thead>
<tr>
<th>NODE</th>
<th>PRESSURE</th>
<th>TEMPERATURE</th>
<th>FLOWRATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7723.00</td>
<td>208.30</td>
<td>79.63</td>
</tr>
<tr>
<td>2</td>
<td>6049.00</td>
<td>278.40</td>
<td>83.07</td>
</tr>
<tr>
<td>3</td>
<td>6585.00</td>
<td>97.16</td>
<td>1.01</td>
</tr>
<tr>
<td>4</td>
<td>5511.55</td>
<td>1907.02</td>
<td>163.71</td>
</tr>
<tr>
<td>5</td>
<td>271.70</td>
<td>42.40</td>
<td>162.30</td>
</tr>
<tr>
<td>6</td>
<td>6719.12</td>
<td>95.07</td>
<td>159.25</td>
</tr>
<tr>
<td>7</td>
<td>3704.84</td>
<td>1704.95</td>
<td>166.76</td>
</tr>
<tr>
<td>8</td>
<td>6723.15</td>
<td>93.87</td>
<td>157.87</td>
</tr>
<tr>
<td>9</td>
<td>3657.60</td>
<td>1682.59</td>
<td>168.14</td>
</tr>
<tr>
<td>10</td>
<td>3569.42</td>
<td>1687.03</td>
<td>168.12</td>
</tr>
<tr>
<td>11</td>
<td>3587.00</td>
<td>436.30</td>
<td>17.51</td>
</tr>
<tr>
<td>12</td>
<td>3543.89</td>
<td>419.48</td>
<td>17.51</td>
</tr>
<tr>
<td>13</td>
<td>6722.71</td>
<td>96.53</td>
<td>1.38</td>
</tr>
<tr>
<td>14</td>
<td>6722.71</td>
<td>96.53</td>
<td>3.05</td>
</tr>
</tbody>
</table>

(continued next page)
Table 7. HPFTP System Reconciliation at 109% RPL

HPFTP ANALYSIS NO 1

BALANCING ADJUSTMENTS

<table>
<thead>
<tr>
<th>NODE</th>
<th>PRESSURE</th>
<th>TEMPERATURE</th>
<th>FLOWRATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>2</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>3</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>4</td>
<td>13.55</td>
<td>-21.98</td>
<td>.01</td>
</tr>
<tr>
<td>5</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>6</td>
<td>-19.88</td>
<td>-1.64</td>
<td>.05</td>
</tr>
<tr>
<td>7</td>
<td>-13.16</td>
<td>-32.05</td>
<td>-.04</td>
</tr>
<tr>
<td>8</td>
<td>-15.85</td>
<td>-2.84</td>
<td>.07</td>
</tr>
<tr>
<td>9</td>
<td>-33.40</td>
<td>-37.41</td>
<td>.04</td>
</tr>
<tr>
<td>10</td>
<td>-15.58</td>
<td>-27.97</td>
<td>.02</td>
</tr>
<tr>
<td>11</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>12</td>
<td>-30.11</td>
<td>-41.42</td>
<td>.00</td>
</tr>
<tr>
<td>13</td>
<td>-16.29</td>
<td>-.18</td>
<td>.00</td>
</tr>
<tr>
<td>14</td>
<td>-16.29</td>
<td>-.18</td>
<td>.00</td>
</tr>
</tbody>
</table>

PERCENT ADJUSTMENT

<table>
<thead>
<tr>
<th>NODE</th>
<th>PRESSURE</th>
<th>TEMPERATURE</th>
<th>FLOWRATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>2</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>3</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>4</td>
<td>.25</td>
<td>-1.14</td>
<td>.01</td>
</tr>
<tr>
<td>5</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>6</td>
<td>-.30</td>
<td>-1.69</td>
<td>.03</td>
</tr>
<tr>
<td>7</td>
<td>-.35</td>
<td>-1.84</td>
<td>-.02</td>
</tr>
<tr>
<td>8</td>
<td>-.24</td>
<td>-2.94</td>
<td>.04</td>
</tr>
<tr>
<td>9</td>
<td>-.90</td>
<td>-2.18</td>
<td>.02</td>
</tr>
<tr>
<td>10</td>
<td>-.43</td>
<td>-1.63</td>
<td>.01</td>
</tr>
<tr>
<td>11</td>
<td>.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>12</td>
<td>-.84</td>
<td>-8.99</td>
<td>.00</td>
</tr>
<tr>
<td>13</td>
<td>-.24</td>
<td>-.18</td>
<td>-.01</td>
</tr>
<tr>
<td>14</td>
<td>-.24</td>
<td>-.18</td>
<td>.07</td>
</tr>
</tbody>
</table>
APPENDIX C1

RECONCILER DOCUMENTATION
Routine: MAIN  (main program for reconciliation model)

Routine Function:

Main calling routine for reconciliation model. Opens and reads input data files associated with thermodynamic property data, control volume definition, test data, and uncertainty estimates. Initializaes parameters for reconciliation model.

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters
TDAT - test data
UDAT - uncertainty estimates
VDAT - control volume definition information
H2PRP - hydrogen table pressures and temperatures, and associated enthalpies, entropies, and densities
O2PRP - oxygen table pressures and temperatures, and associated enthalpies, entropies, and densities
H2OPRP- water table pressures and temperatures, and associated enthalpies, entropies, and densities
TABLE - number of distinct pressures and temperatures in each hydrogen, oxygen, and water table
STD - reference state enthalpies entropies, and absolute entropies for hydrogen, oxygen, and water

Input Variable Definitions:

Input File 'FILEIO.DAT' (identifies input files)
PDAT - alpha variable that identifies name of property data input file
TDAT - alpha variable that identifies name of test data input file
UDAT - alpha variable that identifies name of uncertainty data input file
VDAT - alpha variable that identifies name of volume definition input file
ODAT - alpha variable that identifies name of main reconciler output file
ODATV - alpha variable that identifies name of reconciler output file that is in same format as variable TDAT input file

Input File PDAT (property data input file)
(file name variable identified in FILEIO.DAT)
PTITLE - property data table subsection title
NH2P(I) - number of pressures in H2 property data table I
NH2T(I) - number of temperatures in H2 property data table I
Routine MAIN

H2P1(I) - pressure I in H2 data table 1
H2T1(I) - temperature I in H2 data table 1
H2H1(I,J) - H2 enthalpy associated with pressure I and temperature J in H2 data table 1
H2S1(I,J) - H2 entropy associated with pressure I and temperature J in H2 data table 1
H2D1(I,J) - H2 density associated with pressure I and temperature J in H2 data table 1
H2P2(I) - pressure I in H2 data table 2
H2T2(I) - temperature I in H2 data table 2
H2H2(I,J) - H2 enthalpy associated with pressure I and temperature J in H2 data table 2
H2S2(I,J) - H2 entropy associated with pressure I and temperature J in H2 data table 2
H2D2(I,J) - H2 density associated with pressure I and temperature J in H2 data table 2
H2P3(I) - pressure I in H2 data table 3
H2T3(I) - temperature I in H2 data table 3
H2H3(I,J) - H2 enthalpy associated with pressure I and temperature J in H2 data table 3
H2S3(I,J) - H2 entropy associated with pressure I and temperature J in H2 data table 3
H2D3(I,J) - H2 density associated with pressure I and temperature J in H2 data table 3
H2P4(I) - pressure I in H2 data table 4
H2T4(I) - temperature I in H2 data table 4
H2H4(I,J) - H2 enthalpy associated with pressure I and temperature J in H2 data table 4
H2S4(I,J) - H2 entropy associated with pressure I and temperature J in H2 data table 4
H2D4(I,J) - H2 density associated with pressure I and temperature J in H2 data table 4
NO2P(I) - number of pressures in O2 property data table I
NO2T(I) - number of temperatures in O2 property data table I
O2P1(I) - pressure I in O2 data table 1
O2T1(I) - temperature I in O2 data table 1
O2H1(I,J) - O2 enthalpy associated with pressure I and temperature J in O2 data table 1
O2S1(I,J) - O2 entropy associated with pressure I and temperature J in O2 data table 1
O2D1(I,J) - O2 density associated with pressure I and temperature J in O2 data table 1
O2P2(I) - pressure I in O2 data table 2
O2T2(I) - temperature I in O2 data table 2
O2H2(I,J) - O2 enthalpy associated with pressure I and temperature J in O2 data table 2
O2S2(I,J) - O2 entropy associated with pressure I and temperature J in O2 data table 2
O2D2(I,J) - O2 density associated with pressure I and temperature J in O2 data table 2
Routine MAIN (page 3)

O2P3(I) - pressure I in O2 data table 3
O2T3(I) - temperature I in O2 data table 3
O2H3(I,J) - O2 enthalpy associated with pressure I and temperature J in O2 data table 3
O2S3(I,J) - O2 entropy associated with pressure I and temperature J in O2 data table 3
O2D3(I,J) - O2 density associated with pressure I and temperature J in O2 data table 3
NH2OP(I) - number of pressures in H2O property data table I
NH2OT(I) - number of temperatures in H2O property data table I
H2OP1(I) - pressure I in H2O data table 1
H2OT1(I) - temperature I in H2O data table 1
H2OH1(I,J) - H2O enthalpy associated with pressure I and temperature J in H2O data table 1
H2OS1(I,J) - H2O entropy associated with pressure I and temperature J in H2O data table 1
H2OD1(I,J) - H2O density associated with pressure I and temperature J in H2O data table 1

Input File TDAT (test data input file)
NDESC - number of alpha variable test data descriptions
NTTB - number of data entries in test data table
DESC(I) - alpha variable test data description I
TTB(I) - test data entry I

Input File VDAT (volume definition input file)
IENV - TTB array address of environmental temperature
IPCTTH - TTB array address of % rated power level
MAXSTG - number of stages to be used in SQP iteration sequence
NHG - number of system nodes at which HG flow occurs
NTMFC - number of mass flow circuits in engine system analysis
NTNOD - number of nodes in engine system analysis
NTVOL - number of volumes in engine system analysis
IA(I) - position in TTB array containing the value of the cross-sectional area at node I
IP(I) - position in TTB array containing the value of the pressure at node I
IT(I) - position in TTB array containing the value of the temperature at node I
IW(I) - position in TTB array containing the value of the mass flow rate at node I
MAT(I) - material identifying number at node I
1 = H2, 2 = O2, 3 = hot gas
MIO(I) - number of I/O's associated with mass flow circuit I
Routine MAIN (page 4)

