COMPARISONS OF STEADY-STATE AND TRANSIENT THERMAL HYDRAULIC RESULTS FROM SAS-DIF3DK AND RELAP5 MOD3.2 FOR AN RBMK REACTOR*

F. E. Dunn

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.
DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.
COMPARISONS OF STEADY-STATE AND TRANSIENT THERMAL HYDRAULIC RESULTS FROM SAS-DIF3DK AND RELAP5 MOD3.2 FOR AN RBMK REACTOR

F. E. Dunn
Argonne National Laboratory, Reactor Analysis Division
9700 S. Cass Avenue
Argonne, Illinois 60439
(630) 252-4692

ABSTRACT

The SAS-DIF3DK code couples a detailed 3-D neutron kinetics treatment with a detailed thermal hydraulics treatment. One goal of the work on SAS-DIF3DK is to produce a detailed code that will run a wide range of transients in real time. Achieving this goal will require efficient numerical methods and efficient coding, and it will probably require the use of multiple processors for larger problems. In order to obtain clean code-to-code thermal hydraulics comparisons with a recognized and established code, a detailed thermal hydraulic model was set up for an RBMK assembly and its associated piping. The same identical input model was implemented in both SAS-DIF3DK and RELAP5 mod 3.2. Both steady-state and transient thermal hydraulics calculations were made with this model. Except for cladding temperatures in one transient, the SAS-DIF3DK results were similar to or almost identical to the RELAP5 results, and SAS-DIF3DK ran an order of magnitude or more faster than RELAP5. The cladding temperature differences can be explained in terms of different post-DNB models and heat transfer coefficients.

I. INTRODUCTION

The SAS-DIF3DK code is being developed as a test bed for advanced modeling techniques, using a 3-D neutron kinetics treatment coupled with a detailed thermal hydraulics treatment. The thermal hydraulics treatment supplies detailed material densities and temperatures to the neutron kinetics treatment, and the neutron kinetics treatment supplies detailed power densities to the thermal hydraulics treatment. The 3-D neutron kinetics treatment is described in Ref. 1. Previous SAS-DIF3DK validation and verification work, such as that described in Ref. 1, has emphasized neutronics rather than the thermal hydraulics aspects of the code. This paper describes the thermal hydraulic treatment in the code and describes thermal hydraulic comparisons with the RELAP5 mod 3.2 code.

II. MULTIPLE CHANNEL MODEL

In order to provide a detailed thermal hydraulic representation of the reactor core, SAS-DIF3DK uses a multiple channel treatment in which each channel represents one or more assemblies. Detailed three dimensional distributions are provided to the kinetics treatment even though the channel calculations are one dimensional. As illustrated in Fig. 1, a channel contains coolant; and it can also contain one or more fuel pins, duct walls, water pins, control rods, axial shields, or a pressure tube surrounded by graphite. A channel usually runs the full length of a fuel assembly, including the regions above and below the fuel pins. In the RBMK case, a channel also includes inlet and outlet piping. Channels are largely independent of each other,
and currently cross flow between channels is not accounted for. All core channels see common inlet and outlet pressures that drive the coolant flow rates. Thus, transient flow redistribution between channels is automatically accounted for. In the axial direction a channel is divided into zones. In each zone the coolant flow area and hydraulic diameter are constant. Also, within a zone the solids in contact with the coolant do not change geometry.

III. COOLANT TREATMENT

A five equation homogeneous non-equilibrium model is used for the coolant. The liquid and vapor velocities at a given location are assumed to be the same, but the liquid and vapor temperatures can be different. The coolant calculations include single phase liquid, subcooled boiling, nucleate boiling, post departure from nucleate boiling (post-DNB) heat transfer, and pure vapor. Starting from single phase liquid, when the cladding surface temperature exceeds saturation, the Chen correlation\(^2\) is used, with a nucleate boiling term added to a forced convection term in the heat transfer coefficient. For the onset of net void generation, the correlation of Saha and Zuber\(^3\) is used. Either the Biasa correlation\(^4\) or the Groeneveld table lookup\(^5\) is used for the critical heat flux. For the post-DNB region the Dougall-Rohsenow heat transfer correlation is used.

