

ON Semiconductor

Is Now

onsemi™

To learn more about onsemi™, please visit our website at
www.onsemi.com

onsemi and **onsemi** and other names, marks, and brands are registered and/or common law trademarks of Semiconductor Components Industries, LLC dba "**onsemi**" or its affiliates and/or subsidiaries in the United States and/or other countries. **onsemi** owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of **onsemi** product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent-Marking.pdf. **onsemi** reserves the right to make changes at any time to any products or information herein, without notice. The information herein is provided "as-is" and **onsemi** makes no warranty, representation or guarantee regarding the accuracy of the information, product features, availability, functionality, or suitability of its products for any particular purpose, nor does **onsemi** assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Buyer is responsible for its products and applications using **onsemi** products, including compliance with all laws, regulations and safety requirements or standards, regardless of any support or applications information provided by **onsemi**. "Typical" parameters which may be provided in **onsemi** data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. **onsemi** does not convey any license under any of its intellectual property rights nor the rights of others. **onsemi** products are not designed, intended, or authorized for use as a critical component in life support systems or any FDA Class 3 medical devices or medical devices with a same or similar classification in a foreign jurisdiction or any devices intended for implantation in the human body. Should Buyer purchase or use **onsemi** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **onsemi** and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that **onsemi** was negligent regarding the design or manufacture of the part. **onsemi** is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner. Other names and brands may be claimed as the property of others.

General Thermal Transient RC Networks

Packaging Technology Development

INTRODUCTION

This monograph develops a generalized thermal-RC-network methodology suitable for generating transient response numerical simulations of thermal systems. Specific classes of boundary conditions are considered, namely, steps and ramps of applied power, and steps and ramps in temperature degree-of-freedom constraints. It also discusses the implications of the form of the general solutions to such networks with respect to the existence of mathematically equivalent “non-grounded-capacitor” thermal RC-network models.

Glossary of Symbols

A_{ij}	Cofactor or minor a determinant
c	Constant of integration
C, C_i	Thermal capacitance, thermal capacitance of a variable temperature node
i	$i^{th}, i = j, j, j^{th}, k, l$, node index
k_j	Constant representing generalized factor in a Laplace transform
k_{xy}	Coefficient of power or temperature in a linear model
g_{ij}	Elements of (steady state) \mathcal{G}
\mathcal{G}	The thermal conductance matrix, and its determinant $\det \mathcal{G}$
\mathcal{G}^i	Cramerized thermal conductance matrix, and its determinant $\det \mathcal{G}^i$
\mathcal{G}^{-1}	Thermal impedance matrix, inverse of the thermal conductance matrix \mathcal{G}
M	A subset of variable temperature nodes to be treated as if fixed
m	“order” of power vector
n	$n + 1 (2n + 3)$ expressions relating to number of variable nodes
$\overline{Q}(s)$	Power vector incorporating only initial temperatures, steps and ramps
q_i	Power input at a node
\overline{Q}	Laplace transform of general power vector
R, R_{ij}	Thermal resistance, thermal resistance between two nodes (assumed bidirectional)
R 's	Thermal resistances in general
s	Laplace transform complex variable, and its square s^2
T_i, T_j	Nodal temperatures
\overline{T}	Laplace transform of temperature vector
\overline{T}_i	Laplace transform of one nodal temperature
$\Delta \overline{T}_i$	Laplace transform of one nodal temperature rise due to step heating
$\tilde{\Gamma}$	Submatrix of \mathcal{G} consisting of coefficients of nodes in subset M
$\tilde{\Psi}$	Same as \mathcal{G}^{-1} , the inverse of the thermal conductance matrix \mathcal{G}
ψ_{ij}	Elements of $\tilde{\Psi}$



ON Semiconductor®

www.onsemi.com

APPLICATION NOTE

BASIC NODE MODEL

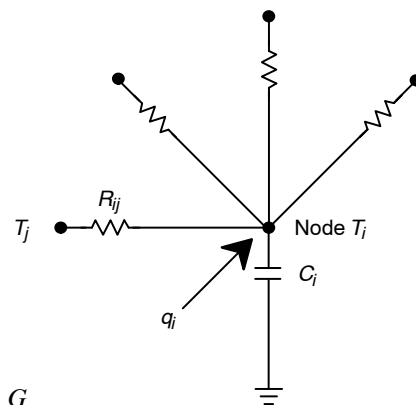


Figure 1. Generalized Thermal Node

Let a typical variable temperature node T_i in a thermal network be represented as shown in the diagram. It consists of a thermal capacitance C_i (tied to thermal “ground”), any number of resistance links to other nodes R_{ij} (whether those nodes be at fixed temperatures or variable in their own right), and a power input q_i at the node. For illustration we have simply labeled one of the other nodes as T_j , but the idea is that when we use a summation over resistance links, the index j will denote each of the other nodes connected to the node of interest (and not the node of interest itself). Then the equation which describes the energy balance at node T_i is:

$$q_i + \sum \frac{T_j - T_i}{R_{ij}} = C_i \frac{dT_i}{dt} \quad (\text{eq. 1})$$