NIO(I) - number of I/O's associated with volume I
MODIR(I,J) - flow direction of I/O J in mass flow circuit I
  1 = inlet flow, -1 = outlet flow
IMFC(I,J) - node number of I/O J in mass flow circuit I
IODIR(I,J) - flow direction of I/O J in volume I
  1 = inlet flow, -1 = outlet flow
IVOLN(I,J) - node number of I/O J in volume I
NODHG(I) - node number of hot gas flow I
NH2HG(I) - number of H2 flows entering hot gas flow I
NO2HG(I) - number of O2 flows entering hot gas flow I
ICEFF(I) - position in TTB array containing the combustion
  efficiency of hot gas flow I
IH2HG(I,J) - node number of H2 feed J to hot gas flow I
IO2HG(I,J) - node number of O2 feed J to hot gas flow I

Input File UDAT (uncertainty estimates data input file)
(file name variable identified in FILEIO.DAT)

IPRPD - unused in this version
ITTBD - unused in this version
IROWA - unused in this version
JCOLA - unused in this version
ITORDX - unused in this version
ITORDY - unused in this version
DPF - pressure fractional increment used in finite
difference approximation of partial derivatives
  with respect to pressure
DTF - temperature fractional increment used in finite
difference approximation of partial derivatives
  with respect to temperature
UP(I) - pressure uncertainty at node I
UT(I) - temperature uncertainty at node I
UW(I) - mass flow uncertainty at node I
UWMFC(I) - mass flow uncertainty associated with mass flow
  circuit I
UEVOL(I) - energy uncertainty associated with volume I
USVOL(I) - entropy uncertainty associated with volume I

Output Variable Definitions:

No output variables

Subroutine Calls:

RECON

Calling Routines:

None
Routine: RECON  (reconciliation model construction and routing)

Routine Function:

Constructs a sequential quadratic programming (SQP) problem whose solution is the optimum reconciliation of uncertain test data and limited theoretical predictions for pressure, temperature and mass flow at specified node locations within the SSME flow network. Routes SQP solution logic. Outputs solution of SQP problem.

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters
TDAT - test data
UDAT - uncertainty estimates
VDAT - control volume definition information
H2PRP - hydrogen table pressures and temperatures, and associated enthalpies, entropies, and densities
O2PRP - oxygen table pressures and temperatures, and associated enthalpies, entropies, and densities
H2OPRP - water table pressures and temperatures, and associated enthalpies, entropies, and densities
TABLE - number of distinct pressures and temperatures in each hydrogen, oxygen, and water table
STD - reference state enthalpies entropies, and absolute entropies for hydrogen, oxygen, and water

Input Variable Definitions:

Common block inputs

Output Variable Definitions:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>node number</td>
</tr>
<tr>
<td>TTB(IP(I))</td>
<td>original test pressure at node I</td>
</tr>
<tr>
<td>TTB(IT(I))</td>
<td>original test temperature at node I</td>
</tr>
<tr>
<td>TTB(IW(I))</td>
<td>original test mass flow rate at node I</td>
</tr>
<tr>
<td>PREC</td>
<td>reconciled pressure at node I</td>
</tr>
<tr>
<td>TREC</td>
<td>reconciled temperature at node I</td>
</tr>
<tr>
<td>WREC</td>
<td>reconciled mass flow rate at node I</td>
</tr>
<tr>
<td>PADJ</td>
<td>pressure adjustment made at node I</td>
</tr>
<tr>
<td>TADJ</td>
<td>temperature adjustment made at node I</td>
</tr>
<tr>
<td>WADJ</td>
<td>mass flow rate adjustment made at node I</td>
</tr>
<tr>
<td>PPCT</td>
<td>percentage pressure adjustment made at node I</td>
</tr>
<tr>
<td>TPCT</td>
<td>percentage temperature adjustment made at node I</td>
</tr>
<tr>
<td>WPCT</td>
<td>percentage mass flow rate adjustment made at node I</td>
</tr>
</tbody>
</table>
Routine RECON (page 2)

Subroutine Calls:

PROP
CPIVOT

Calling Routines:

MAIN
Routine: CPIVOT  (solver routing routine)

Routine Function:

The main routing routine for the complementary pivot method, quadratic programming problem solver.

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:

Common block inputs
N - dimension of square (NxN) main solver array
N = 6*number of nodes + 3*number of volumes + 2*number of mass flow circuits

Output Variable Definitions:

Common block outputs

Subroutine Calls:

MATRIX
INITIA
NEWBAS
SORT

Calling Routines:

RECON
Routine: MATRIX

Routine Function:

Initializes solver inputs

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:

Common block inputs

N - dimension of square (NxN) main solver array

N = 6*number of nodes + 3*number of volumes + 2*number of mass flow circuits

Output Variable Definitions:

Common block outputs

Subroutine Calls:

None

Calling Routines:

CPIVOT
Routine: INITIA

Routine Function:

Determines the initial almost complementary solution in the complementary pivot method solution strategy

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:

Common block inputs
N - dimension of square (NxN) main solver array
  N = 6*number of nodes + 3*number of volumes + 2*number of mass flow circuits

Output Variable Definitions:

Common block outputs

Subroutine Calls:

None

Calling Routines:

CPIVOT
Routine: NEWBAS

Routine Function:

Finds the new basis column to enter in terms of the current basis in the complementary pivot method solver

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:

Common block inputs
N - dimension of square (NxN) main solver array
N = 6*number of nodes + 3*number of volumes + 2*number of mass flow circuits

Output Variable Definitions:

Common block outputs

Subroutine Calls:

SOLVE

Calling Routines:

CPIVOT
Routine: SORT

Routine Function:

Finds the pivot row for the next iteration by the use of (simplex-type) minimum ratio rule as part of the complementary pivot method solver

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:

Common block inputs

N - dimension of square (N x N) main solver array
N = 6 * number of nodes + 3 * number of volumes + 2 * number of mass flow circuits

Output Variable Definitions:

Common block outputs

Subroutine Calls:

None

Calling Routines:

CPIVOT
Routine: PIVOT

Routine Function:
Performs the pivot operation by updating the inverse of the basis and the Q vector as part of the complementary pivot method solver.

Common Blocks:
Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:
Common block inputs
N - dimension of square (NxN) main solver array
N = 6*number of nodes + 3*number of volumes + 2*number of mass flow circuits

Output Variable Definitions:
Common block outputs

Subroutine Calls:
None

Calling Routines:
CPIVOT
Routine: SOLVE

Routine Function:

Correlates quadratic programming problem solution as part of the complementary pivot method solver

Common Blocks:

Blank - quadratic programming algorithm matrices and parameters

Input Variable Definitions:

Common block inputs
N - dimension of square (NxN) main solver array
N = 6*number of nodes + 3*number of volumes + 2*number of mass flow circuits

Output Variable Definitions:

Common block outputs

Subroutine Calls:

None

Calling Routines:

NEWBAS
Routine: PROP

Routine Function:
Calculates hydrogen, oxygen, and hot gas properties

Common Blocks:

H2PRP - hydrogen table pressures and temperatures, and associated enthalpies, entropies, and densities
O2PRP - oxygen table pressures and temperatures, and associated enthalpies, entropies, and densities
H2OPRP - water table pressures and temperatures, and associated enthalpies, entropies, and densities
TABLE - number of distinct pressures and temperatures in each hydrogen, oxygen, and water table

Input Variable Definitions:

Common block inputs
MAT - material type
  1 = H2, 2 = O2, 3 = hot gas
PRSI - pressure
TMPI - temperature
OF - O2/H2 mass ratio
CEFF - combustion efficiency

Output Variable Definitions:

Common block outputs
ZENTH - table enthalpy
ZENTR - table entropy
ZDENS - density

Subroutine Calls:

PRPSAT
PRPMIX
ITERP2

Calling Routines:

RECON
Routine: PRPMIX

Routine Function:
Calculates hot gas mixture thermodynamic properties using a Dalton model

Common Blocks:
- H2PRP - hydrogen table pressures and temperatures, and associated enthalpies, entropies, and densities
- O2PRP - oxygen table pressures and temperatures, and associated enthalpies, entropies, and densities
- H2OPRP - water table pressures and temperatures, and associated enthalpies, entropies, and densities
- TABLE - number of distinct pressures and temperatures in each hydrogen, oxygen, and water table
- STD - reference state enthalpies entropies, and absolute entropies for hydrogen, oxygen, and water

Input Variable Definitions:
Common block inputs
- P - mixture pressure
- TMPI - mixture temperature
- OF - O2/H2 mass ratio
- CEFF - combustion efficiency

Output Variable Definitions:
Common block outputs
- HMIX - mixture enthalpy
- SMIX - mixture entropy
- PH2 - error "out of range" hydrogen pressure
- PH20 - error "out of range" water pressure
- PO2 - error "out of range" oxygen pressure
- TMPI - error "out of range" temperature

Subroutine Calls:
- ITERP2

Calling Routines:
- PROP
Routine: PRPSAT

Routine Function:

Calculates thermodynamic properties near saturation curve

Common Blocks:

H2PRP - hydrogen table pressures and temperatures, and associated enthalpies, entropies, and densities
O2PRP - oxygen table pressures and temperatures, and associated enthalpies, entropies, and densities
H2OPRP - water table pressures and temperatures, and associated enthalpies, entropies, and densities
TABLE - number of distinct pressures and temperatures in each hydrogen, oxygen, and water table
STD - reference state enthalpies entropies, and absolute entropies for hydrogen, oxygen, and water

Input Variable Definitions:

Common block inputs
X - pressure
Y - temperature
TCRT - table critical temperature
NX1 - number of pressure values in table
NY1 - number of temperature values in table
NX2 -
YL - table low temperature
YH - table high temperature
PRS1 - pressure table values
TMP1 - temperature table values
PROP - thermodynamic property table values
PRS2 - saturation pressure table
TMP2 - saturation temperature table
PROPL - saturated liquid property value
PROPV - saturated vapor property value

