

Automated Radiation Modeling for Vehicle Thermal Management

Allen R. Curran
Keith R. Johnson
Eric A. Marttila
Keweenaw Research Center
Michigan Technological University
Houghton, MI 49931

Scott P. Dudley
Ford Truck Operations
Dearborn, MI 48121

ABSTRACT

A fast, semi-automated method for visualizing the time-varying effects of radiative heat transfer, including obscuration and multiple reflections, is presented. Starting with a finite element surface description, an analyst assigns 'groups' to a model by indicating which elements have the same material and surface properties. The elements within each group are combined into isothermal nodes. View factors are then calculated using a variant of the hemi-cube method. Transient nodal temperatures are calculated and displayed using an implicit solution to the finite difference equations derived from the thermal properties of each node and the radiation exchange between nodes.

INTRODUCTION

This paper presents a method by which the effects of radiative heat transfer may be quickly and easily visualized for models that have been previously prepared for finite element analysis. Based on existing computer modeling techniques, the only user interaction required by the software, known as RadTherm, is the assignment of 'groups' to the model that indicate which elements have the same material and surface properties.

The elements within each group are automatically combined into isothermal nodes. For smaller models, each element may be assigned a thermal node, and therefore have a unique temperature. For larger or more finely meshed models, an average temperature will be calculated for a number of adjacent elements. Combining large numbers of elements into a smaller number of thermal nodes improves run-time and drastically reduces the storage required for multi-bounce radiation exchange factors. It is also an appropriate tradeoff of accuracy for speed since temperatures are predicted based only on radiation exchange.

View factors are calculated using a single plane variant of the hemi-cube method. This algorithm is implemented on an SGI workstation to make use of the high speed graphics capability which includes z-buffering that correctly accounts for obscuration. Multi-bounce radiation exchange factors are calculated by following radiation as it leaves one surface and is reflected to all other surfaces using the view factor information and surface emissivities.

Finite difference equations are derived from the properties of each thermal node and the radiation exchange between nodes. Heat transfer by conduction and convection is not included since the intent is to allow an analyst to visualize the effects of radiation exchange with a minimum amount of input. An implicit solution is used to solve for the transient nodal temperatures which are displayed as they are calculated.

THERMAL NODE ASSIGNMENT

GROUP EDITOR - RadTherm imports PATRAN neutral files containing shell elements. To keep user interaction to an absolute minimum, the analyst only needs to 1) indicate the gross regions of the geometry that have common material and thermal properties, and, 2) give values to the properties. The properties which must be the same for each 'group' of elements are:

- 1) Whether temperature is to be computed or assigned.
- 2) Material ID (not needed if temperature is assigned).
- 3) Material thickness (not needed if temperature is assigned).
- 4) Surface condition (emissivity).
- 5) Assigned constant temperature or temperature vs. time curve (needed only for groups with assigned temperatures).
- 6) Imposed constant heat rate or heat rate vs. time curve (defaults to zero, not needed for groups where temperature is assigned).

The elements are grouped using a 3D graphics editor. The analyst selects individual elements or entire regions of elements by clicking or dragging the mouse. Properties are selected from menus wherever practical. (Figure 1).

