Development Status of SINDA/FLUINT and SINAPS

Brent A. Cullimore and Steven G. Ring
Martin Marietta Astronautics Group
Denver, Colorado

Eugene K. Ungar
NASA Johnson Space Center
Houston, Texas

Summary

SINDA/FLUINT (Systems Improved Numerical Differencing Analyzer / Fluid Integrator, formerly SINDA '85) is a computer code used to analyze thermal/fluid systems that can be represented in lumped parameter form. In addition to conduction and radiation heat transfer, the code is capable of modeling both single– and two–phase flow networks, their associated hardware, and their heat transfer processes. In this paper, recent improvements to SINDA/FLUINT are described, as are those in progress that will be available in the fall of 1992 in Version 2.5. Also, a preview of planned enhancements is provided. This paper also introduces SINAPS (SINDA Application Programming System), a powerful graphical pre– and postprocessor that will also be available in the fall of 1992.

Background

Evolving spacecraft thermal control technology is increasingly utilizing two–phase fluid systems to accomplish waste heat acquisition, transport, and rejection. In the case of the Space Station Freedom, the high heat rejection requirement of 82.2 kW and the typical heat transport distances of over 100 feet made a two–phase thermal control system the only rational choice. A conventional heat pipe or single–phase fluid loop thermal control system, such as have been used in previous US spacecraft, would have had unacceptable weight and power penalties. The heat rejection requirements will be even higher and the transport distances will be even longer for lunar and planetary base applications, again forcing the use of two–phase thermal control systems for those missions.

The introduction of two–phase active thermal control systems required a quantum leap in the development of thermal control technology. A similar development effort was required for the analytical tools for modeling such systems. Previously, there was no single computer tool that was suitable for analyzing spacecraft two–phase systems and components, especially when the requirement was levied to integrate such analyses with vehicle–level simulation tools such as SINDA and TRASYS (Thermal Radiation Analysis System). Typically, two–phase systems and components were analyzed by generating application–unique mathematical modeling equations that were then incorporated into numerical solution computer programs. This method of analysis caused much duplication of effort and hindered the transfer of thermal math models and their ability to be modified by other analysts.

Therefore, in the mid 1980's NASA Johnson Space Center (JSC) launched an effort to develop a design simulation tool that was well suited to modeling two–phase systems for space applications. An effort was already nearing completion at NASA JSC which brought the 1972 version of SINDA up to modern standards, completely reworking it and adding submodels and other capabilities that enhance model integration and exchange. The result of that modernization, called SINDA '85, was used as a starting point for the addition of the new fluid analysis capabilities. The final product, SINDA/FLUINT, is a quantum leap above the older versions of SINDA, featuring a comprehensive single– and two–phase, steady and transient fluid analysis package (FLUINT) that works together with traditional SINDA thermal networks to solve arbitrarily complex thermal/fluid problems. Version 2.3, released in early 1990, has become the most commonly used tool for analysis of fluid flow and heat transfer in space–based systems, and has spread to other specialties (propulsion, environmental control) and even other industries (energy, aircraft, automotive, and architectural) because of its generality, analytic power, transportability, and ability to be customized. In 1991, SINDA/FLUINT was awarded the NASA Space Act Award.

SINDA/FLUINT has been continually updated and enhanced since its first release in the late 1980's. The improvements have made the code even more general in scope, better able to handle different and more difficult problems, and more efficient in its use of computer time. References 1, 2, and 3 describe the capabilities of Version 2.3, which is available through COSMIC, NASA's software distributor. In this paper, the capabilities of the current NASA version (Version 2.4) are described, as well as the work currently being completed by Martin Marietta on Version 2.5, which will be available in the fall of 1992. Improvements planned for future versions are also described.

In 1990, NASA JSC initiated an effort to provide a modern graphical pre– and postprocessor for SINDA/FLUINT. Martin Marietta is currently completing the result of this effort: SINAPS, a powerful graphical interface that will be available in the fall of 1992. SINAPS provides a means for graphically building and maintaining SINDA/FLUINT models, and displaying the results on the sketch the user has created. In this paper, the capabilities of SINAPS are detailed.

SINDA/FLUINT Enhancements

Almost all work since 1985 has focused on the continuing development and expansion of the fluid analysis capabilities; only relatively minor improvements have been made to the thermal analysis code. This section lists and describes the major advances in the FLUINT portion of the code.

There has been a steady accrual of relatively minor expansions and corrections over the years. While collectively these improvements have added significantly to the speed, utility, and ruggedness of the code, they are too numerous and detailed to be described in this paper. Suffice it to say that few users regret the effort required to update their version even if the latest round of major improvements was not directly of interest to them.