Output Variable Definitions:

Common block outputs
FPROP - calculated thermodynamic property

Subroutine Calls:

ITERP2
ITERP1
Calling Routines:

PROP
APPENDIX C2

RECONCILER SOURCE CODE
VERSION RV2-0610
PROGRAM RECONV2

CHARACTER*24 DESC
CHARACTER*12 PDAT, TDAT, UDAT, VDAT, ODAT, ODATV

COMMON CPM(200, 200), CPQ(200), L1CP, CPB(200, 200), NL1CP, NL2CP,
1 CPA(200), NE1CP, NE2CP, IRCP, MBASIS(300),
2 CPW(200), CPZ(200)

COMMON /VDAT/ IENV, INCTTH, MAXSTG, NHG, NTMFC, NTMOD, NTVOL,
1 IA(20), IP(20), IT(20), IN(20), MAT(20),
2 MIQ(5), MODIR(5,20), IMFCN(5,20),
3 NIO(5), IODIR(5,20), IVOLN(5,20),
4 NH2HG(5), NO2HG(5), NODHG(5), ICEFF(5),
5 IH2HG(5,5), IO2HG(5,5)

COMMON /TDAT/ TTB(100), NDESC, NTNB, DESC(5)

COMMON /UDAT/ IPRPD, ITTBD, IROWA, JCOLA, ITORDX, ITORDY,
1 DPF, DTF, UP(20), UT(20), UW(20),
2 UEVOL(5), USVOL(5), UWMC(5)

COMMON /H2PRP/
* H2P1(15), H2T1(11), H2H1(15,11), H2S1(15,11), H2D1(15,11),
* H2P2(20), H2T2(11), H2H2(20,11), H2S2(20,11), H2D2(20,11),
* H2P3(29), H2T3(25), H2H3(29,25), H2S3(29,25), H2D3(29,25),
COMMON /O2PRP/
* O2P1(13), O2T1(16), O2H1(13,16), O2S1(13,16), O2D1(13,16),
* O2P2(13), O2T2(17), O2H2(13,17), O2S2(13,17), O2D2(13,17),
* O2P3(5), O2T3(61), O2H3(5,61), O2S3(5,61), O2D3(5,61)
COMMON /H2OPRP/
* H2OP1(7), H2OT1(13), H2OH1(7,13), H2OS1(7,13), H2OD1(7,13)

COMMON /TABLE/
* NH2P(4), NH2T(4), NO2P(3), NO2T(3), NH2OP(1), NH2OT(1)
COMMON /STD/
* H2REF, H2ROF, SHAREF, H2REF, SO2REF, SHAREF, S2A, S2A,
* SWAA

DIMENSION *
* NH2PA(4), NH2TA(4), NO2PA(3), NO2TA(3), NH2OPA(1), NH2OTA(1)

CHARACTER*70 PTITLE

DATA (NH2PA(I), I=1,4)/15, 20, 29, 23/
DATA (NH2TA(J), J=1,4)/11, 11, 25, 25/
DATA (NO2PA(I), I=1,3)/13, 13, 5/
DATA (NO2TA(J), J=1,3)/16, 17, 61/
DATA (NH2OPA(I), I=1,1)/7/
DATA (NH2OTA(J), J=1,1)/13/

DO 90 I=1,4
90 NH2P(I)=NH2PA(I)
DO 91 I=1,3
90 NO2P(I)=NO2PA(I)
91 NO2T(I)=NO2TA(I)
90 NH2OP(1)=NH2OPA(1)
OPEN ( 7, FILE = 'FILEIO.DAT', STATUS = 'OLD' )
READ ( 7, * ) PDAT, TDAT, UDAT, VDAT, ODAT, ODATV

OPEN ( 8, FILE = PDAT, STATUS = 'OLD' )
OPEN ( 12, FILE = TDAT, STATUS = 'OLD' )
OPEN ( 13, FILE = UDAT, STATUS = 'OLD' )
OPEN ( 14, FILE = VDAT, STATUS = 'OLD' )
OPEN ( 21, FILE = ODAT )
OPEN ( 22, FILE = ODATV )

** READ IN H2 PROPERTY TABLE INTO ARRAYS **
DO 10 ITBL=1,4
READ(8,902) PTITLE
DO 10 I=1,NH2P(ITBL)
DO 10 J=1,NH2T(ITBL)
  IF(ITBL.EQ.1) READ(8,*) H2P1(I),H2T1(J),
    H2P1(I),H2T1(J),H2P1(I),H2T1(J)
  IF(ITBL.EQ.2) READ(8,*) H2P2(I),H2T2(J),
    H2P2(I),H2T2(J),H2P2(I),H2T2(J)
  IF(ITBL.EQ.3) READ(8,*) H2P3(I),H2T3(J),
    H2P3(I),H2T3(J),H2P3(I),H2T3(J)
  IF(ITBL.EQ.4) READ(8,*) H2P4(I),H2T4(J),
    H2P4(I),H2T4(J),H2P4(I),H2T4(J)
10 CONTINUE

** READ IN O2 PROPERTY TABLE INTO ARRAYS **
DO 20 ITBL=1,3
READ(8,902) PTITLE
DO 20 I=1,NO2P(ITBL)
DO 20 J=1,NO2T(ITBL)
  IF(ITBL.EQ.1) READ(8,*) O2P1(I),O2T1(J),
    O2P1(I),O2T1(J),O2P1(I),O2T1(J)
  IF(ITBL.EQ.2) READ(8,*) O2P2(I),O2T2(J),
    O2P2(I),O2T2(J),O2P2(I),O2T2(J)
  IF(ITBL.EQ.3) READ(8,*) O2P3(I),O2T3(J),
    O2P3(I),O2T3(J),O2P3(I),O2T3(J)
20 CONTINUE

** READ IN STEAM PROPERTY TABLES INTO ARRAYS **
C
DO 30 ITBL = 1, 1
C
READ(8,902) PTITLE
DO 30 I = 1, NH2OP(ITBL)
DO 30 J = 1, NH2OT(ITBL)
C
IF( ITBL .EQ. 1 ) READ(8,*) H2OP1(I),H2OT1(J),
1 H2OH1(I,J),H2OS1(I,J),H2OD1(I,J)
C
30 CONTINUE
C
READ (12,*) NDESC, NTTB
READ (12,*) ( DESC( I ), I = 1, NDESC )
READ (12,*) ( TTB( I ), I = 1, NTTB )
C
WRITE (21,901) ( DESC( I ), I = 1, NDESC )
C
READ (14,*) IENV, IPCTTH, MAXSTG, NHG, NTMFC, NTMFC, NTVOL
READ (14,*) ( IA(I), I = 1, NTMFC ),
2 ( IP(I), I = 1, NTMFC ),
3 ( IT(I), I = 1, NTMFC ),
4 ( IW(I), I = 1, NTMFC ),
5 ( MAT(I), I = 1, NTMFC ),
6 ( MIO(I), I = 1, NTMFC ),
7 ( NIO(I), I = 1, NTVOL )
C
DO 50 I = 1, NTMFC
READ (14,*) ( MODIR(I,J), J = 1, MIO(I) ),
1 ( IMFCN(I,J), J = 1, MIO(I) )
50 CONTINUE
C
DO 60 I = 1, NTVOL
READ (14,*) ( IODIR(I,J), J = 1, NIO(I) ),
1 ( IVDIR(I,J), J = 1, NIO(I) )
60 CONTINUE
C
IF ( NHG .GT. 0 ) THEN
READ (14,*) ( NODHG(I), I = 1, NHG ),
1 ( NH2HG(I), I = 1, NHG ),
2 ( NO2HG(I), I = 1, NHG ),
3 ( ICEFF(I), I = 1, NHG )
DO 70 I = 1, NHG
READ (14,*) ( IH2HG(I,J), J = 1, NH2HG(I) ),
1 ( IO2HG(I,J), J = 1, NO2HG(I) )
70 CONTINUE
ENDIF
C
READ (13,*) IPRPD, IITBD, IROW, JCOLA, JTORDX, JTORDY,
1 DPF, DTF,
2 ( UP(I), I = 1, NTMFC ),
3 ( UT(I), I = 1, NTMFC ),
4 ( UW(I), I = 1, NTMFC ),
5 ( UWMFC(I), I = 1, NTMFC ),
6 ( UEVOL(I), I = 1, NTVOL ),
7 ( USEVOL(I), I = 1, NTVOL )
C
CALL RECON
C
901 FORMAT ( 10 ( /, 1X, A24 ) )
STOP
END

SUBROUTINE RECON

C
C

CHARACTER*24 DESC
REAL JOULE

DIMENSION DDDPN(20), DDDTN(20), DHDPN(20), DHDTN(20),
1 DSIPN(20), DSGTN(20), DSGTN(20),
2 ASTD(20), HSTD(20), SSTD(20),
3 REV(20), REVP(20), REVNT(20), REVW(20),
4 CPQQT(100,100), A(100,100), TTBREV(100)

COMMON CPQ(200), CPQ(200), L1CP, CPB(200,200), NL1CP, NL2CP,
1 CPA(200), NE1CP, NE2CP, IRCP, MBASIS(300),
2 CPW(200), CPZ(200)

COMMON /VDAT/ IENV, IPCTTH, MAXSTG, NHG, NTMFC, NTNOD, NTVol,
1 IA(20), IP(20), IT(20), IW(20), MAT(20),
2 MIO(5), MODIR(5,20), IMPCN(5,20),
3 NIO(5), IODIR(5,20), IVOLN(5,20),
4 NH2HG(5), NO2HG(5), NODHG(5), ICEMP(5),
5 IH2HG(5,5), IO2HG(5,5)

COMMON /TDAT/ TTB(100), NDESC, NTTB, DESC(5)

COMMON /UDAT/ IPRPD, ITTBD, ITROA , JCOLA, ITORDX, ITORDY,
1 DPF , DTF , UP(20), UT(20), UW(20),
2 UEVOL(5), USVOL(5), UWMFC(5)