IV. NUMERICAL METHODS

The thermal hydraulics treatment includes both steady-state and transient calculations. For the steady-state calculations, a direct solution is used; and it is necessary to iterate to obtain a consistent solution. For the transient solution fully implicit time differencing can be used so there is no Courant limit on the size of the time steps. The time step size is limited mainly by accuracy considerations, and relatively large time steps can be used unless the quantities being calculated are changing very rapidly. There are no iterations in the transient thermal hydraulics solutions in SAS-DIF3DK. For a transient time step, the equations are linearized about the values at the beginning of the step, and the changes in all temperatures, coolant flow rates, vapor flow qualities, and pressures are solved for simultaneously and directly. This provides an efficient numerical solution.

V. CODE COMPARISONS

In order to do clean thermal hydraulic code comparisons, the 3-D neutron kinetics module in SAS-DIF3DK was not used. Instead, the power was specified as a function of time for both codes.

A single channel model was used for code comparisons. An RBMK fuel assembly was modeled in detail. The model also included the inlet piping from the pressure distribution header to the assembly inlet, plus the outlet piping from the assembly outlet to the steam drum. This model is similar to the model in Fig. 1, but it was modified somewhat so that it would run better in RELAP5. For numerical stability, the time step size in RELAP5 must be limited by a material motion Courant limit: the coolant must not move more than the length of an axial node in a time
step. Thus, small nodes are usually not used in RELAP5 models, although a single small node between large nodes can be tolerated. Numerical stability in the SAS-DIF3D code is not restricted by a Courant limit, so the SAS-DIF3D RBMK model of Fig. 1 contained a number of small (one or two cm long) nodes to represent fuel pin end caps and the axial gap between the upper and lower pins in an RBMK assembly. For this comparison study, the small nodes were lumped in with longer nodes so there are no small axial nodes in the model. The fuel pin end caps were lumped in with the fuel pin gas plenums. The gap between upper and lower pins was moved to the upper pin gas plenum. Thus, the fuel in the upper and lower pins was run together to make one long fuel region. Five axial nodes were used for the inlet piping, three axial nodes for the channel inlet and lower channel, one axial node for the lower gas plenum and end cap, twelve axial nodes for the fuel region, one axial node for the upper gas plenum and end cap, four axial nodes for the upper channel, and three axial nodes for the outlet piping.

Radially a significant amount of detail was included in the model in the fuel pin region. Two fuel pins are modeled. One pin represents the inner row of pins, and the other pin represents the outer row of pins. The outer pin has somewhat more power, since the outer row of pins shields the inner row from thermal neutrons. Also modeled are the pressure tube with the graphite around it, the central rod, and smeared grid spacers.

The first code comparisons were with steady-state results for nominal power and coolant flow rate. SAS-DIF3DK calculates the steady-state solution directly. In contrast, RELAP-5 does not do a direct steady-state calculation. The user must input initial temperatures, coolant flow rates and pressures. Then RELAP5 runs a null transient until the conditions converge to a steady-state. The SAS-DIF3DK steady-state case was run first. Then the SAS-DIF3DK results were fed into RELAP5 as the initial conditions for the start of the null transient.

Two transients were run, starting from the same steady-state. The first transient was a protected reactivity insertion. The power was specified as a function of time for this case. Figure 2 shows the power history. The power rises initially to three times nominal power. Then a scram reduces the power to decay heat levels. For this case the inlet and outlet pressures were held constant during the transient. The second transient was a large break LOCA caused by the rupture of a pressure distribution header. The inlet pressure dropped rapidly to one atmosphere, while the power was held constant.