Version 2.4 Enhancements

The primary goal of Version 2.4, completed in December 1991 and documented in Reference 4, was to enable the user to selectively avoid the assumption of homogeneous two–phase flow, and to use instead a slip flow formulation. To achieve this goal, various important features had to be added to the code in preparation for the addition of slip flow modeling, such as flow regime mapping. The ability to discern basic flow regimes and to calculate the frictional pressure drop accordingly can be used independently of the slip flow options. Flow regime mapping options are described first, followed by slip flow modeling options.

Flow Regime Mapping Options—Packaged as an optional pressure drop 'correlation,' the user may elect to have a simplified two–phase flow regime predicted for duct segments, with the pressure gradient estimated on the basis of that regime. Output routines have been modified to print the current flow regime if this option is used and the flow is two–phase. Instead, if the flow is single–phase, the Reynolds number is printed instead.

Four generalized (simplified) regimes are recognized, as illustrated in Figure 1: bubbly, slug, annular, and stratified. The first two are considered 'dispersed,' and the latter two 'separated.' The distinction between regimes is based (1) on the liquid and vapor mass fluxes, (2) on the void fraction, (3) on the hydraulic diameter of the line—assumed nearly circular, (4) on the magnitude of a body force (or acceleration) vector and its orientation with respect to the duct, (5) on fluid properties such as densities, viscosities, and surface tension, and (6) in the event no clear determination can be made, on previous flow regimes (i.e., regime boundaries exhibit hysteresis). Flow regime mapping methods identified in Reference 5 were used extensively, although neither exactly nor exclusively.

Bubbly flow occurs at the extremes of low gravities, high liquid mass fluxes compared to the vapor flux, and low void fractions (less than about 0.46), and is characterized by small vapor bubbles entrained in liquid. If the bubbles coalesce due to increased accelerations, decreased liquid mass flux, or increased void fraction, then the slug flow regime will appear. The slug flow regime exhibits large bubbles that nearly span the diameter of the tube, but which are axially separated from each other by liquid. Both the slug and bubbly flow regimes are characterized by relatively little slip flow, approaching true homogeneous flow. In both cases, predicted pressure drops are based on the McAdam's formulation for homogeneous flow. These two regimes are therefore identical for homogeneous passages, but they behave differently if slip flow is modeled, as described later.