COMMON /H2PRP/
1 H2P1(15),H2T1(11),H2H1(15,11),H2S1(15,11),H2D1(15,11),
2 H2P2(20),H2T2(20,11),H2H2(20,11),H2S2(20,11),H2D2(20,11),
3 H2P3(29),H2T3(25),H2H3(29,25),H2S3(29,25),H2D3(29,25),
4 H2P4(23),H2T4(25),H2H4(23,25),H2S4(23,25),H2D4(23,25),
5 COMMON /O2PRP/
1 O2P1(13),O2T1(16),O2H1(13,16),O2S1(13,16),O2D1(13,16),
2 O2P2(13),O2T2(17),O2H2(13,17),O2S2(13,17),O2D2(13,17),
3 O2P3(5),O2T3(61),O2H3(5,61),O2S3(5,61),O2D3(5,61),
4 COMMON /H2OPRP/
1 H2OP1(7),H2OT1(13),H2OH1(7,13),H2OS1(7,13),H2OD1(7,13)

COMMON /STD/
1 HH2REF,H22REF,HHAREF,SH2REF,S02REF,SHAREF,SH2A,SO2A,
2 SWAAA

COMMON /TABLE/
1 NH2P(4),NH2T(4),NO2P(3),NO2T(3),NH2OP(1),NH2OT(1)

PARAMETER ( JOULE = 778.16, GC = 32.174 )

ISTG = 1

TENV = TTB( IENV )

PARAMETER ( JOULE = 778.16, GC = 32.174 )

ISTG = 1

TENV = TTB( IENV )
DO 10 I = 1, NTNOD
REVA(I) = TTB( IA(I) )
REVp(I) = TTB( IP(I) )
REVt(I) = TTB( IT(I) )
REVw(I) = TTB( IW(I) )
10 CONTINUE
C
20 DO 70 I = 1, NTNOD
   P = REVP(I)
   T = REVt(I)
   W = REVW(I)
   IF ( MAT(I) .GE. 3 ) GO TO 40
   IF ( MAT(I) .GE. 2 ) GO TO 30
C
   CALL PROP ( 1, P, T, 0.0, 0.0, H, S, RHO)
   DENS(I) = RHO
   HN = H - HH2REF
   HSTD(I) = HN
   SN = S - SH2REF + SH2A
   SSTD(I) = SN
   AN = HN - TENV * SN
   ASTD(I) = AN
C
   P2 = P + DPF * P
   CALL PROP ( 1, P2, T, 0.0, 0.0, H2, S2, RH02)
   HN2 = H2 - HH2REF
   SN2 = S2 - SH2REF + SH2A
   DDDPN(I) = ( RH02 - RHO ) / ( P2 - P )
   DHDPN(I) = ( HN2 - HN ) / ( P2 - P )
   DSDPN(I) = ( SN2 - SN ) / ( P2 - P )
C
   T2 = T + DTF * T
   CALL PROP ( 1, P, T2, 0.0, 0.0, H2, S2, RH02)
   HN2 = H2 - HH2REF
   SN2 = S2 - SH2REF + SH2A
   DDDTN(I) = ( RH02 - RHO ) / ( T2 - T )
   DHDTN(I) = ( HN2 - HN ) / ( T2 - T )
   DSDTN(I) = ( SN2 - SN ) / ( T2 - T )
C
   GO TO 70
C
30 CALL PROP ( 2, P, T, 0.0, 0.0, H, S, RHO)
   DENS(I) = RHO
   HN = H - HO2REF
   HSTD(I) = HN
   SN = S - SO2REF + SO2A
   SSTD(I) = SN
   AN = HN - TENV * SN
   ASTD(I) = AN
C
   P2 = P + DPF * P
   CALL PROP ( 2, P2, T, 0.0, 0.0, H2, S2, RH02)
   HN2 = H2 - HO2REF
   SN2 = S2 - SO2REF + SO2A
   DDDPN(I) = ( RH02 - RHO ) / ( P2 - P )
   DHDPN(I) = ( HN2 - HN ) / ( P2 - P )
   DSDPN(I) = ( SN2 - SN ) / ( P2 - P )
C
   T2 = T + DTF * T
CALL PROP ( 2, P, T2 ,0.0, 0.0, H2, S2, RHO2)

HN2 = H2 - HO2REF
SN2 = S2 - SO2REF + SO2A
DDDTN(I) = ( RHO2 - RHO ) / ( T2 - T )
DHDTN(I) = ( HN2 - HN ) / ( T2 - T )
DSDTN(I) = ( SN2 - SN ) / ( T2 - T )

GO TO 70

DO 42 IDHG = 1, NHG
IF ( NODHG(IDHG) .EQ. I ) THEN
   IMG = IDHG
   GO TO 44
ELSE
   ENDIF
END

CONTINUE

WH2 = 0.0
DO 50 IH2IN = 1, NH2HG( IHG )
   NNUM = IH2HG( IHG, IH2IN )
   WH2 = WH2 + REVW( NNUM )
CONTINUE

WO2 = 0.0
DO 60 IO2IN = 1, NO2HG( IHG )
   NNUM = IO2HG( IHG, IO2IN )
   WO2 = WO2 + REVW( NNUM )

CONTINUE

CEFF = TTB( ICEFF( IHG ) )
OF = WO2 / WH2
CALL PROP ( 4, P, T, OF, CEFF, HMX, SMIX, DMIX)
DENS(I) = DMIX
HSTD(I) = HMIX
SSTD(I) = SMIX
AMIX = HMIX * TENV * SMIX
ASTD(I) = AMIX

P2 = P + DPF * P
CALL PROP ( 4, P2, T, OF, CEFF, H2MIX, S2MIX, D2MIX)
DDDPN(I) = ( D2MIX - DMIX ) / ( P2 - P )
DHDPN(I) = ( H2MIX - HMIX ) / ( P2 - P )
DSDPN(I) = ( S2MIX - SMIX ) / ( P2 - P )

T2 = T + DTF * T
CALL PROP ( 4, P, T2, OF, CEFF, H2MIX, S2MIX, D2MIX)
DDDTN(I) = ( D2MIX - DMIX ) / ( T2 - T )
DHDTN(I) = ( H2MIX - HMIX ) / ( T2 - T )
DSDTN(I) = ( S2MIX - SMIX ) / ( T2 - T )

CONTINUE

M = 3 * NTMOD + 2 * NTMFC + 3 * NTVOI
N = 3 * NTMOD

DO 80 I = 1, M
DO 80 J = 1, N
A(I,J) = 0.0
CONTINUE
DO 82 I = 1, N
DO 82 J = 1, N
CPQQT(I,J) = 0.0
CONTINUE

DO 84 ITNOD = 1, NTNOD
I1 = ITNOD
I2 = ITNOD + NTNOD
I3 = ITNOD + 2 * NTNOD
CPQQT(I1,I1) = 4. / UW( ITNOD ) ** 2
CPQQT(I2,I2) = 4. / UP( ITNOD ) ** 2
CPQQT(I3,I3) = 4. / UT( ITNOD ) ** 2
CPQ(I1) = -4. / UW( ITNOD )
CPQ(I2) = -4. / UP( ITNOD )
CPQ(I3) = -4. / UT( ITNOD )
CONTINUE

DO 90 ITNOD = 1, NTNOD
I1 = ITNOD
I2 = ITNOD + NTNOD
I3 = ITNOD + 2 * NTNOD
A(I1,I1) = -1.
A(I2,I2) = -1.
A(I3,I3) = -1.
CPQ(N+I1) = 2. * UW( I1 )
CPQ(N+I2) = 2. * UP( I1 )
CPQ(N+I3) = 2. * UT( I1 )
CONTINUE

NQROW = 2 * N

DO 98 ITMFC = 1, NTMFC
SUMM = 0.
I1 = ITMFC
I2 = ITMFC + NTMFC

DO 92 IIO = 1, MIO( ITMFC )
IOD = MODIR( ITMFC, IIO )
IMN = IMFCN( ITMFC, IIO )
W = REVW( IMN )
SUMM = SUMM + IOD * ( W - UW( IMN ) )
CONTINUE

DO 94 IIO = 1, MIO( ITMFC )
IOD = MODIR( ITMFC, IIO )
IMN = IMFCN( ITMFC, IIO )
A(N+I1,IMN) = IOD
A(N+I2,IMN) = - IOD
CONTINUE

CPQ(NQROW+I1) = SUMM + UWHFC( ITMFC )
CPQ(NQROW+I2) = - SUMM + UWHFC( ITMFC )
CONTINUE

NAROW = N + 2 * NTMFC
NQROW = 2 * N + 2 * NTMFC

DO 110 ITVOL = 1, NTVOL
I1 = ITVOL
I2 = ITVOL + NTMFC
I3 = ITGOL + 2 * NTVOL
CEO = 0.
CSO = 0.
SUMMI = 0.

