

Parametric Thermal Analysis and Optimization Using Thermal Desktop™

Timothy D. Panczak and Brent A. Cullimore

C&R Technologies
www.crtech.com

Copyright © 2000 Society of Automotive Engineers, Inc.

ABSTRACT

Thermal analysis is typically performed using a point design approach, where a single model is analyzed one analysis case at a time. Changes to the system design are analyzed by updating the thermal radiation and conduction models by hand, which can become a bottleneck when attempting to adopt a concurrent engineering approach. This paper presents the parametric modeling features that have been added to Thermal Desktop™ to support concurrent engineering. The thermal model may now be characterized by a set of design variables that are easily modified to reflect system level design changes. Geometric features, optical and material properties, and orbital elements may all be specified using user-defined variables and expressions. Furthermore, these variables may be automatically modified by Thermal Desktop's optimization capabilities in order to satisfy user-defined design goals, or for correlating thermal models to test data. By sharing the set of design variables among analysis models spanning multiple disciplines, further integrated analysis and design may be accomplished. The framework into which Thermal Desktop is embedded in order to support an integrated Thermal/Structural/Optical design, analysis, and optimization system is also presented.

INTRODUCTION

Thermal Desktop is a graphical, CAD-based front end to the NASA standard thermal analyzer Sinda/Fluint [1]. Sinda/Fluint is tightly integrated within Thermal Desktop and is automatically launched when a steady-state or transient thermal analysis is performed. Thermal Desktop's Case Set feature [2] organizes the calculation of

radiation conductors, orbital heating rates, generation of conductors and capacitances, construction of the Sinda/Fluint input deck, compiling and execution of Sinda/Fluint, and the display of temperature results in color on the thermal model as a single "one button" operation.

A revolutionary change was made to Sinda/Fluint with the incorporation of parametric and optimization features in version 4.0 [3]. Dynamic, user defined variables (registers) were incorporated into Sinda/Fluint which allows the thermal model to be parameterized in any desired way. Mathematical expressions referencing these variables may be used in place of the usual numerical input. Furthermore, these variables are dynamic, meaning that they may be updated and their effect propagated throughout the thermal model during the run-time calculation of steady-state or transient results.

The user may change these design variables manually to generate a new design, or they may be turned over to a sophisticated optimizer (the solver) incorporated within Sinda/Fluint. The solver will automatically adjust the values of the requested design variables, within specified constraints, to achieve a user defined goal (minimum weight, minimum deviation from test results, etc.).

The addition of dynamic registers and the solver module to Sinda/Fluint provided a new mechanism for performing sensitivity studies, "what-if" parametric analyses, and automated correlation to test data. However, thermal model data computed outside of Sinda/Fluint, such as orbital heating rates and radiation exchange factors, were not amenable to the same type of parameterization. Although the user could parameterize the resulting heating rate arrays and radiation conductor values, the high level geometry and optical properties that produced the data could not be directly parametrized. This limitation

has been removed by recent enhancements to Thermal Desktop and Sinda/Fluint.

Four important extensions to Thermal Desktop and Sinda/Fluint were implemented to allow dynamic update of Thermal Desktop generated data directly within a Sinda/Fluint run:

1. The ability to define register variables and mathematical expressions within Thermal Desktop. These expressions may be used in place of numerical input for surface parameters, optical and material property values, assembly translations and rotations, and orbital parameters.
2. The implementation of a remote command interface that allows external programs, such as Sinda/Fluint, to pass changes to variables back to Thermal Desktop and to direct Thermal Desktop to perform desired computations. This remote command interface is also used by external optimizers for multi-discipline optimization.
3. The addition of Sinda/Fluint solver data to the Case Set Manager to support “one button” optimization from within Thermal Desktop.
4. The ability to dynamically update recomputed thermal network data (nodal capacitances, conductors, heat source data, array data, etc...) during a Sinda/Fluint execution.

These extensions provide a new level of automating thermal analyses. For example, the Sinda/Fluint solver can automatically vary unknown optical property values, with resulting heatrates and radiation exchange factors updated on the fly by Thermal Desktop, in order to correlate the thermal model to test or operational data.