The steady-state coolant temperatures calculated by the two codes are shown in Fig. 3. The coolant temperatures agree to within a fraction of a degree. The vapor flow quality is shown in Fig. 4. The two codes give almost identical vapor flow qualities. Figure 5 shows the pressure distributions in the core. In the single phase region near the bottom of the core the pressure drops are virtually identical, whereas in the two phase region the SAS-DIF3DK pressure drops slightly faster than the RELAP5 pressure.

SAS-DIF3DK calculates fuel and cladding temperatures at the center of an axial node, using the coolant mid-point temperature for heat transfer from the cladding. In contrast, there is some ambiguity as to where in the axial node the RELAP5 fuel and cladding temperatures are
calculated. In RELAP5, the coolant temperature used in the calculation of heat flow from the cladding to the coolant is the temperature at the top of the node; unless flow has reversed, in which case the temperature at the bottom of the node is used. For the figures in this paper, the fuel and cladding temperatures are plotted at the elevation of the center of the node for both codes.

Figure 6 shows the difference between the cladding surface temperature and the coolant liquid temperature. The abrupt change in slope near an elevation of 1.2 meters is due to the onset of subcooled boiling. The two codes give almost identical results in the single phase region, whereas there is a difference of about 10% in part of the two phase region. Since both codes use the same power in each node, the differences in the two phase region must be due to differences in the nucleate boiling heat transfer coefficient. Both codes use the Chen correlation for the nucleate boiling coefficient, but the Chen correlation is not precisely defined. The Chen correlation uses two functions, an effective two-phase Reynolds number function, F, and a bubble-growth suppression function, S, that are obtained from fits to experimental data. These functions are not given as precise analytic functions; they are presented as graphical functions with experimental scatter indicated on the graphs. The functions must be fit in some form before they are used in a computer code. Since the scatter for these functions is approximately ±50%, and since the functions cannot be read precisely off of the graphs, it would not be surprising if the fits used in different codes differed by 10%.

The fuel center line temperatures for the outer row of pins are shown in Fig. 7. The agreement between the two codes is very good.

For the first transient, the protected reactivity insertion, time step convergence studies were done with both codes. Figure 8 shows the transient SAS-DIF3DK coolant flow rates at the center of the core for different time step sizes. Initially the coolant flow rate drops and reaches a minimum just after 2.0 seconds. The flow rate drop is due to the power rise, which increases the extent of the two phase region and increases the vapor flow quality in this region. The two phase pressure drop is higher than the pressure drop in pure liquid. After the minimum, the coolant flow rate rises as the power drops and the amount of vapor produced decreases. A time step of 0.2 seconds is too large to predict the minimum accurately; but the results for time step sizes of 0.1 and 0.05 seconds are almost identical, indicating that a time step size of 0.1 seconds is adequate. In this case the fuel center line temperature results are insensitive to time step size.

As mentioned previously, the time step size in RELAP5 is limited by numerical stability considerations to a Courant limit for material motion: the coolant should not move more than one axial node in a time step. This limit can be relaxed somewhat for a single small node between larger nodes. In the first transient the maximum transient coolant velocity is about 26 m/s and the size of axial nodes in the core is 0.56667 m, so the Courant limit is 0.56667/26 = 0.022 s. This case was run in RELAP5 with time step sizes of 0.02 and 0.01 seconds. Also, to see what happened when the Courant limit was exceeded, a time step size of 0.04 seconds was used. The RELAP5 vapor flow quality at the top of the core is shown in Fig. 9. In this case, the solution did not become wildly unstable when the Courant limit was exceeded somewhat, but some
accuracy was lost. There were small differences between the results for time step sizes of 0.02 and 0.01 seconds, especially near the peak of the vapor flow quality curve. Again the fuel center line temperature is insensitive to time step size.

The SAS-DIF3DK and RELAP5 results for the first transient are compared in Figs. 10, 11 and 12. For these figures the most accurate results for each code are used: the time step size for the SAS-DIF3DK results is 0.05 seconds, and the time step size for the RELAP5 results is 0.01 seconds. The results from the two codes agree remarkably well.