C DO 99 IIO = 1, NIO( ITGOL )
IOM = IOMIR( ITGOL, IIO )
INM = NIMIR( ITGOL, IIO )
AREA = REVA( INM )
W = REVW( INM )
RHO = DENS( INM )
ENTR = SSTD( INM )
AVAIL = ASTD( INM )

CEO1 = IOD * W * ENTH
IF ( AREA .GT. 0. ) THEN
CEO2 = IOD * W**3 / ( 2. * GC * JOULE * RHO**2 * AREA**2 )
ELSE
CEO2 = 0.
ENDIF
CEO = CEO + CEO1 + CEO2
CSO = CSO + IOD * W * ENTR
99 CONTINUE

C SUMEQ = 0.0
SUMSQ = 0.0

C DO 100 IIO = 1, NIO( ITGOL )
IOM = IOMIR( ITGOL, IIO )
INM = NIMIR( ITGOL, IIO )
AREA = REVA( INM )
W = REVW( INM )
P = REV/( INM )
T = REVT( INM )
RHO = DENS( INM )
ENTR = SSTD( INM )
AVAIL = ASTD( INM )
DDDP = DDDPN( INM )
DHP = DDDPN( INM )
DSDP = DSDPN( INM )
DSDT = DSDTN( INM )
DSDT = DSDTN( INM )

CEM = IOD * ENTH
CEP = IOD * W * DHP
CET = IOD * W * DHDP
CSM = IOD * ENTR
CSP = IOD * W * DSDP
CST = IOD * W * DSDT

C IF ( AREA .GT. 0. ) THEN
CEM = CEM + 3. * IOD * W**2 / ( 2. * GC * JOULE * RHO**2 * AREA**2 )
1 CEP = CEP - IOD * W**3 * DDDP / ( GC * JOULE * RHO**3 * AREA**2 )
1 CET = CET - IOD * W**3 * DDDT / ( GC * JOULE * RHO**3 * AREA**2 )
1
ENDIF

NROW1 = NAROW + I1
A(NROW1,IVN) = CEM
A(NROW1,IVN+NTNOD) = CEP
A(NROW1,IVN+2*NTNOD) = CET

NROW2 = NAROW + I2
A(NROW2,IVN) = -CEM
A(NROW2,IVN+NTNOD) = -CEP
A(NROW2,IVN+2*NTNOD) = -CET

NROW3 = NAROW+I3
A(NROW3,IVN) = -CSM
A(NROW3,IVN+NTNOD) = -CSP
A(NROW3,IVN+2*NTNOD) = -CST

SUMEQ = SUMEQ + CEM * UW(IVN) + CEP * UP(IVN) + CET * UT(IVN) + 1
SUMSQ = SUMSQ + CSM * UW(IVN) + CSP * UP(IVN) + CST * UT(IVN) + 1

DO 100 I = 1, N
DO 120 J = 1, N
CPM(I,J) = CPQQT(I,J)
CONTINUE

DO 110 I = 1, M
DO 130 J = 1, N
CPM(N+I,J) = A(I,J)
CPM(J,N+I) = -A(I,J)
CONTINUE

NP1 = N + 1
MN = M + N
DO 140 I = NP1, MN
DO 140 J = NP1, MN
CPM(I,J) = 0.0
CONTINUE

CALL CPIVOT(MN)
IF (IRCP.EQ.-1) WRITE(21,986)
IF (IRCP.EQ.-2) WRITE(21,987)
IF (IRCP.EQ.-3) WRITE(21,988)

WRITE(21,981)
WRITE(21,982)

DO 300 I = 1, NTNOD
WRITE(21,951) I, TTB(IP(I)), TTB(IT(I)), TTB(IW(I))
CONTINUE
WRITE ( 21, 983 )
WRITE ( 21, 982 )

DO 110 I = 1, NTNOD
REVW( I ) = REVW( I ) + ( CPZ( I ) - UW( I ) )
REVP( I ) = REVP( I ) + ( CPZ( I + NTNOD ) - UP( I ) )
REVT( I ) = REVT( I ) + ( CPZ( I + 2 * NTNOD ) - UT( I ) )
WREC = REVW( I )
PREC = REVP( I )
TREC = REVT( I )
WRITE ( 21, 951 ) I, PREC, TREC, WREC
110 CONTINUE

WRITE ( 21, 984 )
WRITE ( 21, 982 )

DO 320 I = 1, NTNOD
WADJ = REVW( I ) - TTB( IW( I ) )
PADJ = REVP( I ) - TTB( IP( I ) )
TADJ = REVT( I ) - TTB( IT( I ) )
WRITE ( 21, 951 ) I, PADJ, TADJ, WADJ
320 CONTINUE

WRITE ( 21, 985 )
WRITE ( 21, 982 )

DO 330 I = 1, NTNOD
WPCT = 100 * ( REVW( I ) - TTB( IW( I ) ) ) / TTB( IW( I ) )
PPCT = 100 * ( REVP( I ) - TTB( IP( I ) ) ) / TTB( IP( I ) )
TPCT = 100 * ( REVT( I ) - TTB( IT( I ) ) ) / TTB( IT( I ) )
WRITE ( 21, 951 ) I, PPCT, TPCT, WPCT
330 CONTINUE

IF ( ISTG .LT. MAXSTG ) THEN
   ISTG = ISTG + 1
   GO TO 20
ELSE
   ENDIF

CONTINUE

TTBREV( IENV ) = TTB( IENV )
TTBREV( IPCTTH ) = TTB( IPCTTH )

DO 510 I = 1, NTNOD
TTBREV( IA(I) ) = REVA( I )
TTBREV( IP(I) ) = REVP( I )
TTBREV( IT(I) ) = REVT( I )
TTBREV( IW(I) ) = REVW( I )
510 CONTINUE

DO 520 I = 1, NHG
TTBREV( ICEFF(I) ) = TTB( ICEFF(I) )
520 CONTINUE

WRITE ( 22, * ) NDESC, NTTB
WRITE ( 22, 990 ) ( DESC(I), I = 1, NDESC )
WRITE ( 22, 989 ) ( TTBREV(I), I = 1, NTTB )

951 FORMAT ( 9X, I6, 3F15.2 )
C FORMAT ( //IX, 'ORIGINAL NODE DATA' )
C FORMAT ( //IX, 'RECONCILED NODE DATA' )
C FORMAT ( //IX, 'BALANCING ADJUSTMENTS' )
C FORMAT ( //IX, 'PERCENT ADJUSTMENT' )
C FORMAT ( //IX, 'NO COMPLEMENTARY SOLUTION OBTAINED' )
C FORMAT ( //IX, 'TRIVIAL COMPLEMENTARY SOLUTION' )
C FORMAT ( //IX, 'SOLUTION OBTAINED' )
C FORMAT ( 6F12.3 )
C FORMAT ( 2X,1H',A24,1H' )
C
RETURN
END
C
C SUBROUTINE CPIVOT ( N )
C
COMMON AM(200,200),Q(200),L1,B(200,200),NL1,NL2,A(200),NE1,NE2,
1 IR,M BASIS(300),W(200),Z(200)
C
DESCRIPTION OF PARAMETERS IN COMMON
C
AM A TWO DIMENSIONAL ARRAY CONTAINING THE
C ELEME NTS OF MATRIX M.
C Q A SINGLY SUBSCRIPTED ARRAY CONTAINING THE
C ELEMENTS OF VECTOR Q.
C L1 AN INTEGER VARIABLE INDICATING THE NUMBER OF
C ITERATIONS TAKEN FOR EACH PROBLEM.
C B A TWO DIMENSIONAL ARRAY CONTAINING THE
C ELEMENTS OF THE INVERSE OF THE CURRENT BASIS.
C W A SINGLY SUBSCRIPTED ARRAY CONTAINING THE VALUES
C OF W VARIABLES IN EACH SOLUTION.
C Z A SINGLY SUBSCRIPTED ARRAY CONTAINING THE VALUES
C OF Z VARIABLES IN EACH SOLUTION.
C NL1 AN INTEGER VARIABLE TAKING VALUE 1 OR 2 DEPEND-
C ING ON WHETHER VARIABLE W OR Z LEAVES THE BASIS
C NE1 SIMILAR TO NL1 BUT INDICATES VARIABLE ENTERING
C NL2 AN INTEGER VARIABLE INDICATING WHAT COMPONENT
C OF W OR Z VARIABLE LEAVES THE BASIS.
C NE2 SIMILAR TO NL2 BUT INDICATES VARIABLE ENTERING
C A SINGLY SUBSCRIPTED ARRAY CONTAINING THE
C ELEMENTS OF THE TRANSFORMED COLUMN THAT IS
C ENTERING THE BASIS.
C IR AN INTEGER VARIABLE DENOTING THE PIVOT ROW AT
C EACH ITERATION. ALSO USED TO INDICATE
C ALGORITHM TERMINATION
C IR = -2 COMPLEMENTARY SOLUTION DETERMINED
C IR = -1 PROBLEM HAS NO COMPLEMENTARY SOLUTION
C MBASIS A SINGLY SUBSCRIPTED ARRAY-INDICATOR FOR THE
C BASIC VARIABLES. TWO INDICATORS ARE USED FOR
C EACH BASIC VARIABLE-ONE INDICATING WHETHER
C IT IS A W OR Z AND ANOTHER INDICATING WHAT
C COMPONENT OF W OR Z.
C
PROGRAM CALLING SEQUENCE, N IS THE SIZE OF MATRIX AM
C
CALL MATRIX ( N )
CALL INITIA ( N )
IF ( IR .EQ. -2 ) GO TO 5
SUBROUTINE MATRIX ( N )

PURPOSE - TO INITIALIZE THE VARIOUS INPUT DATA

COMMON AM(200,200), Q(200), L1, B(200,200), NL1, NL2, A(200),
               NEI, NE2, IR, MBASIS(300), W(200), Z(200)

IN ITERATION 1, BASIS INVERSE IS AN IDENTITY MATRIX

DO 5 J = 1, N
DO 4 I = 1, N
   IF ( I .EQ. J ) GO TO 3
   B(I,J) = 0.0
   GO TO 4
3   B(I,J) = 1.0
4   CONTINUE
5   CONTINUE

RETURN
END

SUBROUTINE INITIA ( N )

PURPOSE - TO FIND THE INITIAL ALMOST COMPLEMENTARY SOLUTION
          BY ADDING AN ARTIFICIAL VARIABLE Z0.

COMMON AM(200,200), Q(200), L1, B(200,200), NL1, NL2, A(200),
               NEI, NE2, IR, MBASIS(300), W(200), Z(200)

SET Z0 EQUAL TO THE MOST NEGATIVE Q(I)

I = 1
J = 2
1 IF ( Q(I) .LE. Q(J) ) GO TO 2
   I = J
2 J = J + 1
   IF ( J .LE. N ) GO TO 1

UPDATE Q VECTOR

IR = I
   T1 = -Q( IR )
   IF ( T1 .LE. 0.0 ) GO TO 9
   DO 3 I = 1, N
      Q(I) = Q(I) + T1
SUBROUTINE NEWBAS ( N )

C PURPOSE - TO FIND THE NEW BASIS COLUMN TO ENTER IN
C TERMS OF THE CURRENT BASIS.