Optimizations such as the location of boxes on radiator panels, the spacing of heat pipes on radiators, and finding minimum weight thermal structures that meet gradient requirements for thermal distortion can now be automated. Since the Sinda/Fluint solver also has access to orbital parameters, the solver may also be directed to find orbital beta angles that produce worse case hot and cold temperature extremes. Virtually any aspect of a thermal model may now be parameterized and used as the target for optimization.

The user may explicitly add calls in the Sinda/Fluint input deck to direct Thermal Desktop to regenerate data based on changes in register variables, or the execution of Sinda/Fluint may be treated as a “black box” and directed entirely within the Thermal Desktop environment. Thermal Desktop’s Case Set Manager has been extended to include solver input data such as constraint data, design

variables, and the objective function.

The following section discusses the use of the parametric and optimization features in Thermal Desktop. A description of the user interface along with the a discussion of the implementation is presented.

MAIN SECTION

Thermal Model Parameterization and Optimization

A symbolic expression manager, similar to Sinda/Fluint registers, has been added to Thermal Desktop. The manager, shown in Figure 1, allows a variable to be defined as an expression, ranging from a single numerical value to a complicated mathematical expression. These expressions may reference other variables and scientific functions such as sine, cosine, log, etc. Conditional C-style tertiary expressions are also supported, allowing if-then-else construction in the expressions.

The manager displays the list of symbols in alphabetical order, along with the expression and its evaluated result. Variables may be edited by double-clicking on the name in the list, or by selecting the “Edit” button.

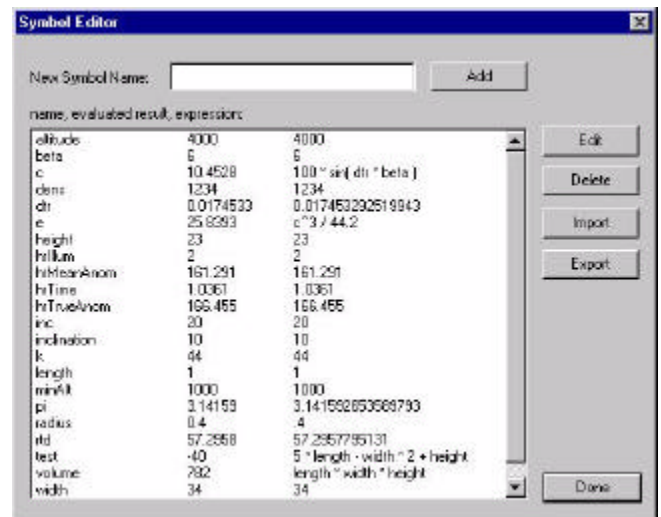


Figure 1. Thermal Desktop’s Symbol Manager

The set of variables and expressions may be exported and imported to and from text files. Each variable also contains a comment field so that the user may enter a useful description of the variable for documentation purposes.

A Thermal Desktop model is parameterized by using variables (or expressions of variables) in the input fields of forms that ordinarily accept numerical input. Double-clicking on an input field (such as the length of a rectangle in the surface edit form) will invoke the expression editor. A symbolic expression may be entered, which when evaluated, will be the value of the input field. The

symbol manager is accessible from the expression editor so that variables may be defined on the fly, or so that the values of other variables may be examined.

When "OK" is selected on the expression editor, the expression is evaluated and the numeric result appears in the input field. The field is set to bold type so that the user can tell that the value was derived from an expression, rather than input directly. The expression may be edited at any time by double-clicking on the input field. If the user attempts to enter a numerical value in the input field, a warning box will appear indicating that an expression is associated with the input. The user may then edit the expression, or delete it and continue with the direct numerical input.

This functionality was provided by deriving a new class from the standard MFC CEdit control, and adding an additional message handler for the double-click operation. An example showing the parameterization of a rotation value for an assembly is shown in Figure 2.