The second transient, the rupture of a pressure distribution header, was in some respects a much more severe transient. The pressure at the inlet end of the inlet piping dropped from about 70 bars to 1 bar in 0.1 second at the beginning of this transient. This led to temporary flow stagnation followed by a strong flow reversal. The core channel was cooled by downward flow from the drum to the ruptured pressure distribution header.

For the second transient time step convergence studies showed that for the first two seconds of the transient a time step size of 0.001 second or less was needed in RELAP5. For the later part of the transient the Courant limit for core axial nodes was 0.009 seconds. For comparison with SAS-DIF3DK results, a RELAP5 case was run with a time step size of 0.001 seconds for the first two seconds, 0.002 seconds for the next four seconds, and 0.008 seconds from 6 to 20 seconds.

In SAS-DIF3DK the user specifies the maximum main time step size, but the code contains an automatic time step controller that reduces the time steps when quantities are changing rapidly. The code uses a multi-level time step scheme. The main time step is used for power and reactivity calculations, as well as regular edits. A heat transfer time step is used for temperature calculations in fuel pins and other solids. A coolant time step is used for coolant temperature, pressure and flow rate calculations. Heat transfer time steps can not span main time step boundaries, and coolant time steps can not span either heat transfer time step boundaries or main time step boundaries. In the second transient, the power is held constant and the fuel pin temperatures change somewhat slowly, so the code will run with relatively large main and heat transfer time steps. On the other hand, the coolant flow rates and vapor flow qualities change rapidly in the earlier part of the transient; so the code will automatically cut back the coolant time steps to fairly small values for the first part of the transient. The code uses an average coolant time step size of less than 0.001 second in the first 0.01 second of the transient when the inlet pressure is dropping rapidly. From 0.01 second to 0.7 second the code uses an average coolant time step size of less than 0.01 second. After 3 or 4 seconds, the code can use a large coolant time step size. Even though the code will run with a coolant time step size much smaller than the heat transfer time step size, the answers tend to be more accurate if the coolant time step size and the heat transfer time step size are in the same ball park. Therefore, for comparison with RELAP5, a SAS-DIF3DK case was run with a main time step size of 0.001 second for the first 0.1 second, 0.01 second from 0.1 to 3.0 seconds, and 0.1 second after 3.0 seconds.
The results from the two codes for the rupture of a pressure distribution header are given in Figs. 13 - 17. The coolant flow rate behavior is qualitatively similar, but in the later part of the transients the flow rates given by the two codes differ by about 15%. The behavior of the coolant pressures and vapor flow qualities are also similar. The differences in coolant flow rates and pressures predicted by the two codes may be partly due to the use of a homogeneous coolant model in SAS-DIF3DK whereas a two fluid six equation model is used in RELAP5, allowing for different liquid and vapor velocities. On the other hand, after the initial rapid change part of the transient the RELAP5 vapor velocities are almost the same as the liquid velocities at most axial nodes. Thus, results similar to a homogeneous model would be expected.