Transformation to S-plane

This differential equation can be converted to an algebraic equation through the use of the Laplace transform. Bolded symbols will represent the **Laplace** transformed quantities corresponding to the original variables. Note also that one aspect of the Laplace transform is that the initial value of the temperature appears explicitly in the transformed equation.

$$\mathbf{q}_i + \sum \frac{\mathbf{T}_j - \mathbf{T}_i}{R_{ij}} = sC_i\mathbf{T}_i - C_iT_{i0} \quad (\text{eq. 2})$$

This can now be rearranged as follows:

$$\left(C_{is} + \sum_{\text{all}} \frac{1}{R_{ij}} \right) \mathbf{T}_i - \sum_{\text{var}} \frac{\mathbf{T}_j}{R_{ij}} = \mathbf{q}_i + \sum_{\text{fixed}} \frac{\mathbf{T}_j}{R_{ij}} + C_iT_{i0} \quad (\text{eq. 3})$$

where we have made a further distinction between “variable” temperature nodes and “fixed” temperature nodes.

$$\begin{pmatrix} C_{1s} + \sum_{\text{all}} \frac{1}{R_{1j}} & \dots & -\frac{1}{R_{1i}} & \dots \\ \vdots & \ddots & \vdots & \vdots \\ -\frac{1}{R_{1i}} & \dots & C_{is} + \sum_{\text{all}} \frac{1}{R_{ij}} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{Bmatrix} \mathbf{T}_1 \\ \vdots \\ \mathbf{T}_i \\ \vdots \end{Bmatrix} = \begin{Bmatrix} \mathbf{q}_1 + \sum_{\text{fixed}} \frac{\mathbf{T}_j}{R_{1j}} + C_1T_{10} \\ \vdots \\ \mathbf{q}_i + \sum_{\text{fixed}} \frac{\mathbf{T}_j}{R_{ij}} + C_iT_{i0} \\ \vdots \end{Bmatrix} \quad (\text{eq. 4})$$

If we define the following quantities:

$$\mathbf{G} = \begin{pmatrix} C_{1s} + \sum_{\text{all}} \frac{1}{R_{1j}} & \dots & -\frac{1}{R_{1i}} & \dots \\ \vdots & \ddots & \vdots & \vdots \\ -\frac{1}{R_{1i}} & \dots & C_{is} + \sum_{\text{all}} \frac{1}{R_{ij}} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} g_{11} & \dots & g_{1i} & \dots \\ \vdots & \ddots & \vdots & \vdots \\ g_{1i} & \dots & g_{ij} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad (\text{eq. 5})$$

(the \mathbf{G} matrix we note, in passing, is symmetric)

In this case, by “fixed” we do not mean so much that the temperature is necessarily constant, but that it is specified (a constraint or boundary condition), as opposed to being an unknown variable in the model. Each temperature node in a particular model will thus appear on either the left or right of the equation – one or the other, depending on its function in the particular model. For instance, in a thermal coldplate model which recognizes heat loss to the environment as well as into the coldplate itself, we will need both “chuck” and “ambient” fixed-temperature nodes. Their temperatures will be specified during the analysis, conceivably even changing over time. It is again important to recognize that the summations are always local to each variable temperature node of the network, and represent whatever set of links connect to that node. The “var” summations on the left include only the links between the node of interest and its adjacent “variable” nodes; the “fixed” summation on the right includes only the links between the (variable) node of interest and its adjacent “fixed” temperature nodes (if any). Note that no summation ever includes an $i = j$ term (i.e., there is no physical sense in having a resistor short itself out at a node!). In addition, we assume that links are completely bidirectional, so there is no distinction between R_{ij} and R_{ji} .

Matrix Form

We can express the entire thermal network in one matrix equation, where each row and column of the matrix (and each element of the vectors) represent the quantities associated with a different variable temperature node.

$$\bar{\mathbf{T}} = \left\{ \begin{array}{c} T_1 \\ \vdots \\ T_i \\ \vdots \end{array} \right\} \quad (\text{eq. 6})$$

$$\bar{\mathbf{Q}} = \left\{ \begin{array}{c} q_1 + \sum_{\text{fixed}} \frac{T_j}{R_{1j}} + C_1 T_{10} \\ \vdots \\ q_i + \sum_{\text{fixed}} \frac{T_j}{R_{ij}} + C_i T_{io} \\ \vdots \end{array} \right\} \quad (\text{eq. 7})$$

Then the entire network can be represented simply as:

$$\underline{\mathcal{G}} \cdot \bar{\mathbf{T}} = \bar{\mathbf{Q}} \quad (\text{eq. 8})$$

In ordinary application of this network model to either experimental data fitting, or otherwise in prediction of thermal performance of a system, we will generally specify all the quantities of $\underline{\mathcal{G}}$ and $\bar{\mathbf{Q}}$, and will seek solutions for $\bar{\mathbf{T}}$. Thus

$$\bar{\mathbf{T}} = \underline{\mathcal{G}}^{-1} \cdot \bar{\mathbf{Q}} \quad (\text{eq. 9})$$

which is easy enough to write symbolically, but provides certain challenges in practice.