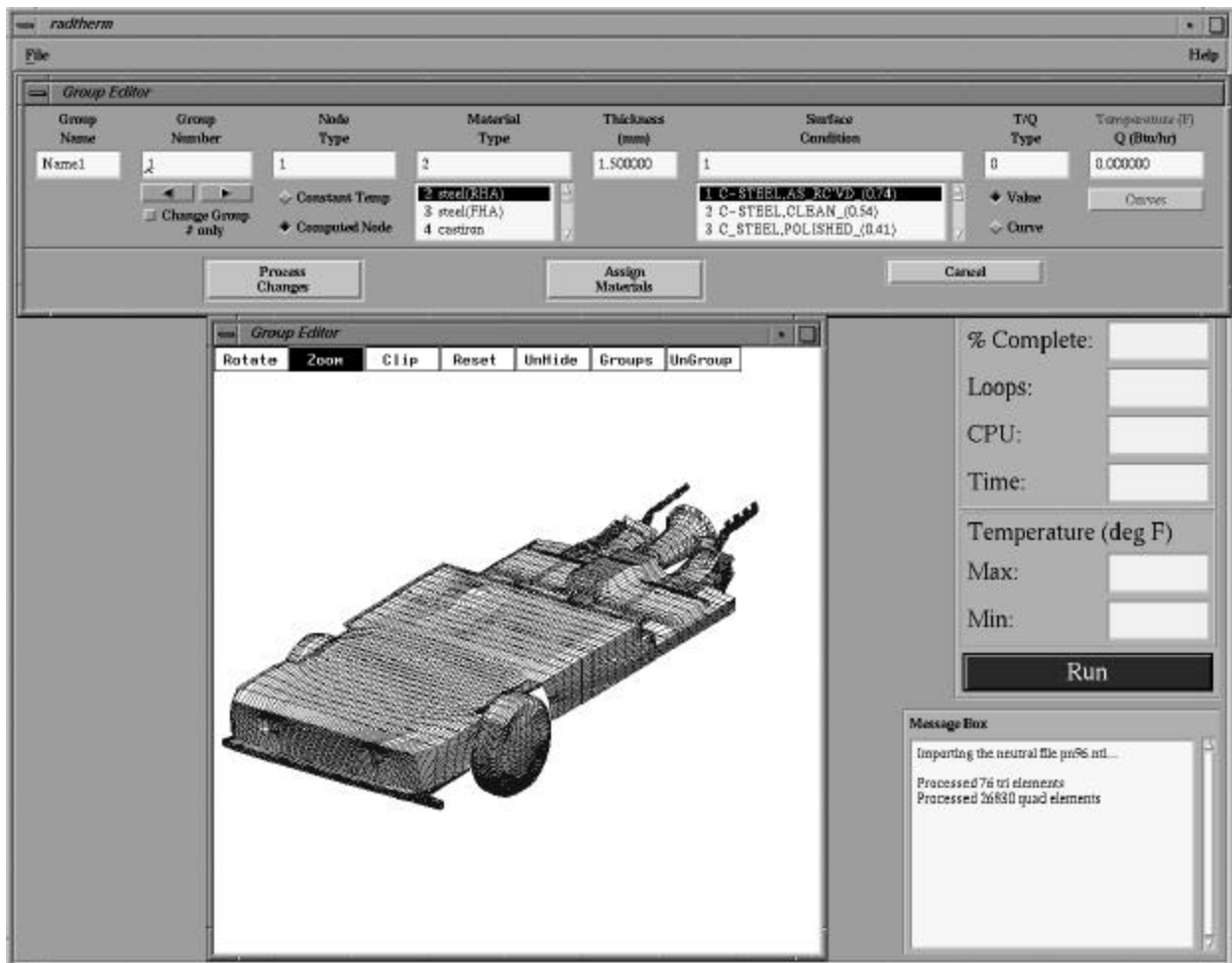


Figure 1: RadTherm's Group Editor is used to assign thermal and material properties to (large) groups of elements.

DISCRETIZER - After the analyst has assigned thermal and material properties to each element via the group editor, RadTherm 'discretizes' or breaks each group into thermal nodes. The discretization is accomplished by combining adjacent elements within each group subject to three (adjustable) constraints:

- 1) The surface normal of the elements must be within 30 degrees of each other.
- 2) The surface area of the combined elements must not exceed a calculated maximum.
- 3) The aspect ratio of the combined elements must not exceed four to one.

A typical model will have 10,000 or more elements, will be assigned a dozen or so groups by the analyst, and will be discretized to have 500 to 1000 thermal nodes.

RADIATION EXCHANGE

The geometry of the object being modeled is surrounded on all sides by a bounding box that is used to set bounding temperatures. The object and its bounding box are considered to be a closed

system of uniformly-irradiated, diffuse-grey surfaces. The net radiation exchange between the surface associated with any two thermal nodes i and j is given by:

$$Q_{ij} = B_{ij} A_i \epsilon_i \sigma (T_i^4 - T_j^4) \quad (1)$$

where B_{ij} is the fraction of energy that is emitted from surface i and absorbed at surface j , both directly and by reflection, A is area, ϵ is emissivity, σ is the Stefan-Boltzmann constant, and T is temperature.

The multi-bounce view factors, B_{ij} , are found by first calculating view factors and then by following reflected radiation until it is eventually absorbed.