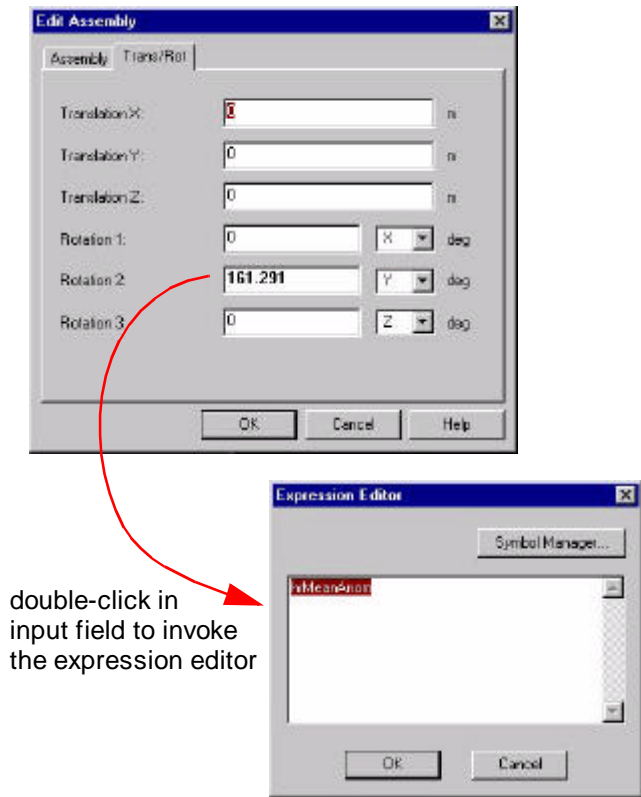


Figure 2. Entering a symbolic expression in an input field

Surface parameters, thicknesses, assembly rotations and translations, material and optical properties, and orbital parameters may reference user defined expressions. In addition, Thermal Desktop defines the calculation-time symbols "hrTime", "hrMeanAnomaly", and "hrTrueAnomaly" during heating rate and variable geometry radk calculations. This allows complicated orbital

maneuvers to be programmed, or any other parameterization as a function of orbital position.

After the model has been parameterized, the user may make sweeping changes to the model by changing the values of the variables using the Symbol Manager. An entire geometric model may be parameterized using a single length variable, for example, causing many surfaces to be updated by changing just one variable. In addition to manually changing values of variables, the user may turn over any set of variables to be controlled by the Sinda/Fluint solver for automated optimization and data correlation.

Extensions to the Case Set Manager have been added to define design variables, constraints, and the goal for the optimizer to seek. The Case Set Manager form is shown in Figure 3. Checking the "Use Optimize/Dynamic Sinda" box will construct a Sinda/Fluint run that includes the necessary inputs to the solver and also enables communication between Sinda/Fluint and Thermal Desktop during Sinda/Fluint execution.

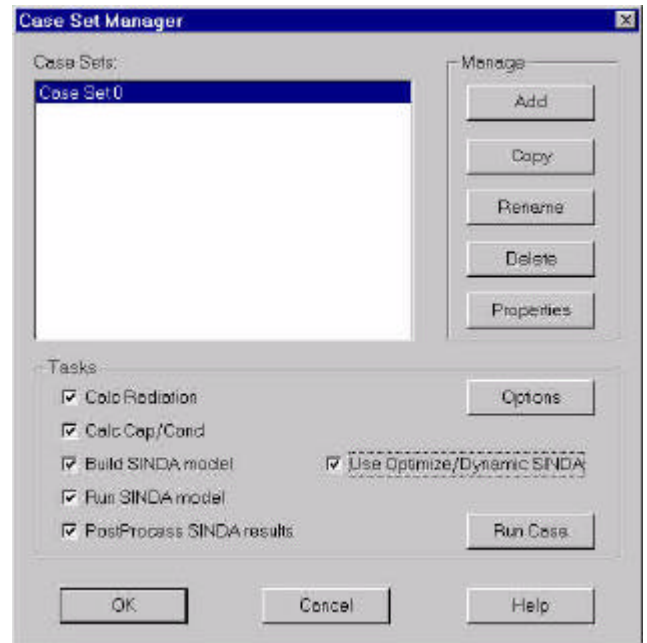


Figure 3. Case Set Manager with optimizer option

Any number of cases can be defined, each case consisting of a series of tasks. The tasks can be one or more radiation exchange factor calculations, one or more orbital heating rate calculations, computation of conductors and capacitances from geometry, the building and execution of a Sinda/Fluint run, and the postprocessing of results back onto the geometric model in color or as X-Y plots. Selecting the Properties button for a case invokes a form that lists the tasks to be performed and allows specific tasks to be customized. The Case Set

Properties form for a case is shown in Figure 4, with the Optimization tab selected. This input page allows details of the optimization to be specified.