Boundary Conditions

Before we go any further into specific applications, however, let us consider what sort of values are of interest for $\bar{\mathbf{Q}}$. Obviously, constant power inputs and truly fixed “reference” temperature nodes will be useful. If we limited ourselves to these boundary conditions, we could solve problems of any desired complexity in a piecewise fashion, simulating time-varying power and constraint conditions with square-edged constant waveforms (i.e. constant “steps”) approximating the shapes of the actual conditions. In so doing, we see that even though we might begin our first step with all nodes at some common temperature (thus momentarily eliminating the initial temperature constant terms), once we had started the solution, the nodes would move to different temperatures. The ending temperatures from each completed step would then become the initial

temperature terms of the following step – so in general we must retain these initial temperature terms, and in general they will all be different from each other. We may also find that square-edged power and temperature constraints are too unrealistic for some situations, so let us permit them to include a “ramp” characteristic (i.e., constant non-zero slopes as needed) from the outset, and see if the extra manipulations become too burdensome. Utilizing the Laplace transforms of steps and ramps, we can write the i^{th} element of $\bar{\mathbf{Q}}$ as follows:

$$Q_i = \frac{q_i}{s} + \frac{\dot{q}_i}{s^2} + \frac{\sum \frac{T_j}{R_{ij}}}{s} + \frac{\sum \frac{\dot{T}_j}{R_{ij}}}{s^2} + C_i T_{io} \quad (\text{eq. 10})$$

Where all quantities (except s) are now true constants, and the “dotted” values are the slopes of the associated quantities. Finally, collecting powers of s yields:

$$Q_i = \left(\dot{q}_i + \sum \frac{\dot{T}_j}{R_{ij}} \right) \frac{1}{s^2} + \left(q_i + \sum \frac{T_j}{R_{ij}} \right) \frac{1}{s} + C_i T_{io} \quad (\text{eq. 11})$$

It should now be clear that if we need to handle power “ramps,” it costs us nothing in complexity to throw in constrained-temperature node “ramps” as well. So with the step and ramp restriction, our system can be expressed entirely as polynomials in s , like this (where all the new subscripted quantities are simply constants)

$$\left(\begin{array}{cccc} C_{1s} + g_{11} & \dots & g_{1i} & \dots \\ \vdots & \ddots & \vdots & \vdots \\ g_{1i} & & C_{is} + g_{ii} & \\ \vdots & & \vdots & \ddots \end{array} \right) \left\{ \begin{array}{c} T_1 \\ \vdots \\ T_i \\ \vdots \end{array} \right\} = \frac{1}{s^2} \left\{ \begin{array}{c} C_1 T_{10} s^2 + a_1 s + b_1 \\ \vdots \\ C_i T_{io} s^2 + a_i s + b_i \\ \vdots \end{array} \right\} \quad (\text{eq. 12})$$

Or, with an obvious definition of $\overline{Q(s)}$

$$\underline{\mathcal{G}} \cdot \bar{\mathbf{T}} = \frac{1}{s^2} \overline{Q(s)} \quad (\text{eq. 13})$$

Solution Method

A brute-force solution of this system (using Cramer’s Rule) now can be carried out to the point that Laplace transforms of the nodal temperatures are expressed as:

$$T_i = \frac{1}{s^2} \frac{\det \underline{G}^i}{\det \underline{G}} \quad (\text{eq. 14})$$

Where \underline{G}^i is the matrix obtained by substituting $\overline{Q}(s)$ for the i^{th} column of \underline{G} . It should be evident that $(\det \underline{G})$ ends up being a polynomial in s of order n , where n is the number of variable nodes in the network, and that $(\det \underline{G}^i)$ ends up being a polynomial of order $n + 1$. With the additional s^2 term in the denominator, standard techniques will yield a fairly straightforward inverse transform of T_i into the time domain, resulting in a solution T_i for each node consisting of possibly a constant plus a linear term in time, and then mainly exponentials (in negative powers of time, with time constants derived from the roots of $\det \underline{G}$). Should there be repeated roots, the result will get somewhat more complicated, nevertheless, the methodology is standard and direct.