C COMMON AM(200,200), Q(200), LI, B(200,200), NL1, NL2, A(200),
1 NE1, NE2, IR, MBASIS(300), W(200), Z(200)

C IF NL1 IS NEITHER 1 NOR 2 THEN THE VARIABLE ZO LEAVES THE
C BASIS INDICATING TERMINATION WITH A COMPLEMENTARY SOLUTION
C
C IF ( NL1 .EQ. 1 ) GO TO 2
C IF ( NL1 .EQ. 2 ) GO TO 5
C
C CALL SOLVE ( N )
C IR = -3
C RETURN

C 2 NE1 = 2
C NE2 = NL2
C
C UPDATE NEW BASIC COLUMN BY MULTIPLYING BY BASIS INVERSE.
C
C DO 4 I = 1, N
C T1 = 0.0
C DO 3 J = 1, N
C T1 = T1 - B(I,J) * AM(J, NE2 )
C A(I) = T1
C 4 CONTINUE
C RETURN

C
SUBROUTINE SORT ( N )
C
PURPOSE - TO FIND THE PIVOT ROW FOR THE NEXT ITERATION BY THE
USE OF (SIMPLEX-TYPE) MINIMUM RATIO RULE.
C
COMMON AM(200,200), Q(200), L1, B(200,200), NL1, NL2, A(200),
1 NE1, NE2, IR, MBASIS(300), W(200), Z(200)
C
I = 1
1 IF ( A(I) .GT. 0.0 ) GO TO 2
I = I + 1
IF ( I .GT. N ) GO TO 6
GO TO 1
C
2 T1 = Q(I) / A(I)
IR = I
3 I = I + 1
IF ( I .GT. N ) GO TO 5
IF ( A(I) .GT. 0.0 ) GO TO 4
GO TO 3
C
4 T2 = Q(I) / A(I)
IF ( T2 .GE. T1 ) GO TO 3
IR = I
T1 = T2
GO TO 3
C
5 RETURN
C
FAILURE OF THE RATIO RULE INDICATES TERMINATION WITH
NO COMPLEMENTARY SOLUTION.
C
6 IR = -1
RETURN
END
C
C
SUBROUTINE PIVOT ( N )
C
PURPOSE - TO PERFORM THE PIVOT OPERATION BY UPDATING THE
THE INVERSE OF THE BASIS AND Q VECTOR.
C
COMMON AM(200,200), Q(200), L1, B(200,200), NL1, NL2, A(200),
1 NE1, NE2, IR, MBASIS(300), W(200), Z(200)
C
DO 1 I = 1, N
1 B(IR,I) = B(IR, I) / A(IR)
Q(IR) = Q(IR) / A(IR)
DO 3 I = 1, N
IF ( I .EQ. IR ) GO TO 3
Q(I) = Q(I) - Q(IR) * A(I)
DO 2 J = 1, N
B(I,J) = B(I,J) - B(IR,J) * A(I)
2 CONTINUE
3 CONTINUE

C UPDATE THE INDICATOR VECTOR OF BASIC VARIABLES
NL1 = MBASIS(IR)
L = N + IR
NL2 = MBASIS(L)
MBASIS(IR) = NE1
MBASIS(L) = NE2
L1 = L1 + 1

RETURN
END

SUBROUTINE SOLVE ( N )
C
C PURPOSE - TO CORRELATE COMPLEMENTARY PROBLEM SOLUTION
C
COMMON AM(200,200), Q(200), LI, B(200,200), NLI, NL2, A(200),
1 NE1, NE2, IR, MBASIS(300), W(200), Z(200)
C
DO 1 I = 1, N
W(I) = 0.0
Z(I) = 0.0
1 CONTINUE
I = N + I
J = 1

2 K1 = MBASIS(I)
K2 = MBASIS(J)
C
IF ( Q(J) .GE. 0.0 ) GO TO 3
Q(J) = 0.0
3 IF ( K2 .EQ. 1 ) GO TO 5
Z(K1) = Q(J)
GO TO 7
5 W(K1) = Q(J)
7 I = I + 1
J = J + 1
IF ( J .LE. N ) GO TO 2
C
RETURN
END

C******************************************************************************
SUBROUTINE PROP (MAT,PRSI,TP,OF,CEFF,ZENTH,ZENTR,ZDENS)
C
PROP - PROPERTY PROGRAM CALCULATING HYDROGEN,
OXYGEN, STEAM AND HOT GAS PROPERTIES

COMMON /H2PRP/
* H2P(15), H2T(11), H2H(15,11), H2S(15,11), H2D(15,11),
* H2P(20), H2T(12), H2H(20,11), H2S(20,11), H2D(20,11),
* H2P(29), H2T(25), H2H(29,25), H2S(29,25), H2D(29,25),
* H2P(33), H2T(4), H2H(33,4), H2S(33,4), H2D(33,4),

COMMON /O2PRP/
* O2P(13), O2T(16), O2H(13,16), O2S(13,16), O2D(13,16),
* O2P(13), O2T(17), O2H(13,17), O2S(13,17), O2D(13,17),
* O2P(5), O2T(61), O2H(5,61), O2S(5,61), O2D(5,61),

COMMON /H2OPRP/
* H2O(7), H2O(13), H2O(7,13), H2O(7,13), H2O(7,13)

COMMON /TABLE/
* NH2P(4), NH2T(4), NO2P(3), NO2T(3), NH2OP(1), NH2OT(1)

DIMENSION
* TSH2(11), PSH2(11), HLH2(11), HVH2(11), SLH2(11), SVH2(11),
* DLH2(11), DVH2(11),
* TSO2(16), PSO2(16), HLO2(16), HVO2(16), SLO2(16), SVO2(16),
* DLO2(16), DVVO2(16)

TSH2 - H2 SATURATION TEMPERATURE
PSH2 - H2 SATURATION PRESSURE
HLH2 - H2 SATURATION ENTHALPY - LIQUID
HVH2 - H2 SATURATION ENTHALPY - VAPOR
SLH2 - H2 SATURATION ENTROPY - LIQUID
SVH2 - H2 SATURATION ENTROPY - VAPOR
DLH2 - H2 SATURATION DENSITY - LIQUID
DVH2 - H2 SATURATION DENSITY - VAPOR

DATA (TSH2(J), J=1,11)/
* 30.0, 32.0, 34.0, 36.0, 38.0, 40.0, 42.0, 44.0, 46.0, 48.0, 50.0/

DATA (PSH2(J), J=1,11)/
* 4.170, 6.446, 9.527, 13.561, 18.694, 25.089, 32.915, 42.334,
* 53.514, 66.625, 81.838/

DATA (HLH2(J), J=1,11)/
* -123.995, -120.090, -115.893, -111.380, -106.524, -101.289,
* -95.636, -89.513, -82.850, -75.556, -67.493/

DATA (HVH2(J), J=1,11)/
* 70.977, 74.584, 77.848, 80.729, 83.256, 85.199, 86.614, 87.431,
* 87.546, 86.817, 85.043/

DATA (SLH2(J), J=1,11)/
* 1.506, 1.629, 1.752, 1.876, 2.002, 2.129, 2.259, 2.391, 2.528,
* 2.670, 2.819/

DATA (SVH2(J), J=1,11)/
* 8.005, 7.713, 7.451, 7.214, 6.998, 6.794, 6.601, 6.415, 6.234,
* 6.054, 5.871/

DATA (DLH2(J), J=1,11)/
* 4.6500, 4.5832, 4.5127, 4.4378, 4.3580, 4.2724, 4.1801, 4.0798,
* 3.9698, 3.8479, 3.7108/

DATA (DVH2(J), J=1,11)/
DATA (T02(J),J=1,16)/
* 160.0,164.0,168.0,172.0,176.0,180.0,184.0,188.0,192.0,
* 196.0,200.0,204.0,208.0,212.0,216.0,220.0/
DATA (P02(J),J=1,16)/
* 12.810,16.183,20.200,24.935,30.467,36.876,44.243,52.654,
* 62.194,72.951,85.013,113.421,129.952,148.162,
* 168.146/
DATA (H02(J),J=1,16)/
* -58.356,-56.730,-55.096,-53.455,-51.804,-50.144,-48.473,
* -46.790,-45.093,-43.380,-41.650,-39.901,-38.130,-36.334,
* -34.511,-32.657/
DATA (HVO2(J),J=1,16)/
* 33.777,34.457,35.110,35.734,36.326,36.884,37.408,37.894,
* 38.340,38.745,39.109,39.419,39.683,39.894,40.049,40.144/
DATA (SLO2(J),J=1,16)/
* 0.698,0.708,0.717,0.727,0.736,0.746,0.755,0.764,0.772,
* 0.781,0.790,0.798,0.806,0.815,0.823,0.831/
DATA (SVO2(J),J=1,16)/
* 1.273,1.263,1.254,1.245,1.237,1.229,1.221,1.214,1.207,
* 1.200,1.193,1.187,1.180,1.174,1.168,1.162/
DATA (DLO2(J),J=1,16)/
* 71.630,70.941,70.243,69.536,68.818,68.089,67.347,66.593,
* 65.823,65.037,64.234,63.412,62.567,61.699,60.804,59.880/
DATA (DVO2(J),J=1,16)/
* 0.246,0.305,0.374,0.455,0.547,0.653,0.774,0.911,1.065,
* 1.239,1.433,1.650,1.893,2.162,2.461,2.794/

51 FORMAT(/3X,'PROP - REQUESTED PRS > ','F7.2,2X,
* 'AND TMP > ','F7.2,2X,'FOR H2 IS OUT OF RANGE')
52 FORMAT(/3X,'PROP - REQUESTED PRS > ','F7.2,2X,
* 'AND TMP > ','F7.2,2X,'FOR O2 IS OUT OF RANGE')
53 FORMAT(/3X,'PROP - REQUESTED PRS > ','F7.2,2X,
* 'AND TMP > ','F7.2,2X,'FOR STEAM IS OUT OF RANGE')

** INTERPOLATE RESULTS FROM SINGLE ARRAY **

IPRP=0
NPX1=2
NPY1=2
ZENTH=0.0
GO TO (10,20,30,40) MAT

IF(TMPI.GT. 70.0.AND.TMPI.LT. 110.0) IPRP=2
IF(TMPI.GT.1400.0.AND.TMPI.LT.2000.0) IPRP=4
GO TO (11,12,13,14) IPRP

IF(TMPI.GT. 30.0.AND.TMPI.LT. 50.0) IPRP=1
IF(TMPI.GT. 70.0.AND.TMPI.LT. 110.0) IPRP=2
IF(TMPI.GT.1400.0.AND.TMPI.LT.2000.0) IPRP=4
GO TO (10,20,30,40) MAT