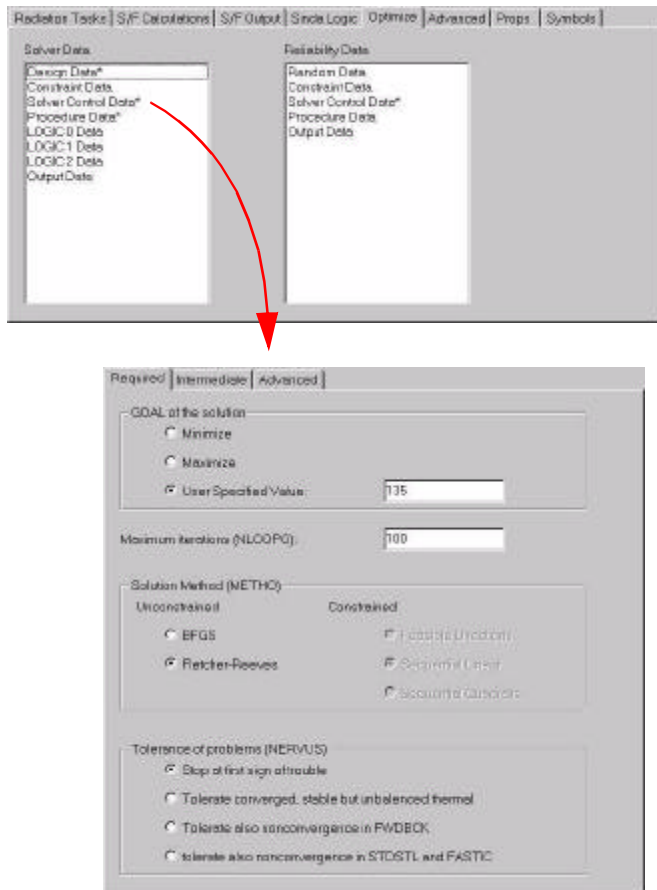


Figure 4. Specifying optimization control values

Double clicking “Design Data” in the list box on the Optimization page will invoke a form that allows the user to specify which variables should be under the control of the optimizer. The “Constraint Data” selection allows the user to specify the allowable range that a particular design variable can assume. Selecting “Control Data” invokes the form shown in the lower part of Figure 4.

This form allows the user to specify what type of goal seeking is desired, such as minimizing, maximizing, or targeting a specific value. Solution methods and convergence conditions may also be specified.

The optimizer will adjust the specified design variables, within the constraints, to achieve the desired goal. The goal is specified by the variable “OBJECT”. This variable contains the result of the objective function. The objective function is defined by the user and may be as simple or complicated as desired. For example, suppose it is desired to find the location on a circuit board for a chip such that its temperature is minimized. Two design vari-

ables would be defined, say XPOS and YPOS, that would define the location of the chip. The Thermal Desktop geometry is parameterized with these variables such that the chip moves on the circuit board according to the values of these variables. The chip is connected to the circuit board using Thermal Desktop’s automatic contact conductance calculations, allowing the chip to be connected to different nodes on the circuit board as its position is varied. The conductors are automatically computed based on the specified contact coefficient and amount of area contact.

The user would specify that XPOS and YPOS are design variables, subject to the constraints of the dimensions of the board, or it’s allowable placement envelope. The goal type would be a minimization, and the objective function would simply be the temperature of the chip.

The “Procedure Data” selection on the Optimize page shown in Figure 4 invokes a form that lets the user specify exactly what is to be done to compute the value of the objective function. Figure 5 shows the procedure data input for this example.