VIEW FACTORS - A view factor is the fraction of energy that leaves one surface and is absorbed at a second surface, assuming both bodies are black. The algorithm used by RadTherm to compute view factors is a single plane variant of the hemi-cube method [1]. This method computes view factors by repeatedly rendering the object geometry as seen from points on the surface associated with each thermal node. Each pixel in the rendered image is assigned a weight such that summing up the weights for all the pixels corresponding to the same thermal node yields the view factor from the thermal node at the viewing point to the thermal node scanned in the image. RadTherm makes use of the fact that the geometry has been previously meshed for finite element use by choosing the centroid of each element as the points at which view factors are computed. For thermal nodes which contain more than one element, an area weighted average is taken of the view factors from each element. Pseudo code for the view factor algorithm is given in Figure 2.

```

Set perspective transformation with fov near 180E
    for each thermal node i
        Initialize node-to-node view factor array
        for each element belonging to this thermal node
            Initialize element-to-node view factor array
            Set viewpoint at centroid of element looking out
                along surface normal
            Draw all elements using thermal node number
                as color map index
            for each pixel in bitmap
                Scanned thermal node j = color map index
                Add pixel weighting to view factor array
                    element j
            Multiply element-to-node view factor array by
                ratio of element area to node area
            Sum element-to-node view factor array into node-
                to-node view factor array
            Write node-to-node view factor array for thermal
                node i to view factor file

```

Figure 2: Pseudo code for calculating view factors on a workstation with hardware graphics. For information on the pixel weightings, see [1].

RADIATION EXCHANGE FACTORS - The theoretical development below is based on [2]. For grey bodies, the view factor, F_{ij} , represents the fraction of energy that is emitted from surface i and arrives at all other surfaces, j . The first estimate of the multi-bounce view factor, ${}^0B_{ij}$, is the fraction of energy emitted from i which is absorbed directly at j :

$${}^0B_{ij} = F_{ij} \mathbf{e}_j \quad (2)$$

The leading superscript indicates the number of times that energy from i has been reflected. In the case of direct absorption, the number of reflections is zero. The fraction of energy emitted from

surface i and reflected by all other surfaces, k , is given by:

$${}^1R_{ik} = F_{ik}(I - \mathbf{e}_k) \quad (3)$$

The fraction of energy emitted from surface i , reflected by all other surfaces k , and absorbed at surfaces j , is ${}^1R_{ik} F_{kj} \mathbf{e}_j$. The second estimate of the multi-bounce view factor, which is the sum of the fraction of energy absorbed directly at surface j and after one reflection, is given by:

$${}^1B_{ij} = {}^0B_{ij} + {}^1R_{ik} F_{kj} \mathbf{e}_j \quad (4)$$

The fraction of energy emitted from surface i , reflected by all other surfaces j , and reflected at surfaces k , is given by:

$${}^2R_{ik} = {}^1R_{ij} F_{jk}(I - \mathbf{e}_k) \quad (5)$$

Generalizing the equations above, the estimate of the multi-bounce view factor after N reflections is:

$${}^NB_{ij} = {}^{N-1}B_{ij} + {}^NR_{ik} F_{kj} \mathbf{e}_j \quad (6)$$

The simple algorithm defined by equations 2 through 6 is used by RadTherm to convert view factors to multi-bounce view factors. The estimation process is terminated when ${}^NB_{ij}$ is arbitrarily close to ${}^{N-1}B_{ij}$ or when a maximum number of reflections has been exceeded.

CALCULATING TEMPERATURES

The thermal model in RadTherm is derived from PRISM [3], the Infrared Signature Model developed for the U.S. Army Tank-Automotive Command. Although RadTherm uses an implicit technique to solve for transient nodal temperatures, the more intuitive explicit solution will be described first.

EXPLICIT SOLUTION - The explicit solution to the energy balance for a thermal node can be stated in words as the net energy rate to the thermal node is equal to the heat stored, or mathematically as:

$$\Sigma Q = M \cdot C_p \left(\frac{dT}{dt} \right) \quad (7)$$

where Q is heat rate, M is mass, C_p is specific heat, T is temperature, and t is time. The heat rate terms include radiation and any imposed heat sources added by the analyst. Since this equation is solved with an explicit numerical solution, all of the temperatures used to calculate radiation portion of Q are from the previous time step and the new temperature T_i is calculated directly from the algebraic solution:

$$T_i' = T_i + \frac{\Delta t}{M_i \bullet C_p} \Sigma Q_i \quad (8)$$

Although simple to implement, the explicit solution is inherently unstable. To illustrate this, the radiation heat rate term is reformulated as an effective conductance times the temperature difference between thermal node i and j as follows:

$$C_{ij} = \mathbf{S} SF_{ij} A_i (T_i^2 + T_j^2) (T_i + T_j) \quad (9)$$

where $SF_{ij} = B_{ij} \mathbf{e}_j$ is the radiation exchange factor, and T_i and T_j are temperatures from the previous time step. Note that the pseudo-linearized radiation conductance of equation 9 when multiplied by $(T_j - T_i)$ yields the more familiar $Q = \mathbf{S} SF_{ij} A_i (T_j^d - T_i^d)$.