IF(PRSl.GT. 20.0.OR.PRSl.GT. 370.0) GO TO 50
CALL PRPSAT(PRSl,TMPI,ZENTH,
* NH2P(1),NH2T(1),11,29.95,50.05,
* H2P1,H2T1,H2H1,PSH2,TSH2,HSH2,H2H1,H2S1,H2D1
CALL PRPSAT(PRSl,TMPI,ZENTR,
* TSH2(11),NH2P(1),NH2T(1),11,29.95,50.05,
* H2P1,H2T1,H2S1,PSH2,TSH2,SLH2,SVH2)
CALL PRPSAT(PRSl,TMPI,ZDENS,
* TSH2(11),NH2P(1),NH2T(1),11,29.95,50.05,
* H2P1,H2T1,H2D1,PSH2,TSH2,DLH2,DVH2)
RETURN

IF(PRSl.GT. 3400.0.OR.PRSl.GT. 7200.0) GO TO 50
CALL ITERP2(PRSl,TMPI,H2P2,H2T2,H2H2,
* NH2P(2),NH2T(2),NPX1,NPY1,NH2P(2),ZENTH,N1)
CALL ITERP2(PRSl,TMPI,H2P2,H2T2,H2S2,
* NH2P(2),NH2T(2),NPX1,NPY1,NH2P(2),ZENTR,N1)
CALL ITERP2(PRSl,TMPI,H2P2,H2T2,H2D2,
* NH2P(2),NH2T(2),NPX1,NPY1,NH2P(2),ZDENS,N1)
RETURN

IF(PRSl.GT. 1400.0.OR.PRSl.GT. 7000.0) GO TO 50
CALL ITERP2(PRSl,TMPI,H2P3,H2T3,H2H3,
* NH2P(3),NH2T(3),NPX1,NPY1,NH2P(3),ZENTH,N1)
CALL ITERP2(PRSl,TMPI,H2P3,H2T3,H2S3,
* NH2P(3),NH2T(3),NPX1,NPY1,NH2P(3),ZENTR,N1)
CALL ITERP2(PRSl,TMPI,H2P3,H2T3,H2D3,
* NH2P(3),NH2T(3),NPX1,NPY1,NH2P(3),ZDENS,N1)
RETURN

IF(PRSl.GT. 1400.0.OR.PRSl.GT. 5800.0) GO TO 50
CALL ITERP2(PRSl,TMPI,H2P4,H2T4,H2H4,
* NH2P(4),NH2T(4),NPX1,NPY1,NH2P(4),ZENTH,N1)
CALL ITERP2(PRSl,TMPI,H2P4,H2T4,H2S4,
* NH2P(4),NH2T(4),NPX1,NPY1,NH2P(4),ZENTR,N1)
CALL ITERP2(PRSl,TMPI,H2P4,H2T4,H2D4,
* NH2P(4),NH2T(4),NPX1,NPY1,NH2P(4),ZDENS,N1)
RETURN

IF(TMPl.GT. 160.0.AND.TMPI.LT. 240.0) IPRP=1
IF(IPRP.EQ.1.AND.PRSl.LT.650.0) IPRP=1
IF(IPRP.EQ.1.AND.PRSl.GT.650.0) IPRP=2
IF(TMPl.GT. 600.0.AND.TMPI.LT.1500.0) IPRP=3
GO TO (21,22,23) IPRP

IF(PRSl.LT. 30.0.OR.PRSl.GT. 630.0) GO TO 50
IF(TMPl.GT. 160.0.AND.TMPI.LT. 219.9) GO TO 50
CALL PRPSAT(PRSl,TMPI,ZENTH,
**TSO2(16), NO2P(1), NO2T(1), 16, 159.95, 220.05,**
**O2, O2(1), O2H1, PSO2, TSO2, HLO2, HVO2**

CALL PRPSAT(PRSI, TMP1, ZENTR,
**TSO2(16), NO2P(1), NO2T(1), 16, 159.95, 220.05,**
**O2, O2(1), O2S1, PSO2, TSO2, SLO2, SVO2**

CALL PRPSAT(PRSI, TMP1, ZDENS,
**TSO2(16), NO2P(1), NO2T(1), 16, 159.95, 220.05,**
**O2, O2(1), O2D1, PSO2, TSO2, DLO2, DVO2**

RETURN

C
22 IF(PRSI.LT.2000.0.OR.PRSI.GT.8000.0) GO TO 50
CALL ITERP2(PRSI, TMP1, O2P2, O2T2, O2H2,
**NO2P(2), NO2T(2), NPX1, NPY1, NO2P(2), ZENTH, N1**)
CALL ITERP2(PRSI, TMP1, O2P2, O2T2, O2S2,
**NO2P(2), NO2T(2), NPX1, NPY1, NO2P(2), ZENTR, N1**)
CALL ITERP2(PRSI, TMP1, O2P2, O2T2, O2D2,
**NO2P(2), NO2T(2), NPX1, NPY1, NO2P(2), ZDENS, N1**)
RETURN

C
23 IF(PRSI.LT.2000.0.OR.PRSI.GT.4000.0) GO TO 50
CALL ITERP2(PRSI, TMP1, O2P2, O2T2, O2H3,
**NO2P(3), NO2T(3), NPX1, NPY1, NO2P(3), ZENTH, N1**)
CALL ITERP2(PRSI, TMP1, O2P2, O2T3, O2S3,
**NO2P(3), NO2T(3), NPX1, NPY1, NO2P(3), ZENTR, N1**)
CALL ITERP2(PRSI, TMP1, O2P3, O2T3, O2D3,
**NO2P(3), NO2T(3), NPX1, NPY1, NO2P(3), ZDENS, N1**)
RETURN

C
30 IF(TMPI.LT.1400.0.OR.TMPI.GT.2000.0) GO TO 50
IF(PRSI.LT.100.0.OR.PRSI.GT.700.0) GO TO 50
CALL ITERP2(PRSI, TMP1, H2OP1, H2OT1, H2OH1,
**NH2OP(1), NH2OT(1), NPX1, NPY1, NH2OP(1), ZENTH, N1**)
CALL ITERP2(PRSI, TMP1, H2OPI, H2OT1, H2OS1,
**NH2OP(1), NH2OT(1), NPX1, NPY1, NH2OP(1), ZENTR, N1**)
CALL ITERP2(PRSI, TMP1, H2OPI, H2OT1, H2OD1,
**NH2OP(1), NH2OT(1), NPX1, NPY1, NH2OP(1), ZDENS, N1**)
RETURN

C
40 CALL PRPMIX(PRSI, TMP1, OF, CEFF, HMIX, SMIX)
ZENTH=HMIX
ZENTR=SMIX
ZDENS=0.0
RETURN

C
50 IF(MAT.EQ.1) WRITE(21,51) PRSI, TMP1
IF(MAT.EQ.2) WRITE(21,52) PRSI, TMP1
IF(MAT.EQ.3) WRITE(21,53) PRSI, TMP1
RETURN

C
END

******************************************************************************
SUBROUTINE PRPMIX (P, TMP1, OF, CEFF, HMIX, SMIX)

COMMON /H2PRP/
**H2P1(15), H2T1(11), H2H1(15,11), H2S1(15,11), H2D1(15,11),**
* H2P2(20), H2T2(11), H2H2(20, 11), H2S2(20, 11), H2D2(20, 11),
* H2P3(29), H2T3(25), H2H3(29, 25), H2S3(29, 25), H2D3(29, 25),
COMMON /O2PRP/
  * O2P1(13), O2T1(16), O2H1(13, 16), O2S1(13, 16), O2D1(13, 16),
  * O2P2(13), O2T2(17), O2H2(13, 17), O2S2(13, 17), O2D2(13, 17),
  * O2P3(5), O2T3(61), O2H3(5, 61), O2S3(5, 61), O2D3(5, 61)
COMMON /H2OPRP/
  * H2OPI(7), H2OT1(13), H2OH1(7, 13), H2OS1(7, 13), H2OD1(7, 13)

COMMON /TABLE/
* NH2P(4), NH2T(4), NO2P(3), NO2T(3), NH2OP(1), NH2OT(1)
COMMON /STD/
* HH2REF, HO2REF, HWAREF, SH2REF, SO2REF, SWAREF, SH2A, SO2A,
* SWAA

XMWH2 = 2.0160
XMWO2 = 31.9988
XMWH20 = 18.0153
HCOMB = -6825.6550

NPX1 = 2
NPY1 = 2
ITST1 = 0
ITST2 = 0
ITST3 = 0
ITST4 = 0
ITST5 = 0
ITST6 = 0

XF = 1.0 / (1.0 + OF)
XO = 1.0 - XF
XH2 = XF - XO * 2.0 * CEFF * XMWH2 / XMWH2
XH20 = XO * 2.0 * CEFF * XMWH20 / XMWH2
XO2 = 1.0 - XH2 - XH20

EH2 = XH2 / XMWH2
EH20 = XH20 / XMWH20
EO2 = XO2 / XMWH2
ET = EH2 + EH20 + EO2

YH2 = EH2 / ET
YH20 = EH20 / ET
YO2 = 1.0 - YH2 - YH20

PH2 = P * YH2
PH20 = P * YH20
PO2 = P * YO2

IF(TMPL.LT.1000.0.OR.TMPL.GT.2000.0) ITST1=1
IF(PH2.LT.1400.0.OR.PH2.GT.5800.0) ITST2=1
CALL ITERP2(PH2, TMPL, H2P4, H2T4, H2H4,
  * NH2P(4), NH2T(4), NPX1, NPY1, NH2P(4), HH2, N1)
CALL ITERP2(PH2, TMPL, H2P4, H2T4, H2S4,
  * NH2P(4), NH2T(4), NPX1, NPY1, NH2P(4), SH2, N1)