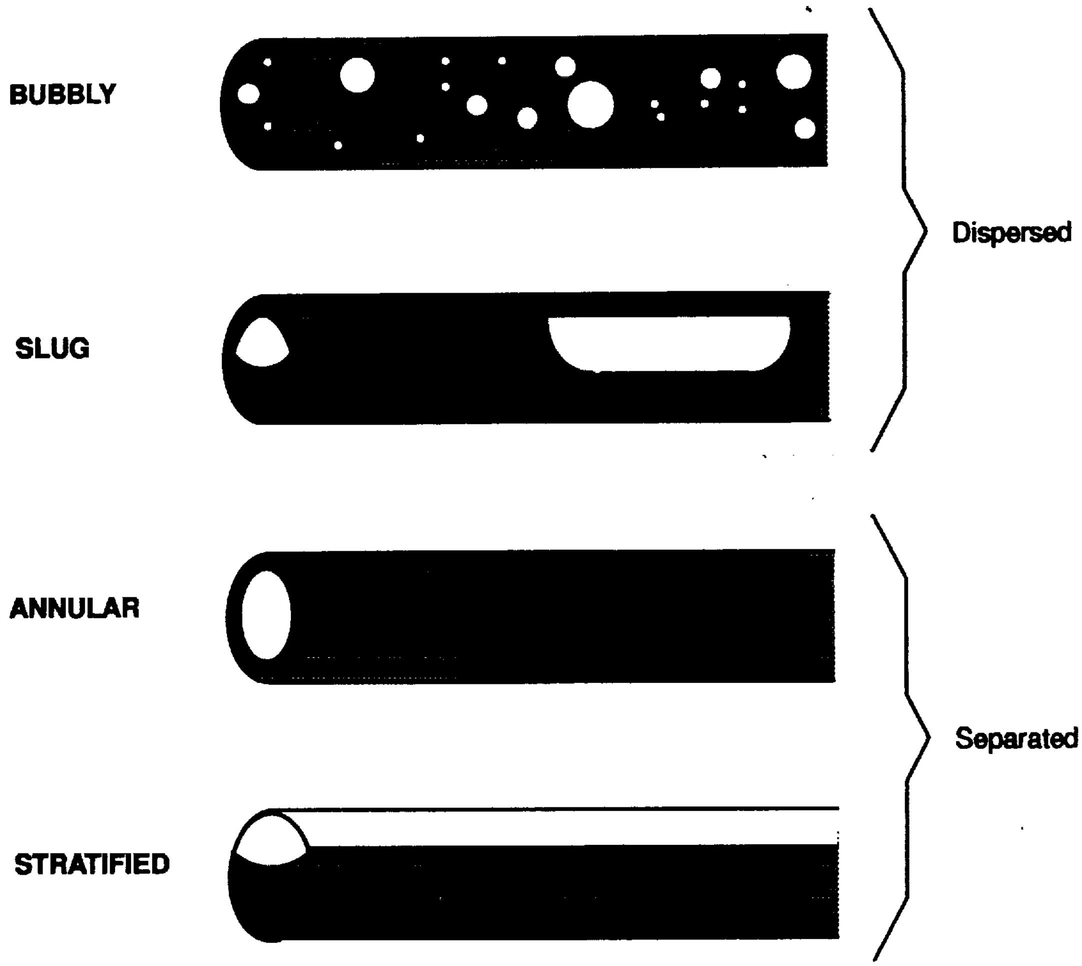

Figure 1 — Simplified Two–Phase Flow Regimes

The annular regime may result if the void fraction continues to grow (above about 0.76), or if the liquid flows downhill, or if there is high enough vapor flux to sustain the uphill flow of liquid. This regime is characterized by a continuous vapor core surrounded and 'lubricated' by a continuous liquid annulus. In most two–phase systems, annular is by far the most common regime. When the regime is determined to be annular, the Lockhart–Martinelli correlation is used.

The stratified regime, characterized by liquid pooling in the bottom of the tube, results if either the vapor mass flux or the liquid fraction is low enough, or the gravity high enough (and the flow is not vertically upward). The stratified regime cannot exist in microgravity. The methods used to predict pressure gradient involve predicting the height of liquid and the fractions of each phase in contact with the wall, assuming a circular cross section (per the method of Taitel and Dukler). Unfortunately, this model is highly sensitive to void fraction, and because the stratified regime typically exhibits the greatest degree of slip, the error in a homogeneous approximation to void fraction can be significant. In other words, pressure drops in the stratified regime are suspect if the default homogeneous options are used. Typically, a homogeneous assumption results in overestimation of pressure drop for stratified flow, whereas if slip flow is modeled as described next, the predicted pressure drop is usually lower than that of all other regimes for the same flow quality.

Slip Flow Modeling Options—By default, a homogeneous assumption is applied in all flow passages, meaning that the vapor velocities and the liquid velocities are assumed equal: there is zero relative velocity or *slip*. With the homogeneous approximation, two–phase flow is modeled as the flow of a mixture of both phases—one momentum equation describes the entire duct segment. This assumption is usually adequate and is both simple to implement and fast to execute. Because of this assumption, there is no difference between thermodynamic quality and flow quality. Thermodynamic quality is the fraction of vapor *mass* within a segment divided by the total *mass* in that segment. Flow quality is the ratio of vapor *mass flowrate* through a segment divided by the total *mass flowrate* through that segment.

In reality, vapor usually moves faster than liquid, and sometimes even in opposite directions. A slip flow formulation takes this into account, using one momentum equation per phase. Slip flow options may be applied to any FLUINT duct segment; the homogeneous approximation is retained for pumps, valves, capillary devices, etc.

Unlike homogeneous flow, with slip flow the thermodynamic quality is no longer the same as the flow quality. Conservation of mass dictates that flow quality must be the same (eventually) whether a homogeneous or slip flow formulation is used. However, the thermodynamic quality is no longer constrained by the homogeneous assumption: it becomes the new degree of freedom necessary to accommodate a new momentum equation. In other words, the thermodynamic quality and its manifestations, such as density and void fraction, will vary as needed to balance the flow forces. Because vapor generally travels faster than liquid, *the predicted void fraction will be smaller with slip flow than with homogeneous flow at the same flow quality*. In other words, more liquid will reside in the line, and the thermodynamic quality will be smaller than the flow quality as depicted in Figure 2 for the stratified regime.

Because most pressure drop and heat transfer correlations are based on flow quality, slip flow and homogeneous formulations predict almost the same steady state as long as flow is cocurrent; the local homogeneous assumption does not affect the overall pressure drop and heat transfer rates. The major difference is the proportion of liquid and vapor in lines. For example, in annular flow a slip formulation predicts typically three to four times as much liquid will reside in a pipe compared to a homogeneous prediction. Of course, this amount is small to begin with, and so quoting a factor of three to four might be misleading.