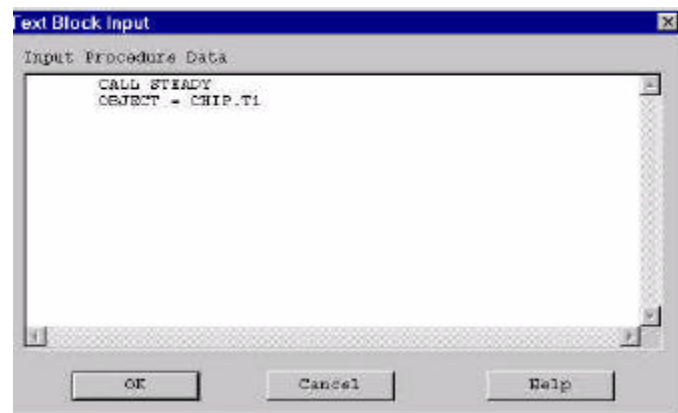


Figure 5. Defining the procedure and objective function

The procedure in this case is to perform a steady state analysis. The objective function is to simply set the value of OBJECT to the temperature of the chip. For more complicated optimizations, OBJECT might be set to the mass of the system, the average temperature deviation from test results, or perhaps total system cost. The procedure may be as simple as the steady state case used here, or may be a series of entirely different transient runs, for example when simultaneously correlating to a number of different test conditions.

The run starts with the procedure block being called using initial values of XPOS and YPOS. After OBJECT is computed, the Sinda/Fluint solver will vary the values of XPOS and YPOS, call the procedure block again, and examine the value of OBJECT. The solver continues this

process, making intelligent decisions on how to change XPOS and YPOS such that OBJECT will be driven to the desired goal.

After the Sinda/Fluint solver updates the values of the design parameters, it sends them to Thermal Desktop via the remote command interface. It then tells Thermal Desktop to recompute data that is dependent on these variables and waits for Thermal Desktop to finish. When Thermal Desktop is finished with calculations, Sinda/Fluint dynamically updates the thermal network with the newly computed values, and continues on to the procedure step.

The entire process is started by selecting the “Run Case” button on the Case Set Manager form. After building the initial input deck and launching Sinda/Fluint, Thermal Desktop waits for subsequent commands from Sinda/Fluint. When a set of changes to design variables are passed back to Thermal Desktop, it examines these variables to see what kind of tasks need to be recomputed.

Thermal Desktop maintains the dependencies between variables and the tasks that must be recomputed. For example, if only a change in a material property, such as conductivity, is the result of the new set of design variables, then only the tasks that are dependent on that change are recomputed. In this case, only linear conductors are recomputed. Radiation tasks would not have to be recomputed for this set of design variable changes. This greatly improves the efficiency of the optimization by avoiding unnecessary recalculations.

The link between Thermal Desktop and external applications is implemented by incorporating a Component Object Model (COM) [4] object internal to Thermal Desktop. This object allows two-way communication between Thermal Desktop and any external application that uses the interface object.

This interface object is used by Sinda/Fluint to enable communication between the two applications. The Sinda/Fluint solver automatically uses the interface to update Thermal Desktop with new values of design variables and to direct Thermal Desktop to recompute necessary data, however, the user may use this interface at any time during a Sinda/Fluint run with explicit calls.

For example, modifications that need to be made in real time can be simulated by asking Thermal Desktop to recompute data at specified transient time points in the VARIABLES logic blocks. Virtually any type of optimization or parameterization can be accomplished with this new architecture.

Multi-Discipline Parameterization and Optimization

The COM object created to allow other applications to communicate with Thermal Desktop may be used to facilitate multi-discipline optimization spanning thermal, structural, and optical models. In this case, the activities of Thermal Desktop are directed by a high level optimizer that controls a common set of design variables used by all three disciplines.