We can code this solution method into a computer program, and delay conversion from symbolic to numerical analysis until the last possible moment. A tool such as Theorist® or Mathematica® can be used to generate the determinants in full symbolic form, though if there are very many nodes in the model, there may be dubious value in so doing (especially since the equations will be unique for each network topology analyzed). Other tools, such as LabVIEW®, can be used to compute the determinants in a quasi-symbolic form (i.e. identifying which elements of the various matrices and vectors must be multiplied together and added with cofactor multiplicands), and resorting to numerical computations only whenever new polynomial

coefficients are needed. It is not clear which approach, if either, has an advantage, since once the model consists of more than five nodes, there will never be a general, closed-form solution for the roots of the denominator. In either case, the determinants must be recomputed (symbolically or quasi-symbolically) whenever the topology changes; and in neither case do determinant structures have to be recomputed due to changes in specific network parameters. The question revolves more around the format of what gets saved in between parameter changes (and the associated costs in time and memory). On the other hand, there will be symbolic expressions for the total coefficients of the final numerator and denominator polynomials (each of which is a complex combination of the entire set of R ’s and C ’s, all of which appear multiple times in various permutations), or quasi-symbolic representations of the determinants — with potentially hundreds of individual equations in terms of row-element references. The former likely trades off a much lengthier derivation time of the symbolic coefficients (for a possibly somewhat faster recomputation of their numerical values), whereas the latter may be faster in generating the quasi-symbolic determinants, at the expense of having more numerical substitutions to be made into the actual computation of the polynomial coefficients.

There is at least one special case of further interest here, however. That is the “step heating” problem, where the entire network is at a uniform-temperature thermal equilibrium, and one or possibly more nodes are then powered up, each with a constant power (unpowered nodes being considered as constant zero power). We can refer all temperatures to that uniform initial temperature, eliminate the resulting zero terms (including the power and temperature “ramp” values), and obtain:

$$\begin{pmatrix} C_1s + g_{11} & \dots & g_{1i} & \dots \\ \vdots & & \ddots & \\ g_{i1} & & C_1s + g_{ii} & \\ \vdots & & & \ddots \end{pmatrix} \begin{Bmatrix} \Delta T_1 \\ \vdots \\ \Delta T_i \\ \vdots \end{Bmatrix} = \frac{1}{s} \begin{Bmatrix} q_1 \\ \vdots \\ q_i \\ \vdots \end{Bmatrix} \quad (\text{eq. 15})$$

Considering but one of the nodal solutions, it may now be seen that the result for ΔT_i will be of the form:

$$\Delta T_i = \frac{1}{s} \frac{\sum (q_j \cdot A_{ij})}{\det \underline{G}} \quad (\text{eq. 16})$$

Rearranging slightly, we can also write this as:

$$\Delta T_i = \sum q_j \cdot \frac{1}{s} \frac{A_{ij}}{\det \underline{G}} \quad (\text{eq. 17})$$

Thus the Laplace transform of the solution for node i will be a series of n polynomial fractions, one for each node in the system. Each term has the same denominator, but each has

a different numerator according to which node it represents, and each is multiplied by the power dissipation at that node. One term will be present for each powered node, and the solution for every node in the system (whether itself powered or not) will have the same number of independent terms. Powered nodes thus will have a so-called “self heating” term, plus “interaction heating” terms for each of the other powered nodes.

Implications for “Non-Grounded-Capacitor” Models

The main point in this derivation has been to show that an arbitrary thermal RC-network model can have any of its

individual node's solutions expressed as a combination of terms individually proportional to the power inputs of each associated heated node. Obviously, a computer program can be written to generate these solutions systematically. There is, however, an unanticipated secondary conclusion to be drawn from this work. We continually refer (for instance, see AND8215/D and AND8222/D) to grounded-capacitor models as having physical significance, but to non-grounded models as providing nothing more than mathematical convenience. The derivation (of eq. 17) demonstrates that there is a mathematically convenient equivalent non-grounded ladder for each node in a grounded-capacitor network. It is not clear how one should

combine multiple non-grounded networks together to simultaneously provide equivalent behavior for multiple nodes of the general grounded-capacitor network. Because the A_{ij} are different for each node, each node in the "non-grounded" model must be represented by a different combination network. Further, it is not at all obvious that such a combined network would respond correctly to other combinations of inputs (especially from non-uniform starting temperatures, and certainly not to different combinations of power and temperature ramps). This difficulty may be seen a little more clearly if we return to equation 12 and dissect the right-hand side:

$$\mathcal{G} \cdot \mathbf{T} = \begin{Bmatrix} C_1 T_{10} \\ \vdots \\ C_i T_{i0} \\ \vdots \end{Bmatrix} + \frac{1}{s} \begin{Bmatrix} q_1 + \sum \frac{T_j}{R_{1j}} \\ \vdots \\ q_i + \sum \frac{T_j}{R_{ij}} \\ \vdots \end{Bmatrix} + \frac{1}{s^2} \begin{Bmatrix} \dot{q}_1 + \sum \frac{\dot{T}_j}{R_{1j}} \\ \vdots \\ \dot{q}_i + \sum \frac{\dot{T}_j}{R_{ij}} \\ \vdots \end{Bmatrix} \quad (\text{eq. 18})$$

If we choose to separate the temperature into three obvious components:

$$\mathbf{T} = \mathbf{T}_0 + \mathbf{T}_{step} + \mathbf{T}_{ramp} \quad (\text{eq. 19})$$