The big differences between the transient results of the two codes are the cladding and fuel temperatures shown in Fig. 17. Both codes predict that in much of the core the critical heat flux is exceeded early in the transient when the coolant flow rates go through zero and the vapor flow quality becomes high. This is not because the heat flux increases; it is because the critical heat flux decreases as the vapor flow quality increases. After the departure from nucleate boiling, the cladding temperatures increase rapidly; so when the coolant flow rates become large and negative and when the vapor flow quality comes back down the cladding is too hot to reestablish nucleate boiling. The fuel temperatures rise because of the increase in the cladding temperatures. The RELAP5 cladding temperatures rise more than twice as much as the SAS-DIF3DK temperatures after the departure from nucleate boiling, indicating that the RELAP5 post-DNB heat transfer treatment is more conservative than that used in SAS-DIF3DK. The differences between the two codes are due to two factors; different heat transfer coefficients are used for post-DNB conditions, and the two codes use different heat transfer between the liquid and the vapor in the post-DNB region. For this transient the RELAP5 vapor temperature in the middle of the core gets significantly higher than the liquid temperature, whereas in SAS-DIF3DK the vapor temperature stays near the saturation temperature. SAS-DIF3DK uses the Dougall-Rohsenow correlation for the post-DNB heat transfer coefficient, whereas RELAP5 uses the Chen-Sundaram-Ozkaynak correlation for transition boiling and a modified Bromley correlation for film boiling. Reference 9 discusses post-DNB heat transfer. According to this reference, the Dougall-Rohsenow correlation is conservative for boiling water conditions, and this correlation greatly underpredicts post-DNB heat transfer for vapor flow qualities less than 10%. This reference also discusses the heat transfer between the vapor and the liquid. Two limiting cases are discussed: the thermal equilibrium model, with good heat transfer between phases and the vapor temperature staying at saturation, and the frozen droplet model, with no heat transfer between phases. For this case the SAS-DIF3DK results are close to the thermal equilibrium model, whereas the RELAP5 results are closer to the frozen droplet model. For high coolant flow rates the thermal equilibrium model is close to the experimental data, but this model is conservative. For low coolant flow rates the frozen droplet model is closer to the experimental data, and it is conservative. After the first second of this transient the flow rates are high, so the SAS-DIF3DK results must be conservative.
COMPUTER RUNNING TIME

One of the goals in the development of SAS-DIF3DK is to provide a detailed thermal hydraulic treatment that can run in real time. This may require the use of multiple processors for large problems, with one or more channels running on each processor. It is highly desirable that the numerical procedures in the code be efficient enough that at least one channel can run in real time on a single processor.

Table 1 lists the computer running times for the steady state and for the two transients on a single processor Sun Ultra 60 workstation. The speed of this computer is typical of currently available workstations. For the first transient a time step size of 0.1 seconds gave accurate results in SAS-DIF3DK and this case ran 10.8 times as fast as real time. RELAP5 required a time step size of 0.02 seconds for this transient and ran essentially in real time. For the second transient, SAS-DIF3DK ran 3.2 times as fast as real time. SAS-DIF3DK uses less computing time per step, but the main reason for the improved performance of SAS-DIF3DK is larger time steps and therefore fewer time steps. Mainly because of the material motion Courant limit, RELAP5 requires 5-6 times as many steps as SAS-DIF3DK to calculate the same transient.

Table 1, Computer Running Times

<table>
<thead>
<tr>
<th>Case</th>
<th>Code</th>
<th>time step size (s)</th>
<th>CPU time (s)</th>
<th>simulation time (s)</th>
<th>time steps</th>
<th>CPU/step (s)</th>
<th>speed/real time</th>
</tr>
</thead>
<tbody>
<tr>
<td>steady state</td>
<td>SAS-DIF3DK</td>
<td>.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>steady state</td>
<td>RELAP-5</td>
<td>.005</td>
<td>35.4</td>
<td>9.325</td>
<td>1865</td>
<td>.0190</td>
<td>0.26</td>
</tr>
<tr>
<td>transient 1</td>
<td>SAS-DIF3DK</td>
<td>.05</td>
<td>6.9</td>
<td>40</td>
<td>842</td>
<td>.0082</td>
<td>5.80</td>
</tr>
<tr>
<td>transient 1</td>
<td>SAS-DIF3DK</td>
<td>.10</td>
<td>3.7</td>
<td>40</td>
<td>437</td>
<td>.0085</td>
<td>10.81</td>
</tr>
<tr>
<td>transient 1</td>
<td>SAS-DIF3DK</td>
<td>.20</td>
<td>1.9</td>
<td>40</td>
<td>229</td>
<td>.0082</td>
<td>21.10</td>
</tr>
<tr>
<td>transient 1</td>
<td>RELAP5</td>
<td>.01</td>
<td>76.3</td>
<td>40</td>
<td>4000</td>
<td>.0191</td>
<td>0.52</td>
</tr>
<tr>
<td>transient 1</td>
<td>RELAP5</td>
<td>.02</td>
<td>41.0</td>
<td>40</td>
<td>2000</td>
<td>.0205</td>
<td>0.98</td>
</tr>
<tr>
<td>transient 1</td>
<td>RELAP5</td>
<td>.04</td>
<td>21.8</td>
<td>40</td>
<td>1000</td>
<td>.0218</td>
<td>1.83</td>
</tr>
<tr>
<td>transient 2</td>
<td>SAS-DIF3DK</td>
<td>.001 - .1</td>
<td>6.2</td>
<td>20</td>
<td>943</td>
<td>.0066</td>
<td>3.23</td>
</tr>
<tr>
<td>transient 2</td>
<td>RELAP5</td>
<td>.001 - .008</td>
<td>116.4</td>
<td>20</td>
<td>5751</td>
<td>.0202</td>
<td>0.17</td>
</tr>
</tbody>
</table>