In transients, the differences can be more dramatic, especially for separated flow regimes where vapor can shift quickly and liquid lags behind. As a specific example, a SINDA/FLUINT model was developed to predict the time it takes to clear a small tube of liquid by

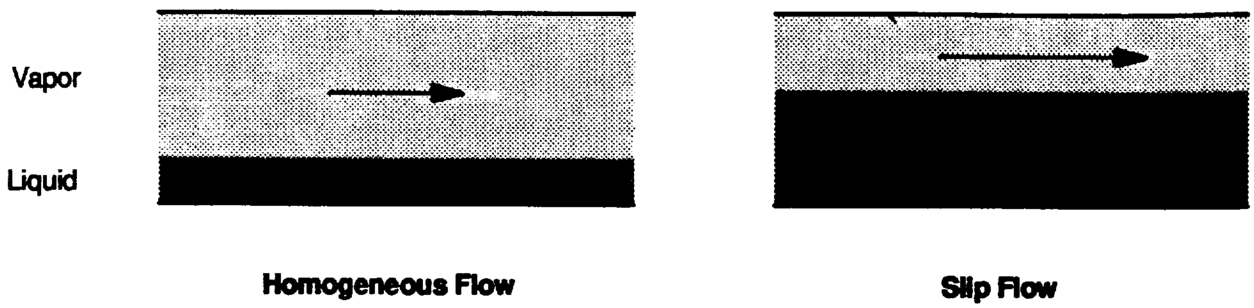

Figure 2 — Homogeneous versus Slip Flow in Stratified Regime at Same Flow Quality

heating it, noting that much more liquid is displaced by generated vapor than is actually evaporated. The default homogeneous assumption resulted in a prediction of 8 seconds to clear the line, whereas allowing slip flow in the same model nearly doubled the duration of the liquid purge event. Since annular flow was quickly established, slip flow allowed the vapor to escape the tube without displacing as much liquid in the process.

This extra degree of modeling power does not come without its price. In addition to greater solution expense, a new layer of uncertainties is revealed. New parameters must be estimated, including (1) the frictional drag between phases, (2) the degree of sharing of inertia, also called added mass and virtual mass, (3) the apportionment of wall friction to each phase, and (4) the momentum transfer associated with phase change. By default, FLUINT will estimate such factors automatically, which requires knowing the flow regime. Hence, flow regime mapping options are defaulted when specifying slip flow. Alternately, like almost all other SINDA/FLUINT options, knowledgeable users can calculate their own coefficients.

Other Improvements—A wide variety of improvements have been implemented to help speed up models utilizing time–dependent fluid elements (called *tanks* and *tubes* in FLUINT) where two–phase flow exist. In general, integrations are smoother, more accurate, and can take larger time steps. Various other improvements have been made in time step predictions, reduced numbers of properties calls, etc., that resulted in speed improvements averaging about 25%.

Also, new simulation options were added to help the user model the mixing of perfect gases, stationary noncondensible gas bubbles, and bellows accumulators. This last option has been applied to other situations requiring two control volumes to share the same physical boundary without exchanging mass.

Because Version 2.5 is to be completed only nine months after 2.4, it was decided that Version 2.4 would be distributed only to NASA centers and Space Station Freedom contractors, and that the next release to COSMIC, who serves a much wider audience, would be delayed until Version 2.5 was completed and tested.

Version 2.5 Enhancements

Three independent improvements have been made in the test version of SINDA/FLUINT Version 2.5, which is scheduled to be officially delivered to NASA JSC in September 1992.

Fast Tabular Fluid Descriptions—It has long been known that one of the significant cost drivers in the solution of fluid is the detail and range of fluid property descriptions. Speed increases can be gained by restricting the description (e.g., providing a liquid–only description) or by simplifying it (e.g., pseudo–perfect vapor equation of state). Also relevant is the fact that ammonia is the most common fluid for two–phase spacecraft thermal management systems. Thus, a new description of ammonia was created that didn't compromise accuracy over the range of temperatures of interest to spacecraft systems, yet runs twice as fast as the built–in ammonia description. This new description uses tabular look–ups, whereas other descriptions describe properties functionally. Once such methods were developed for ammonia, other analogous descriptions were quickly generated for other fluids including hydrogen, nitrogen, oxygen, argon, and ethane.

Single–Phase Heat Transfer with Coarse Discretization—FLUINT slightly underestimates heat transfer for coarsely discretized single–phase lines. This results from assuming an average wall state and an average fluid state over each segment. While such treatment is consistent with the rest of the finite difference (lumped parameter) approximation, which demands nodalization adequate to resolve gradients, it often conflicts with the way many engineers treat a single–phase heat transfer problem: as a constant wall temperature over a segment that has distinct inlet and outlet states. As a result, new heat transfer options have been added to allow such models. For single–phase flows, the predictions are equivalent to a log mean temperature difference (LMTD) solution.

Figure 3 shows how the new methods improve results and/or enable smaller models while yielding the same answers. Comparisons with closed–form solutions are made for this transient thermal/hydraulic analysis of a water pipe with varying inlet temperature and constant wall temperature (Reference 2). To obtain results that are indistinguishable from the closed–form solution, only five control volumes are needed with the LMTD methods compared with twenty for the default downstream–weighted method. Still, the results using traditional methods are good even with only five control volumes. Furthermore, in models of real systems, where gradients in wall temperature or other properties dominate, the differences are usually negligible.