Where

$$\mathbf{T}_0 = \mathcal{G}^{-1} \cdot \begin{Bmatrix} C_1 T_{10} \\ \vdots \\ C_i T_{i0} \\ \vdots \end{Bmatrix} \quad (\text{eq. 20})$$

$$\mathbf{T}_{step} = \frac{1}{s} \mathcal{G}^{-1} \cdot \begin{Bmatrix} q_1 + \sum \frac{T_j}{R_{1j}} \\ \vdots \\ q_i + \sum \frac{T_j}{R_{ij}} \\ \vdots \end{Bmatrix} \quad (\text{eq. 21})$$

And

$$\mathbf{T}_{ramp} = \frac{1}{s^2} \mathcal{G}^{-1} \cdot \begin{Bmatrix} \dot{q}_1 + \sum \frac{\dot{T}_j}{R_{1j}} \\ \vdots \\ \dot{q}_i + \sum \frac{\dot{T}_j}{R_{ij}} \\ \vdots \end{Bmatrix} \quad (\text{eq. 22})$$

We see that the influences of initial conditions, constant power (and constant temperature constraints), and ramps of power (with ramps of constraint temperature) each add their own independent contributions to the overall response of the

network. It can now be seen precisely how the different contributions relate to each other – and it is not in simple "proportion." It is true (and perhaps a valuable insight), that fixed-temperature nodes act just as if they represented fixed power inputs. Likewise, power ramps and temperature constraint ramps are completely equivalent in effect. However, going back into the time domain, the relationship between initial conditions and constant power would be that the "power" contribution is related to the integral of the "initial condition" contribution.

To see this, first consider a straightforward modification of Eq. 17, which now incorporates the fixed-temperature contribution to the solution for any node i :

$$\Delta \mathbf{T}_{i-step} = \sum \left[\left(q_j + \sum \frac{T_k}{R_{jk}} \right) \cdot \frac{1}{s} \frac{A_{ij}}{\det \mathcal{G}} \right] \quad (\text{eq. 23})$$

And by similarity, we can simply write the other two components of \mathbf{T}_i :

$$\Delta \mathbf{T}_{i-ramp} = \sum \left[\left(\dot{q}_j + \sum \frac{\dot{T}_k}{R_{jk}} \right) \cdot \frac{1}{s^2} \frac{A_{ij}}{\det \mathcal{G}} \right] \quad (\text{eq. 24})$$

$$\Delta \mathbf{T}_{i0} = \sum \left(C_j T_{j0} \cdot \frac{A_{ij}}{\det \mathcal{G}} \right) \quad (\text{eq. 25})$$

Note that the cofactors A_{ij} depend only on \mathcal{G} (and upon the node for which we're solving). In each case, we have the inner product of a constant vector (specific to ramp, step, or initial conditions), and a vector that depends only on \mathcal{G} (and

is therefore fixed for whichever node is of interest). To put it another way, all three contributions to the overall solution can be written in the form:

$$\Delta T_{i,m} = \frac{1}{s^m} \sum \left(k_j \cdot \frac{A_{ij}}{\det \underline{G}} \right) \quad (\text{eq. 26})$$

For any specific node i , therefore, the time-domain solution will be of the form:

$$\Delta T_{i(m)} = \sum \left(k_{j(m)} e^{-\frac{t}{\tau_j}} \right) \quad (\text{eq. 27})$$

As we increase m , the solution will change like this:

$$\Delta T_{i(m+1)} = c - \sum \left(\tau_j k_{j(m)} e^{-\frac{t}{\tau_j}} \right) \quad (\text{eq. 28})$$

$$\begin{pmatrix} \sum_{\text{all}} \frac{1}{R_{1j}} & \dots & -\frac{1}{R_{1i}} & \dots \\ \vdots & \ddots & \vdots & \ddots \\ -\frac{1}{R_{1i}} & \dots & \sum_{\text{all}} \frac{1}{R_{ij}} & \dots \\ \vdots & \ddots & \vdots & \ddots \end{pmatrix} \begin{Bmatrix} T_1 \\ \vdots \\ T_i \\ \vdots \end{Bmatrix} = \begin{Bmatrix} q_1 + \sum_{\text{fixed}} \frac{T_j}{R_{1j}} \\ \vdots \\ q_i + \sum_{\text{fixed}} \frac{T_j}{R_{ij}} \\ \vdots \end{Bmatrix} \quad (\text{eq. 29})$$

As previously noted, temperature-constrained nodes play an equivalent role to heat inputs, thus demonstrating that in a linear system, temperatures at every variable-temperature node will be merely linear combinations of all heat inputs and fixed-temperature boundary nodes.