In addition to defining the radiation conductance term, two other values are useful which will impart additional meaning and criteria to the temperature solution. A capacitance term provides the magnitude of thermal inertia or storage capacity for a given node and is defined as:

$$CAP_i = M_i \bullet C_p \quad (10)$$

A stability constant, $STAB_i$, is defined for node i :

$$STAB_i = \frac{\Delta t}{CAP_i} \sum_{j \neq i}^N C_{ij} \quad (11)$$

Using the definitions in equations 12 and 10, equation 8 can be reformulated as:

$$T_i' = T_i (1 - STAB_i) + \frac{\Delta t}{CAP_i} \left(Q_i + \sum_{j \neq i}^N C_{ij} T_j \right) \quad (13)$$

where Q_i now contains only the imposed heat source terms. Since the new temperature, T_i' , must be based on the (positive) old temperature, T_i , the stability constant must be less than one. From equation 10 it can be seen that $STAB_i < 1.0$ sets an absolute upper limit to the time step: with this Δt based on the smallest $CAP_i / \mathbf{S} C_{ij}$ in the system.

$$\Delta t < \frac{CAP_i}{\sum_{j \neq i}^N C_{ij}} \quad (14)$$

IMPLICIT SOLUTION - The implicit solution is unconditionally stable since it utilizes the new temperatures T_i' in its current time step temperature computation. It requires a matrix or iterative solution at each new time step to solve the NxN matrix for temperature. In spite of the increased complexity in problem formulation and additional internal computations, the implicit formulation will often result in a faster solution with higher accuracy than the explicit solution. The implicit

difference expression is:

$$\frac{CAP_i}{\Delta t} (T_i' - T_i) = \sum_{j \neq i}^N \left[C_{ij} \left(\frac{T_j' + T_j}{2} - \frac{T_i' + T_i}{2} \right) \right] + Q_i \quad (15)$$

and solving for T_i' results in:

$$T_i' = \frac{\left[\sum_{j \neq i}^N C_{ij} T_j + \sum_{j \neq i}^N C_{ij} T_j' + 2Q_i - T_i \sum_{j \neq i}^N C_{ij} \left(1 - \frac{2}{STAB_i} \right) \right]}{\sum_{j \neq i}^N C_{ij} \left(1 + \frac{2}{STAB_i} \right)} \quad (16)$$

Since the unknown temperature T_i' is dependent on the unknown temperatures T_j' , either a matrix or iterative solution must be used. The relatively large number of thermal nodes and the nonlinear radiation conductances, C_{ij} , indicate the use of an iterative method. A variation of the *Gauss-Seidel* procedure is used where new values are used in the equations as soon as they become available. Convergence is achieved when the greatest nodal temperature difference between the present iteration and the previous iteration is less than an arbitrary amount, typically 0.01 to 0.001 °C.

A simple technique is available to increase convergence. The method of over-relaxation extrapolates two successive iterations to reduce the number of required iterations. By defining T_i^{k+1} as the present iterated temperature and T_i^k as the previous iterated temperature, the new over-relaxed temperature is

$$T_i = T_i^k + \beta (T_i^{k+1} - T_i^k) \quad (17)$$

with $1.0 < \beta < 2.0$ as the over-relaxation factor. A divergent solution will occur for $\beta \geq 2.0$. Note that when $\beta = 1.0$, the unrelaxed temperature is produced. This method of solution is called the implicit *Crank-Nicholson* method. The optimum β is based on each particular problem and is typically influenced by nonlinear terms such as radiation. Since RadTherm models have many nonlinear conductances, iterative solutions are avoided and faster convergent solutions are achieved with an under-relaxed factor of $0.5 \leq \beta < 1.0$.