Speeds of Sound and Choking Detection—The user's ability to detect sonic limits was enhanced by providing program options that detect choking in all or portions of a model. The liquid phase remains incompressible, although compressibilities and compressed liquid densities may be calculated and used in concurrent logic, perhaps to calculate effective compliances of control volume walls, or to measure the appropriateness of an incompressible assumption. Other by–products include two–phase speed–of–sound routines.

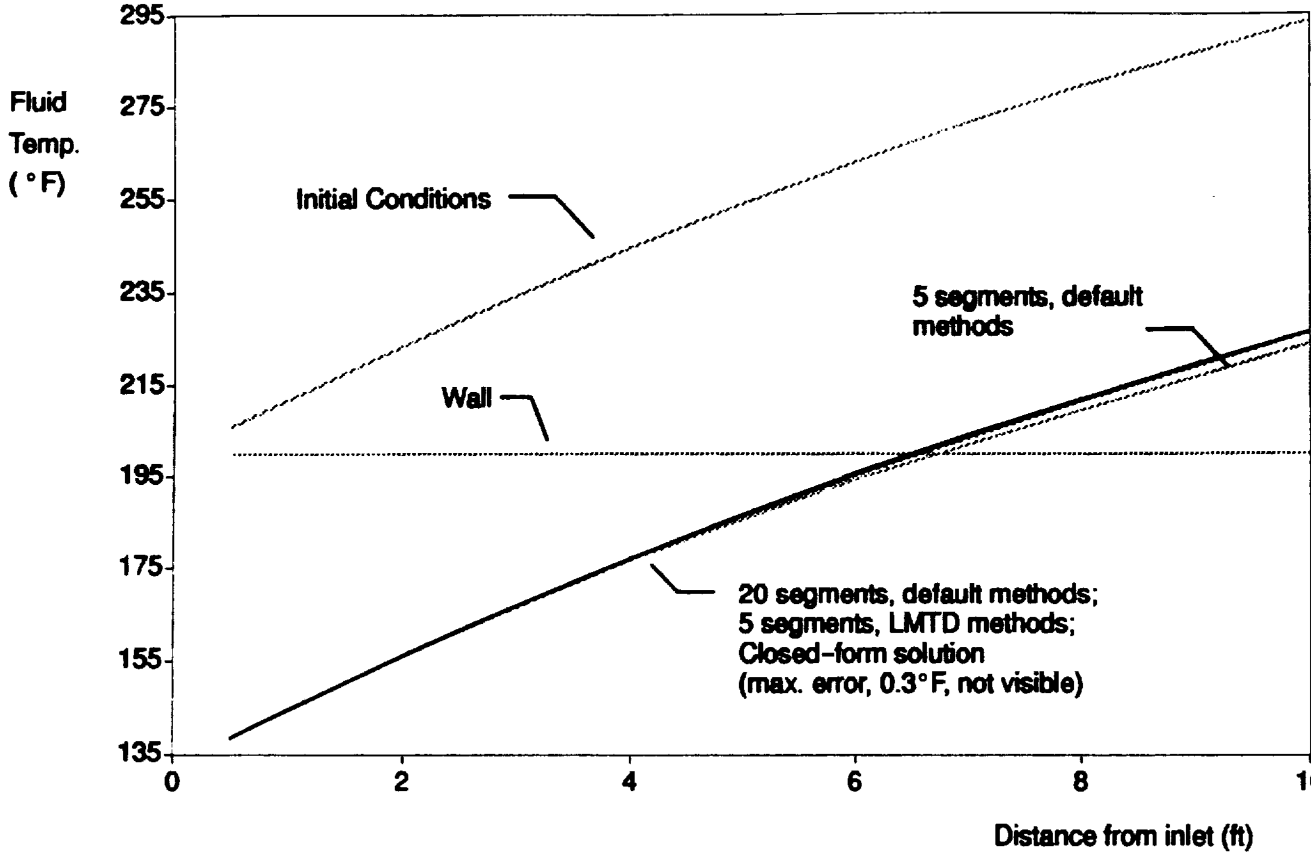

Figure 3 — Comparison of FLUINT Predictions with Closed-form Solution

Other Improvements—Several relatively minor improvements were made in addition to the major thrusts of Version 2.5. These include (1) the addition of K–factors (head loss factors) to duct models, eliminating the need for separate elements to include entrance, bend, and exit losses; (2) the option of using a crash file in addition to normal restart and parametric run options, saving a snapshot of the simulation as often as desired without running out of disk space, (3) reduced memory requirements (matrix inversion work space) for large models, and (4) various internal improvements in time step predictions and slip flow options.

Future Enhancements

Improvements Planned for Version 2.6—FLUINT uses a first–order implicit time step integration that is performed in parallel with whatever method is used to integrate the thermal networks. Heat rates between thermal and fluid models are held constant to conserve energy. If all property domains and derivatives, friction coefficients, heat rates, etc. truly remain constant over the time interval, then the solution is fully implicit and an arbitrarily large time step can be taken. Since these parameters in fact often vary, a best estimate is made of the time step that can be made without excessive changes in such parameters.