If we further define

$$\Psi = \underline{G}^{-1} \quad (\text{eq. 30})$$

then

$$\bar{T} = \Psi \cdot \bar{q} \quad (\text{eq. 31})$$

Recalling our earlier notation, the elements of \underline{G} are g_{ij}

$$\underline{G} = \begin{pmatrix} \sum_{\text{all}} \frac{1}{R_{1j}} & \dots & -\frac{1}{R_{1i}} & \dots \\ \vdots & \ddots & \vdots & \ddots \\ -\frac{1}{R_{1i}} & \dots & \sum_{\text{all}} \frac{1}{R_{ij}} & \dots \\ \vdots & \ddots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} g_{11} & \dots & g_{1i} & \dots \\ \vdots & \ddots & \vdots & \ddots \\ g_{1i} & \dots & g_{ii} & \dots \\ \vdots & \ddots & \vdots & \ddots \end{pmatrix} \quad (\text{eq. 32})$$

Similarly, $\underline{\Psi}$, the inverse of \underline{G} , has elements ψ_{ij}

$$\underline{\Psi} = \begin{pmatrix} g_{11} & \dots & g_{1i} & \dots \\ \vdots & \ddots & \vdots & \ddots \\ g_{1i} & \dots & g_{ii} & \dots \\ \vdots & \ddots & \vdots & \ddots \end{pmatrix}^{-1} = \begin{pmatrix} \psi_{11} & \dots & \psi_{1i} & \dots \\ \vdots & \ddots & \vdots & \ddots \\ \psi_{1i} & \dots & \psi_{ii} & \dots \\ \vdots & \ddots & \vdots & \ddots \end{pmatrix} \quad (\text{eq. 33})$$

Note that the order (size) of these matrices corresponds to the number of independent heat sources, not the total number of nodes in the system. (The boundary nodes increase the number of linearly independent terms contained within the power vector.) Separating the “true” heat sources from the temperature boundary nodes in the power vector, we then can express the temperature solution at a variable temperature node as follows:

$$T_i = \sum_{\text{var}} \psi_{ij} \cdot q_j + \sum_{\text{var}} \psi_{ij} \cdot \sum_{\text{fixed}} \frac{T_j}{R_{ij}} \quad (\text{eq. 34})$$

where c is a constant of integration. However, because the k_j are different for the different “orders” of the solution contributions, one cannot actually derive the values of the constants for one order, from a solution to another order. The most important point here is that there is not a simple relationship between the various contributions to the overall solution, and the non-grounded networks which describe one portion of the solution will therefore not work for the others.

Steady-State Resistor Networks

If transient solutions are not desired, a matrix equation describing the system can be created by returning to the time domain and eliminating time-dependent terms (refer to earlier Eq. 4):

So, if one were to derive from experiment (or more likely, a simulation) all the coefficients of a linear combination of heat sources and boundary nodes characterizing a thermal system, those coefficients could be reinterpreted into node-to-node resistances.

Three-heat-source example

For example, in a 3-heat source system (nodes 1, 2, 3) with two boundary nodes (nodes 4, 5), we might determine the following relationship describing the internal nodes' temperatures as a functions of the independent variables:

$$T_1 = k_{11} q_1 + k_{12} q_2 + k_{13} q_3 + k_{14} T_4 + k_{15} T_5 \tag{eq. 35a}$$

$$T_2 = k_{21} q_1 + k_{22} q_2 + k_{23} q_3 + k_{24} T_4 + k_{25} T_5 \tag{eq. 35b}$$

$$T_3 = k_{31} q_1 + k_{32} q_2 + k_{33} q_3 + k_{34} T_4 + k_{35} T_5 \tag{eq. 35c}$$

If we would prefer to create a pure “resistor model” of the system from these linear combinations, compare Eqs.35 with Eqs. 36 (derived for this specific example from Eq.34) and equate coefficients:

$$T_1 = \psi_{11}q_1 + \psi_{12}q_2 + \psi_{13}q_3 + \left(\frac{\psi_{11}}{R_{14}} + \frac{\psi_{12}}{R_{24}} + \frac{\psi_{13}}{R_{34}}\right)T_4 + \left(\frac{\psi_{11}}{R_{15}} + \frac{\psi_{12}}{R_{25}} + \frac{\psi_{13}}{R_{35}}\right)T_5 \tag{eq. 36a}$$

$$T_2 = \psi_{21}q_1 + \psi_{22}q_2 + \psi_{23}q_3 + \left(\frac{\psi_{21}}{R_{14}} + \frac{\psi_{22}}{R_{24}} + \frac{\psi_{23}}{R_{34}}\right)T_4 + \left(\frac{\psi_{21}}{R_{15}} + \frac{\psi_{22}}{R_{25}} + \frac{\psi_{23}}{R_{35}}\right)T_5 \tag{eq. 36b}$$

$$T_3 = \psi_{31}q_1 + \psi_{32}q_2 + \psi_{33}q_3 + \left(\frac{\psi_{31}}{R_{14}} + \frac{\psi_{32}}{R_{24}} + \frac{\psi_{33}}{R_{34}}\right)T_4 + \left(\frac{\psi_{31}}{R_{15}} + \frac{\psi_{32}}{R_{25}} + \frac{\psi_{33}}{R_{35}}\right)T_5 \tag{eq. 36c}$$