SAMPLE RESULTS

The geometry shown in Figure 3 will be used to illustrate the steps and time required to prepare a RadTherm model. With 26906 shell elements, this is the largest model run to date. All times below are elapsed or user time and are for an SGI Indigo 2 Extreme with an R4400 processor and 64 Mbytes of memory.

- 1) MODEL PREPARATION - It took a thermal analyst approximately two hours to prepare the model for RadTherm. This included assigning 18 groups to the geometry

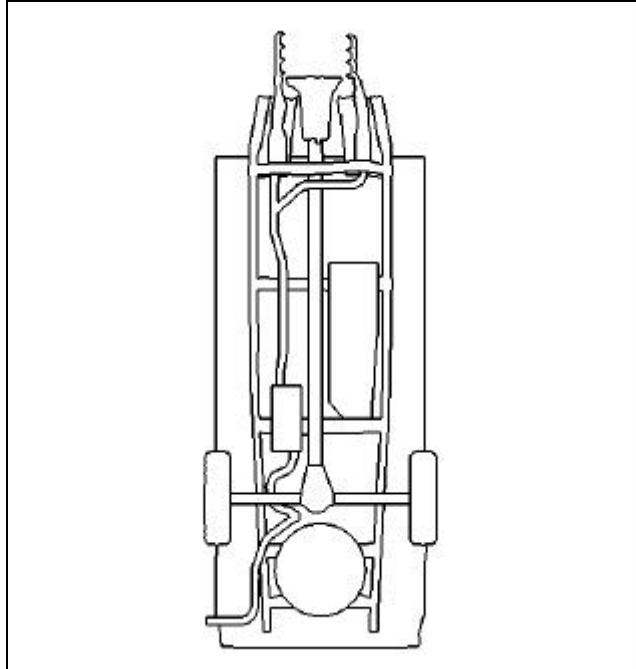


Figure 3: The outline of the 18 element groups as seen from underneath the vehicle.

- 2) **DISCRETIZATION** - Discretization resulted in 2408 thermal nodes (Figure 4). This step took 4 minutes.

- 3) **RADIATION EXCHANGE FACTORS** - View factor calculation took 123 minutes. View factors are not recalculated unless the analyst changes the grouping of the elements and/or discretizes to get a different number of thermal nodes. Calculating multi-bounce radiation exchange factors took an additional 27 minutes. Radiation exchange factors are not recomputed unless the view factors change and/or the analyst changes the surface condition (emissivity) of one or more of the groups.

- 4) **TEMPERATURE CALCULATION** - An hour of simulation took approximately 8 minutes. The results of the visualization are shown Figure 5. Steady-state results can be obtained almost instantaneously by setting the simulation end time equal to the start time.

(Figure 3), giving properties to the groups, and preparing temperature vs. time curves for the constant temperature groups. The rest of the steps are done by the software with only minimal user interaction.

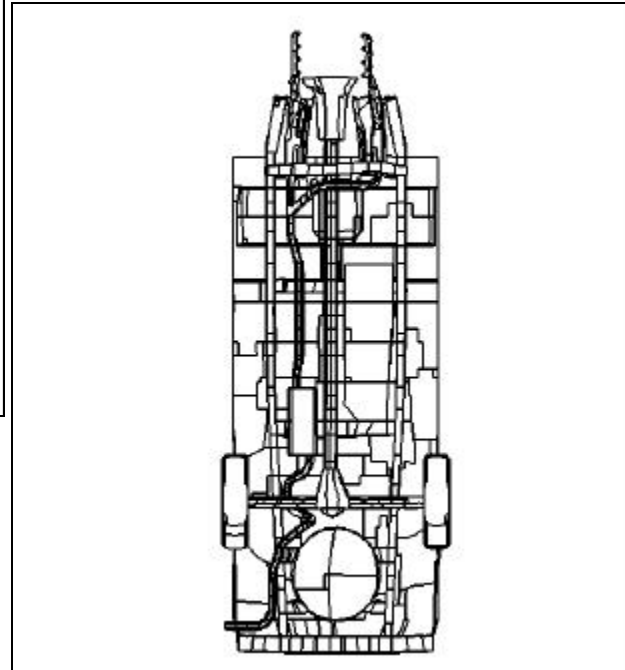


Figure 4: The outline of some of the 2408 thermal nodes as seen from underneath the vehicle.