Extensive logic is employed to estimate this time step and to check predicted changes against the previous step. While this feedback method successfully avoids time steps that are too small (from the mathematical standpoint if not from the user's standpoint), the only way to be absolutely sure that this estimated time step is not too big is to proceed to integrate the equations and solve for the next network state. If unforeseen changes in operating regimes, boundary conditions, or other parameters are excessive, then at best excessive error will have been generated. At worst, the solution will fail or find a spurious answer such as negative masses in control volumes, or excursions beyond fluid property limits.

In FLUINT, the selected strategy is to spend about 10% to 20% of the cost per solution to make a good and somewhat conservative estimate of the time step. The program is unable to back up and try again if the time step is too big, which fortunately rarely happens. This strategy avoids speed and memory penalties associated with the ability to store and retrieve previous states as well as the problem of trying to measure the generated error and then decide what error is acceptable. A strategy taken in other codes is to take a user–input time step, and then solve iteratively (typically on the order of ten iterations per time step, each about the cost of one FLUINT time step) for the final state. Instead of predicting time steps, the challenge becomes how to converge efficiently on a perhaps elusive final state.

The main thrust of Version 2.6 will be to investigate methods for detecting excessive time steps and correcting them, either by backing up and reducing time steps or by iteratively correcting the solution.

Potential Areas for Future Expansions—Several areas of potential growth have been identified for which no firm development plans exist. These include: (1) full range fluid descriptions with compressible liquid phases that avoid the current discontinuity between saturated liquid and supercritical fluid when the thermodynamic path does not pass through the dome; (2) optionally avoiding the assumption of thermal equilibrium between phases inside of ducts (some limited nonequilibrium capabilities already exist); (3) nonreacting mixtures of substances, especially noncondensible gas phase, air/water systems; (4) higher fidelity capillary models including pore size distributions, wetting hysteresis, partial deprime and liquid recession in the wick; and (5) thermal matrix inversion methods as alternatives to the current iterative closure methods.

SINAPS: SINDA Application Programming System

SINDA/FLUINT, like its predecessors, frees the user from the constraints of real geometry: the model may be limited to a certain volume of material (akin to finite element modeling, or FEM), or it can incorporate a complete vehicle (unlike a finite element approach). The price for this flexibility has been the lack of graphical input and associated postprocessing power, which would help not only in model validation and maintenance, but also in visualization and reporting of results.

Translations to and from solid modeling programs and FEM codes have represented a partial solution for some component design analyses. No analogous capability was present for system–level analyses, or for problems that are intractable with a finite element approach but are amenable to a lumped parameter approach. While postprocessing programs exist to generate X–Y plots of SINDA/FLUINT results, analysts normally communicate with the program via ASCII files. As models grow, the potential for modeling errors or misinterpreted results also grows. (Anecdotally, one small model—a standard sample problem that has been reviewed by many analysts—was found to contain a slight error when rebuilt using SINAPS.)

Nevertheless, hand drawn schematics of SINDA/FLUINT networks are often used to document models in reports and facsimile transmissions. If analysts are able to communicate with each other via such 'artificial' geometry, then it was reasoned they should be able to communicate with the program via similar 2D sketches. After all, similar computer aided engineering packages exist in the electrical design community. Thus, in 1990, NASA JSC initiated an effort to provide a modern graphical pre– and postprocessor for SINDA/FLUINT.

SINAPS is an advanced new companion program to SINDA/FLUINT that enables users to graphically sketch their models using a mouse– and menu–driven user interface. Forms and editing windows exist to satisfy other nongraphic SINDA/FLUINT input requirements. SINAPS then produces complete SINDA/FLUINT ASCII input files, and imports binary output files that were perhaps produced on other machines. This enables *graphical display of predictions on the same schematic used to create inputs.* In addition to pop–up X–Y, polar, and bar plots, features such as "color by flowrate," "thicken by conductance," and "shade by temperature" are supported. Figures 4 and 5 present two sample SINAPS screen images (in black and white for reproduction) that depict some of the features available.

SINAPS is intended to become a complete, modern front–end to SINDA/FLUINT, eliminating the need to communicate via ASCII input and output files. In fact, it contains many powerful features that are unavailable in the basic SINDA/FLUINT system, such as algebraic inputs, shared models, customized components, etc. To assist current SINDA/FLUINT users in the transition to SINAPS, it will accept existing ASCII input files, and will work interactively with the user to produce a graphical depiction of that model.

SINAPS is transportable. It was developed simultaneously on a Macintosh II and a SUN SPARCstation, and can be rehosted on most other workstations, operating systems, and windowing systems. Perhaps more importantly, a SINDA/FLUINT model (and its graphical depiction) built using SINAPS can be easily moved from one type of machine to another, allowing analysts to build models on whatever machines are available, even if that availability changes from day to day. Combining this feature with the fact that SINAPS and SINDA/FLUINT need not reside on the same machine gives the analyst tremendous flexibility.