The advantages of combining disciplines for optimization are significant. In a combined analysis, the design parameters may be optimized directly to meet desired performance goals, such as optical wavefront error, with minimum weight or minimum cost objective functions. When the process is separated, the high level performance requirements must be flowed down into suitable structural (resonance frequencies, stiffnesses, distortion limits) and thermal requirements (maximum temperature gradients within structural elements). In addition, it is often unclear how to partition optical performance requirements into suitable structural and thermal requirements.

For example, the thermal analyst is often given a specific temperature gradient that cannot be exceeded. This derived requirement may not be the optimum choice and usually leads to overdesign of the system. The cost of the system engineering effort to partition and negotiate requirements among different engineering organizations can also be significant.

Often the high level optical performance specifications are unclear and subject to negotiation themselves. Parameterizing the system as a whole allows many high level trades to be considered without the effort to flow down requirements at each trade point. Optimization over the full up system ensures that requirements are met with a minimum of overdesign.

The first step to enabling a multi-discipline approach was to provide mechanisms for exchanging data among the optical, structural, and thermal analysis codes. The analysis codes chosen are Thermal Desktop, Sinda/Fluint, NASTRAN, and Optical Research Associates' Code V[®].

The optical codes require distortions in the form of grid displacements or Zernicke polynomials from the structural codes, and possibly temperatures for lenses that are made of a material with an index of refraction that is temperature dependent from the thermal codes. Structural codes require temperatures and temperature gradients to compute distortions. In extreme cases, the thermal model may have to be updated based on structural distortions. This first step in facilitating the communi-

cation between analysis codes has already been accomplished [5,6].

The second step, parameterization all of the models under a common set of design variables, and allowing the separate analysis codes to be guided by an external optimizer is currently in progress. The thermal parameterization and optimization portion has been accomplished, and the remaining tasks are scheduled to be completed by Q4 2000.

CONCLUSION

Thermal Desktop's parametric features, coupled with the optimization capabilities of Sinda/Fluint, provide a new and revolutionary capability for the design and analysis of thermal systems. In addition to unknowns in conduction and thermal capacitance, thermal models with unknowns in radiation exchange and orbital heating can now be automatically correlated to test data within a Sinda/Fluint run. Furthermore, designs may be automatically synthesized that are optimized over all thermal parameters, including parameters that affect radiation and orbital heating. Parameterization may occur at the thermal network level, or at a higher geometric level. This new capability eliminates the tedious and error-prone work of repetitive point design and merges thermal analysis into the concurrent design environment.

The integration of Thermal Desktop and Sinda/Fluint, with their new parametric features, into a multi-discipline analysis and optimization environment will also provide an unprecedented level of integrated analysis.

ACKNOWLEDGMENTS

This work was supported in part by NASA Goddard Space Flight Center under contract NAS5-99165, with Mr. Jeff Bolognese serving as contract monitor.

REFERENCES

1. Panczak, T., Ring, S., and Welch, M., "A CAD-based Tool for FDM and FEM Radiation and Conduction Modeling," SAE paper 981577, 28th International Conference on Environmental Systems, July 1998.
2. Welch, Mark J. and Panczak, Tim, "Automating Thermal Analysis with Thermal Desktop," SAE paper 1999-01-2156, 29th International Conference on Environmental Systems, July 1999.
3. Cullimore, Brent A., "Optimization and Data Correlation, and Parametric Analysis Features in SINDA/FLUINT version 4.0," SAE paper 981547, 28th International Conference on Environmental Systems, July 1998.
4. Rogerson, Dale, Inside COM, Microsoft's Component Object Model, ISBN 1-57231-349-8, Microsoft Press, Copyright 1997.
5. Panczak, T., "Integrating Thermal and Structural Analysis with Thermal Desktop", SAE paper 1999-01-2126, 29th International Conference on Environmental Systems, July 1999.
6. Cullimore, Brent and Panczak, Timothy, "Integrated Design and Optimization of Thermal/Structural/Optical Systems", Quarterly report #4, April 18, 2000, NASA Goddard Space Flight Center Contract NAS5-99165.

CONTACT

Further information on Thermal Desktop and Sinda/Fluint can be found at www.crtech.com or by contacting info@crtech.com. The authors may be contacted at tim@crtech.com and brent@crtech.com.