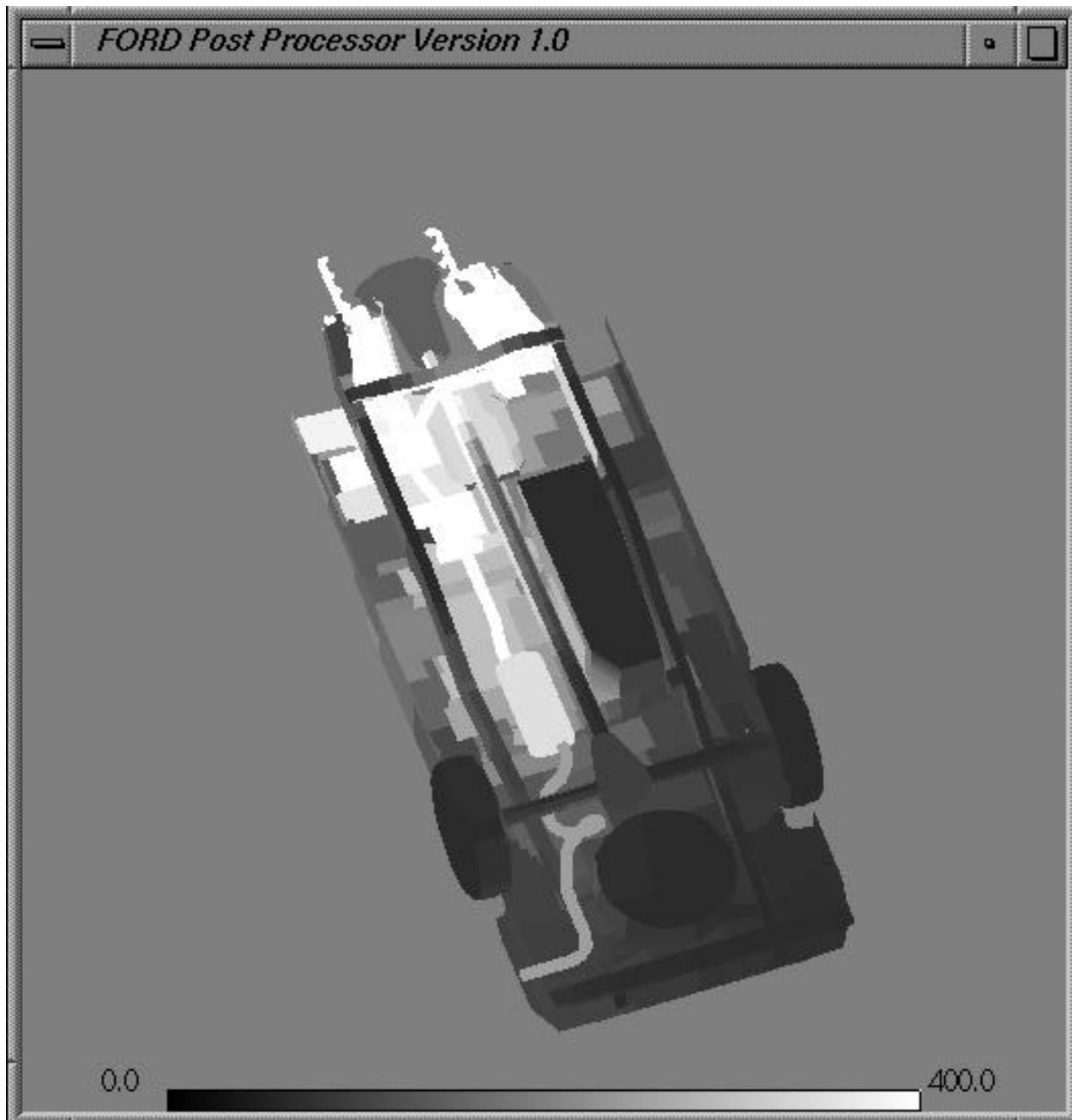


Figure 5: A visualization of the effects of radiation exchange.

REFERENCES

- [1] Cohen, Michael F., John R. Wallace, Radiosity and Realistic Image Synthesis, Academic Press Professional, ISBN 0-12-178270-0, Cambridge, MA, 1993.
- [2] O'Neill, R.F., J.L. Zich, Computer Program Script-F II, McDonnell Douglas Technologies Inc., San Diego, CA, 1989.
- [3] PRISM 3.1 User's Manual, Keweenaw Research Center, Michigan Technological University, Houghton, MI, 1993.