SINAPS will be available in the fall of 1992, and will correspond to SINDA/FLUINT Version 2.5.

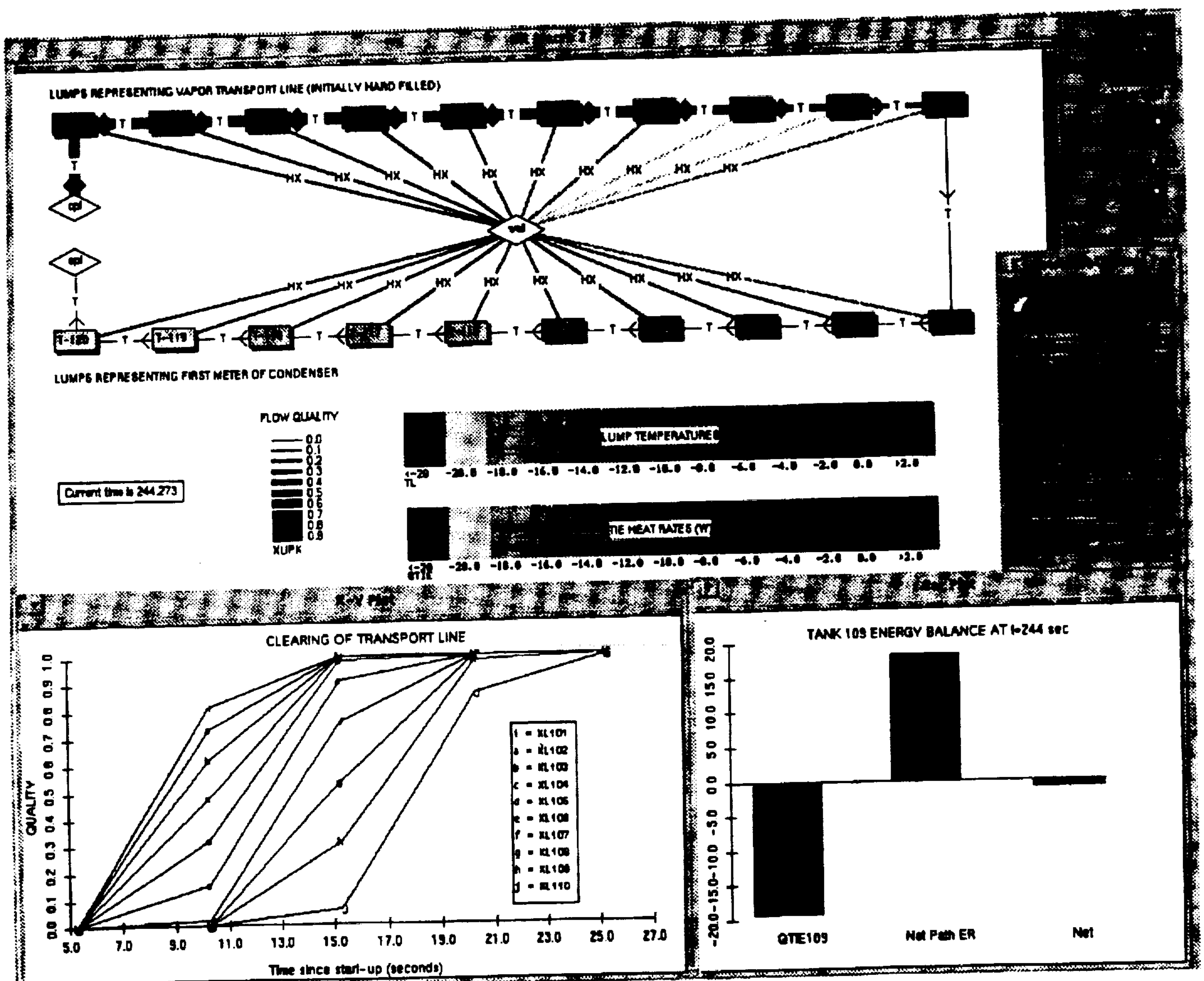

Figure 4 — Sample SINAPS Screen: Capillary Pump Loop Start-up Transient

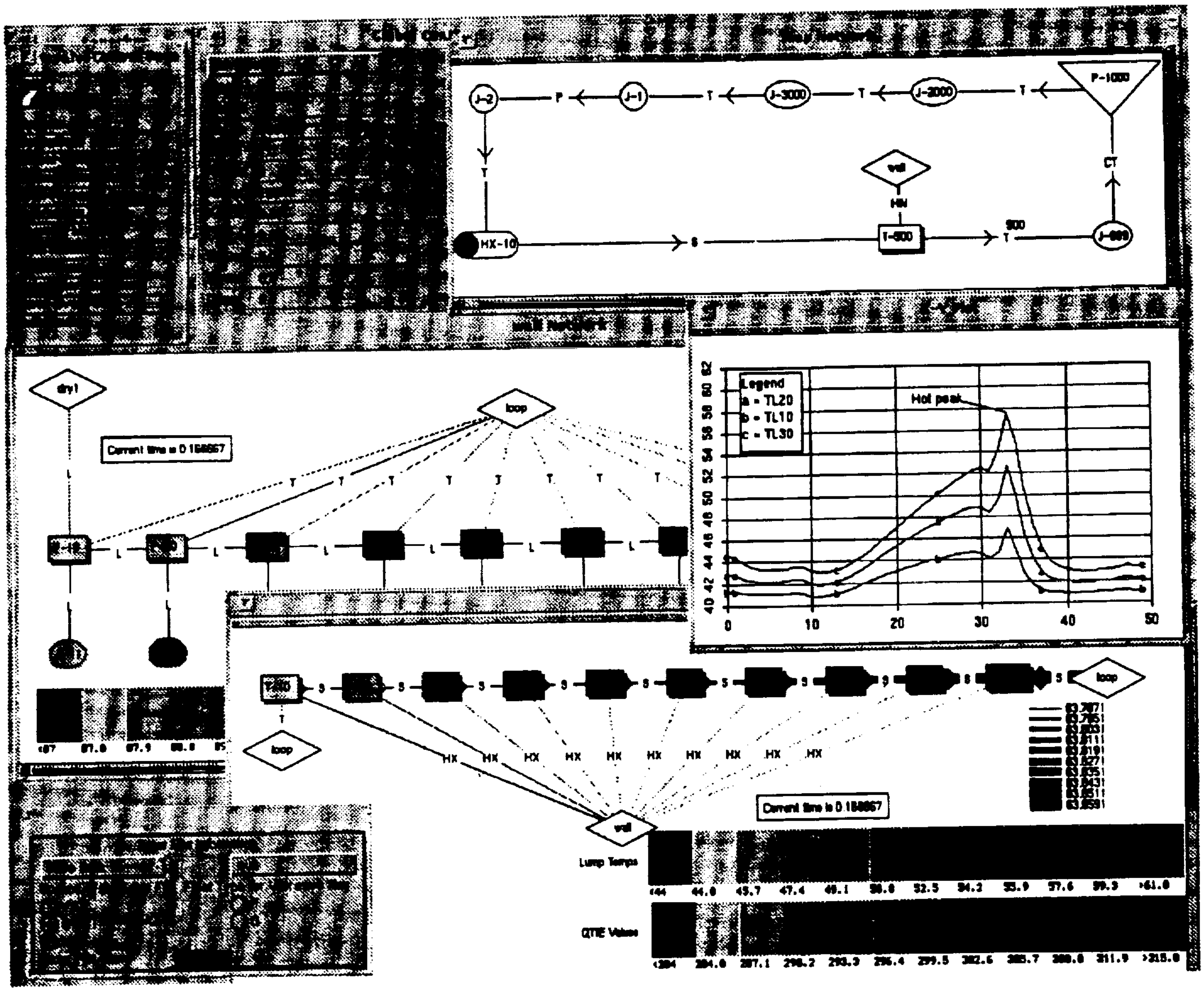

Figure 5 — Sample SINAPS Screen: Pump Instabilities with Temperature Control Valve in Series

Acknowledgements

Neither SINDA/FLUINT nor SINAPS would exist were it not for the continuing support of the NASA Johnson Space Center. These codes were produced by Martin Marietta under contracts NAS9–17053, NAS9–17448, and NAS9–18411.

References

1. Cullimore, B.A.; and Lin, C.H.: FLUINT: Generalized Fluid System Analysis with SINDA '85. AIAA–87–1466; AIAA 22nd Thermophysics Conference, June 1987.

2. Cullimore, B.A: Applications of a Generalized Thermal/Hydraulic Simulation Tool. AIAA–89–1754; AIAA 24th Thermophysics Conference, June 1989.

3. Cullimore, B.A.; Ring, S.G.; Goble, R.G.; and Jensen, C.L.; SINDA '85/FLUINT Systems Improved Numerical Differencing Analyzer and Fluid Integrator Version 2.3, User's Manual; MCR 90–512, Contract NAS9–17448; March, 1990.

4. Cullimore, B.A.; Ring, S.G.; Goble, R.G.; and Jensen, C.L.; SINDA/FLUINT Systems Improved Numerical Differencing Analyzer and Fluid Integrator Version 2.4, User's Manual; MCR 91–1393, Contract NAS9–18411; December, 1991.

5. Crowley, C.J.; and Izenson, M.G.; Design Manual for Microgravity Two–Phase Flow and Heat Transfer; AL–TR–89–027; October, 1989.