We can infer the following

$$\begin{array}{lll} \psi_{11} = k_{11} & \psi_{21} = k_{21} & \psi_{31} = k_{31} \\ \psi_{12} = k_{12} & \psi_{22} = k_{22} & \psi_{32} = k_{32} \\ \psi_{13} = k_{13} & \psi_{23} = k_{23} & \psi_{33} = k_{33} \end{array}$$

That is, the square 3x3 k_{ij} matrix is, in fact, the Ψ matrix

$$\tilde{\Psi} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{22} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \tag{eq. 37}$$

Further, inspection of the T_4 and T_5 coefficients shows that the following relationships must be satisfied:

$$\overline{k_4} = \tilde{\Psi} \cdot \left[\frac{1}{R_4} \right] \qquad \overline{k_5} = \tilde{\Psi} \cdot \left[\frac{1}{R_5} \right] \tag{eq. 38}$$

$$\overline{R_4} = \frac{1}{\tilde{\Psi}^{-1} \cdot \overline{k_4}} \qquad \overline{R_5} = -\frac{1}{\tilde{\Psi}^{-1} \cdot \overline{k_5}} \tag{eq. 39}$$

The notation may be obscure, but the meaning in each of the above equations is that a vector on the left hand side is found through a multiplication operation between a matrix and a vector on the right hand side. Thus the resistors between the interior (variable) nodes and the boundary (fixed) nodes are readily obtainable.

Recalling that the Ψ matrix is the inverse of the G matrix (Eq. 30)

$$\tilde{G} = \tilde{\Psi}^{-1} = \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{22} \\ k_{31} & k_{32} & k_{33} \end{pmatrix}^{-1} = \begin{pmatrix} \sum_{\text{all } j \neq i} \frac{1}{R_{1j}} & -\frac{1}{R_{12}} & -\frac{1}{R_{13}} \\ -\frac{1}{R_{21}} & \sum_{\text{all } j \neq i} \frac{1}{R_{2j}} & -\frac{1}{R_{23}} \\ -\frac{1}{R_{31}} & -\frac{1}{R_{32}} & \sum_{\text{all } j \neq i} \frac{1}{R_{3j}} \end{pmatrix} \tag{eq. 40}$$

Thus, finally, the inter-heat source resistances are the negative reciprocals of the off-diagonal elements of the inverted Ψ matrix.

$$g_{ij} = -\frac{1}{R_{ij}} (i \neq j) \quad (\text{eq. 41})$$

The main diagonal terms offer no new information, although they do provide a consistency check for the model, as they represent the parallel resistance sum of all the resistors connected to each heat source (including the boundary nodes):

In a truly linear system, the math should work out. In a nonlinear system, or one where heat sources (and temperature constraints) may comprise areas rather than points, and temperatures are measured at isolated points (or averages over areas), neither of the theoretically symmetric matrices will be exactly symmetric. Likewise, the consistency check between individual sums of parallel resistors (as deduced from the coefficients of descriptive linear combinations, as in Eq. 35), and the main diagonal terms of the conductance matrix, will not hold precisely.

Corollary 1: For any physically realizable linear thermal system, all coefficients of heat sources and boundary temperatures will be positive (i.e. for directly deduced equations such as Eq. 35 – whether through simulation or experiment). As we have just seen, the heat-source coefficients represent the Ψ matrix of the system. On the other hand, by definition, whereas the main diagonal elements of the \mathcal{G} matrix are positive, the off diagonal elements are negative. Such a relationship does not hold for a random (even symmetric) positive definite matrix and its inverse. It would seem this must have to do with the physical constraint that the total conductance away from a node (through all paths in parallel) cannot be less than the conductance of any individual path. In other words, the magnitude of each main diagonal element of \mathcal{G} , will always equal or exceed the magnitude of the sum of all other elements in its row or column (all negative).

$$g_{ii} \geq -\sum_{j \neq i} g_{ij} \quad (\text{eq. 42})$$

$$g_{ii} \geq -\sum_{j \neq i} g_{ji}$$

Corollary 2: The sum of all the independent temperature coefficients (that is, the effective power contributions from boundary temperature nodes, as opposed to the true power contributions) for a given temperature calculation is always unity. In the example of Eq. 35 above, this implies

$$k_{14} + k_{15} = 1 \quad (\text{eq. 43})$$

That is to say

$$\frac{\psi_{11}}{R_{14}} + \frac{\psi_{11}}{R_{15}} = \psi_{11} \left(\frac{1}{R_{14}} + \frac{1}{R_{15}} \right) \quad (\text{eq. 44})$$

One can see that this must be true for a linear thermal system (such as we have described it herein), because if there was no internal heat dissipation (all $q_i = 0$), and all temperature boundaries were equal (all $T_j = T_b$), all interior

temperatures must equilibrate to that uniform boundary temperature. If there are m heat sources and n boundary temperatures, then (again following Eq. 35) we would write:

$$T_i = \sum_{j=1}^m k_{ij} \cdot q_j + \sum_{j=m+1}^{m+n} k_{ij} \cdot T_j$$

$$T_i = \sum_{j=1}^m k_{ij} \cdot 0 + T_b \sum_{j=m+1}^{m+n} k_{ij} \quad (\text{eq. 45})$$

$$T_i = 0 + T_b \sum_{j=m+1}^{m+n} k_{ij}$$

$$1 = \sum_{j=m+1}^{m+n} k_{ij}$$

In other words, the only way it can be true that all interior temperatures equilibrate to an arbitrary uniform boundary temperature is if the sum of the coefficients of the boundary temperatures is unity.