1. Time steps for RELAP5, coolant time steps for SAS-DIF3DK.
CONCLUSIONS

Identical steady state and transient cases were run in SAS-DIF3DK and RELAP5. The results from the two codes agreed remarkably well. What differences occurred can mostly be explained in terms of different models or different implementations of models. For the transients considered here, SAS-DIF3DK achieves the developmental goal of real-time simulation. This level of performance comes about because the advanced modeling techniques in SAS-DIF3DK use less computing time per step, but mostly because the time step length in SAS-DIF3DK is not limited by the material motion Courant limit. In comparison to the methods used in codes like RELAP5, SAS-DIF3DK uses fewer time steps to simulate the same transient.

REFERENCES:


Fig. 1, SAS-DIF3DK Model of an RBMK Fuel Assembly and Associated Piping.
Fig. 2, Power for RBMK Protected Reactivity Insertion Case

Fig. 3, Comparison of RBMK Steady-State Coolant Liquid Temperatures

Fig. 4, RBMK Steady-State Vapor Flow Quality Comparison

Fig. 5, Comparison of RBMK Steady-State Core Pressures
Fig. 6, RBMK Steady-State Cladding Surface Temperature - Coolant Liquid Temperature

Fig. 7, Comparison of RBMK Steady-State Fuel Center Line Temperatures

Fig. 8, Time Step Comparisons, SAS-DIF3DK Coolant Flow Rate at Center of Core, Reactivity Insertion Case

Fig. 9, Time Step Comparisons, RELAP5 Vapor Flow Quality, Top of Core, Reactivity Insertion Case
Fig. 10, RBMK Protected Reactivity Insertion, Coolant Flow Rates, Center of Core, SAS-DIF3DK and RELAP5 Results

Fig. 11, RBMK Protected Reactivity Insertion Vapor Flow Quality, Top of Core, SAS-DIF3DK and RELAP5 Results

Fig. 12, RBMK Protected Reactivity Insertion Fuel Center Line Temperature, Center of Core SAS-DIF3DK and RELAP5 Results

Fig. 13, Transient Inlet Flow Rates Calculated by SAS-DIF3DK and RELAP5 for RBMK Pressure Distribution Header Rupture
Fig. 14, Transient Outlet Flow Rate Calculated by SAS-DIF3DK and RELAP5 for RBMK Pressure Distribution Header Rupture

Fig. 15, Mid-Core Flow Quality Calculated by SAS-DIF3DK and RELAP5 for RBMK Pressure Distribution Header Rupture

Fig. 16, Mid-Core Coolant Pressure Calculated by SAS-DIF3DK and RELAP5 for RBMK Pressure Distribution Header Rupture

Fig. 17, Mid-Core Fuel and Cladding Temperatures for RBMK Pressure Distribution Header Rupture