IF(TMPL.LT.1400.0.OR.TMPL.GT.2000.0) ITST3=1
IF(PH2O.LT.100.0.OR.PH2O.GT.700.0) ITST4=1
CALL ITERP2(PH2O, TMPL, H2OP1, H2OT1, H2OH1,
IF(YC2.LT.0.001) THEN
  DH02 = 0.0
  DS02 = 0.0
ELSE
  IF(TMPI.GT.600.0.AND.TMPI.LT.1500.0) ITST5=1
  IF(PO2.LT.2000.0.OR.PO2.GT.4000.0) ITST6=1
  CALL ITERP2(PO2,TMPI,O2P3,O2T3,O2S3,
* NO2P(3),NO2T(3),NPXI,NPYI,NO2P(3),SO2,NI)
  CALL ITERP2(PO2,TMPI,O2P3,O2T3,O2S3,
* NO2P(3),NO2T(3),NPXI,NPYI,NO2P(3),SO2,NI)
DHO2 = HO2 - HO2REF
DSO2 = SO2 - SO2REF + S02A
ENDIF

DHH2 = HH2 - HH2REF
DHH2OM = (HH20 - HWAREF) + HCOMB
DSH2 = SH2 - SH2REF + SH2A
DSH2O = SH2O - SWAREF + SWAA

HMIX = XH2*DHH2 + XH20*DHH20M + XO2*DHO2
SMIX = XH2*DSH2 + XH20*DSH20 + XO2*DSO2

IF (ITST1.EQ.1.OR.ITST2.EQ.1) WRITE(21,51) PH2,TMPI
IF (ITST3.EQ.1.OR.ITST4.EQ.1) WRITE(21,52) PH20,TMPI
IF (ITST5.EQ.1.OR.ITST6.EQ.1) WRITE(21,53) PO2,TMPI

51 FORMAT(/3X,'PRPMIX - REQUESTED PH2 PRS > ',F7.2,2X,
* 'AND TMP > ',F7.2,2X,'FOR "H2" IS OUT OF RANGE')
52 FORMAT(/3X,'PRPMIX - REQUESTED PH20 PRS > ',F7.2,2X,
* 'AND TMP > ',F7.2,2X,'FOR "H20" IS OUT OF RANGE')
53 FORMAT(/3X,'PRPMIX - REQUESTED PO2 PRS > ',F7.2,2X,
* 'AND TMP > ',F7.2,2X,'FOR "O2" IS OUT OF RANGE')

RETURN
END
ZTSAT=0.0
ARGA=0.0
ARGB=0.0
ZTSATT=0.0

CALL ITERP2(X,Y,PRS1,TMP1,PROP,NXI,NY1,NPX1,NPY1,NR1,ZPROP1,N1)
FPROP=ZPROP1
IF(Y.GT.TCRT) GO TO 70
CALL ITERP1(X,PRS2,TMP2,NX1,NPY1,PRSI,TMPI,PROP,NXI,NPY1,NR1,ZTSAT,N1)
IF(Y.LT.ZTSAT) GO TO 61

* * GAS CALCULATIONS * *

CALL ITERP1(X,PRS2,PROPV,NX2,NPX2,ZPGAS,N2)
CALL ITERP2(X,YH,PRS1,TMP1,PROP,NX1,NY1,NPX1,NPY1,NR1,ZTST,N1)
DTST=ZTST-ZPGAS
IF(DTST.GT.0.0001) GO TO 50
ZPGAS=ZPGAS
IF(ZPROP1.LT.ZPGAS) GO TO 70
GO TO 51
50 ZPHGAS=ZPGAS
IF(ZPROP1.GT.ZPGAS) GO TO 70
C
51 LPR=1
53 PRSD=PRS1(LPR)-0.0001
IF(PRSD.GT.X) GO TO 52
LPR=LPR+1
GO TO 53
C
52 ARG=PRS1(LPR)
CALL ITERP1(ARGA,PRS2,TMP2,NX2,NPX2,ZTSATT,N2)

C
54 TMPD=TMP1(LTP)-0.0001
IF(TMPD.GT.ZTSATT) GO TO 55
LTP=LTP+1
GO TO 54
C
55 ARGB=TMP1(LTP)
YY=ARGB
IF(DTST.GT.0.0001) CALL ITERP2(X,YY,PRS1,TMP1,PROP,NX1,NY1,
* NPX1,NPY1,NR1,ZPGAS,N1)
IF(DTST.LT.0.0001) CALL ITERP2(X,YY,PRS1,TMP1,PROP,NX1,NY1,
* NPX1,NPY1,NR1,ZTST,N1)
ZPROP=ZPHGAS-(ZPHGAS-ZPGAS)*((ARGB-Y)/(ARGB-ZTSAT))
FPROP=ZPROP
C
GO TO 70

* * LIQ CALCULATIONS * *

61 CALL ITERP1(X,PRS2,PROPL,NX2,NPX2,ZPLIQ,N2)
CALL ITERP2(X,YL,PRS1,TMP1,PROP,NX1,NPY1,PRSI,TMPI,PROP,NXI,NPY1,NR1,ZTST,N1)
DTST=ZTST-ZPLIQ
IF(DTST.GT.0.0001) GO TO 59
ZPLIQ=ZPLIQ
IF(ZPROP1.LT.ZPLIQ) GO TO 70
GO TO 60
59 ZPHLIQ=ZPLIQ
IF(ZPROP1.GT.ZPLIQ) GO TO 70

C 60 LPR=1
63 PRSD=FRSI(LPR)-0.0001
IF(PRSD.GT.X) GO TO 62
LPR=LPR+1
GO TO 63

C 62 ARGA=FRSI(LPR-1)
    CALL ITERPI(ARGA,PRS2,TMP2,NX2,NPX2,ZTSATT,N2)

C 64 LTP=1
66 TMPD=TMPI(LTP)-0.0001
IF(TMPD.GT.ZTSATT) GO TO 65
LTP=LTP+1
GO TO 64

C 65 ARGB=TMPI(LTP-1)
    YY=ARGB
    IF(DTST.GT.0.0001) CALL ITERP2(X,YY,PRSI,TMPI,PROP,NXI,NYI,
*   NPXI,NPYI,NRI,ZPLLIQ,NI)
    IF(DTST.LT.0.0001) CALL ITERP2(X,YY,PRSI,TMPI,PROP,NXI,NYI,
*   NPXI,NPYI,NRI,ZPHLIQ,NI)
    ZPROP=ZPHLIQ-(ZPHLIQ-ZPLLIQ)*((ZTSAT-YY)/(ZTSAT-ARGB))
    FPROP=ZPROP

C 70 CONTINUE

C RETURN
END

C****************************************************************************************************
SUBROUTINE ITERPI (X,XT,YT,NX,NPX,Y,NERR)
C
C ITERPI - SINGLE INTERPOLATION ROUTINE.
C
C DIMENSION XT(1),YT(1)
NERR=0
INTER=1
NP=NPX
IF(NX.LT.NP) NP=NX
IH=NP/2
I=1
IF(XT(I)-X)30,20,10
10 IH=0
12 NERR=1
GO TO 70
13 NERR=2
GO TO 70
20 INTER=2
22 Y=YT(I)
GO TO 999
30 I=NX
IF(XT(I)-X)13,20,40
40 N1=1
N2=NX
45 MP=(N1+N2)/2
50 IF(XT(MP)-X)52,54,56
52 N1=MP
GO TO 60
54 I=MP
GO TO 20
56 N2=MP
60 IF((N2-N1).NE.1)GO TO 45
   C
   IF (N2.GT.(IH+1)) GO TO 65
   I=IH+1
   GO TO 70
65 I=N2
   C
   IF(N2.GT.I) I=N2
70 K=I-IH
   N=K+NP-1
   Y=0.
   IF(N-NX)90,90,80
80 N=NX
   N=NX-NP+1
90 DO 120 J=K, N
   P=1.0
   DO 110 I=K, N
      IF(I-J)100,110,100
100 P=P*(X-XT(I))/(XT(J)-XT(I))
   CONTINUE
   Y=Y+YT(J)*P
120 CONTINUE
   C
   ENTRY ENTERP (X,XT,YT,Y)
   Y=0.
   GO TO (90,22),INTER
999 CONTINUE
   RETURN
END

C**ITERP2 (X,Y,XT,YT,ZT,NX,NY,NPX,NPY,NR,Z,NERR)
C
C
DIMENSION XT(1),YT(1),ZT(NR,1),ZC(15)
NERRB=0
NPYY=NPY
IF(NY.LT.NPY) NPYY=NY
IH=NPYY/2
I=1
IF(YT(I)-Y)30,20,10
10 IH=0
12 NERRB=201
   GO TO 70
13 NERRB=204
   GO TO 70
20 CALL ITERP1(X,XT,ZT(I,1),NX,NPX,Z,NERR)
   GO TO 999
30 I=NY
   IF(YT(I)-Y)13,20,40
40 N1=1
   N2=NY
45 MP=(N1+N2)/2
50 IF(YT(MP)-Y)52,54,56
52 N1=MP
   GO TO 60
54 I=MP
   GO TO 20
56 N2=MP
60 IF((N2-N1).NE.1) GO TO 45
    I=N2
70   IF(I.LT.(IH+1)) I=IH+1
   K=I-IH
   N=K+NPYY-1
   IF(N-NY)90,90,80
80   N=NY
   K=NY-NPYY+1
90   J=0
   DO 100 I=K, N
      J=J+1
   IF(J.NE.1) GO TO 95
      CALL ITERP1(X,XT,ZT(I,I),NX,NPX,ZC(J),NERRA)
      GO TO 100
95   CALL ENTERP(X,XT,ZT(I,I),ZC(J))
100  CONTINUE
999 NERR=NERRA+NERRB
      RETURN
END
**Abstract**

Effort was dedicated to development and testing of a formal strategy for reconciling uncertain test data with physically limited computational prediction. Specific weaknesses in the logical structure of the current PBM version are described with emphasis given to the main routing subroutines BAL and DATRED. Selected results from a variational analysis of PBM predictions are compared to TTB variational study results to assess PBM predictive capability. The motivation for systematic integration of uncertain test data with computational predictions based on limited physical models is provided. The theoretical foundation for the reconciliation strategy developed in this effort is presented, and results of a reconciliation analysis of the SSME high pressure fuel side turbopump subsystem are examined.