Corollary 3: In many situations, one primary purpose for building a thermal model is to permit it to be explored over a variety of boundary conditions. With this goal in mind, if the number of potential boundary temperature constraints is small compared to the total number of variable degrees of freedom in a linear thermal system, it may be much more computationally efficient to treat some (or all) of the boundary constraints as variables.

That is, if you start with the \mathcal{G} matrix derived directly from a nodal model (Eq. 32), even if all thermal properties are known (meaning, the links between all nodes in a model are determined), a complete matrix inversion from \mathcal{G} to Ψ is required any time a node changes from variable to fixed-temperature – because the size of the problem changes (i.e. the number of rows and columns in \mathcal{G} and Ψ). For example, changing a two nodes from variable to fixed, in a 1000 degree of freedom model, requires performing a new 998-order matrix inversion.

The alternative is to realize that any variable temperature node can be quickly forced to a desired temperature (thus mimicking a fixed temperature constraint). Suppose M is a small subset of all the variable temperature nodes in a model. First, extract a submatrix \mathcal{L} from \mathcal{G} of all the coefficients pertaining to only those nodes of interest:

$$\mathcal{L} = [g_{kl}]_{\substack{k \text{ in } M \\ l \text{ in } M}} \quad (\text{eq. 46})$$

This submatrix gives the sensitivity of the temperatures of those nodes (and only those nodes) to perturbations of power at those nodes. Therefore, its inverse tells you the how much heat needs to be input at those nodes in order to change their temperatures by an arbitrary amount, to wit

$$\overline{\Delta q}_{in M} = \underline{\Gamma}^{-1} \cdot \overline{\Delta T}_{in M} \quad (\text{eq. 47})$$

For those specific quasi-fixed nodes, we can view the difference between their previously computed temperatures and their *intended* temperatures as an “error” to be corrected by providing additional heat (or removing heat) at those nodes. That is to say

$$\overline{\Delta T}_{err} = \overline{T}_{quasi\ fixed} - [T_k]_{k\ in\ M} \quad (\text{eq. 48})$$

Since changing the heat input at *any* (variable) node within a model will necessarily change the temperature at *every* variable temperature node, one can then recompute the entire (modified) temperature solution in one of two ways:


Option 1: Augment the original power vector by the amounts computed in Eq. 47 (original power vector plus power perturbation). Then recompute the entire temperature solution at once:

$$\overline{T}_{final} = \Psi \cdot (\overline{q} + \overline{\Delta q}) \quad (\text{eq. 49})$$

Option 2: Compute an overall temperature *perturbation* based on the power perturbation alone, and superimpose that perturbation onto the original solution.

$$\overline{T}_{final} = \overline{T}_{original} + \Psi \cdot \overline{\Delta q} \quad (\text{eq. 50})$$

LabVIEW is a trademark of National Instruments Corporation.
 Mathematica is a registered trademark of Mathematica Policy Research, Inc.
 Theorist is a registered trademark of Theorist Interactive, LLC.

ON Semiconductor and  are trademarks of Semiconductor Components Industries, LLC dba ON Semiconductor or its subsidiaries in the United States and/or other countries. ON Semiconductor owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of ON Semiconductor's product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent-Marking.pdf. ON Semiconductor reserves the right to make changes without further notice to any products herein. ON Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does ON Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Buyer is responsible for its products and applications using ON Semiconductor products, including compliance with all laws, regulations and safety requirements or standards, regardless of any support or applications information provided by ON Semiconductor. "Typical" parameters which may be provided in ON Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. ON Semiconductor does not convey any license under its patent rights nor the rights of others. ON Semiconductor products are not designed, intended, or authorized for use as a critical component in life support systems or any FDA Class 3 medical devices or medical devices with a same or similar classification in a foreign jurisdiction or any devices intended for implantation in the human body. Should Buyer purchase or use ON Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold ON Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that ON Semiconductor was negligent regarding the design or manufacture of the part. ON Semiconductor is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
 Literature Distribution Center for ON Semiconductor
 19521 E. 32nd Pkwy, Aurora, Colorado 80011 USA
Phone: 303-675-2175 or 800-344-3860 Toll Free USA/Canada
Fax: 303-675-2176 or 800-344-3867 Toll Free USA/Canada
Email: orderlit@onsemi.com

N. American Technical Support: 800-282-9855 Toll Free
 USA/Canada
Europe, Middle East and Africa Technical Support:
 Phone: 421 33 790 2910

ON Semiconductor Website: www.onsemi.com
Order Literature: <http://www.onsemi.com/orderlit>

For additional information, please contact your local Sales Representative