

Using GNU Scientific Library for Temperature Computation in VLSI Systems

Sławomir Mikula, Andrzej Kos
AGH University of Science and Technology, Cracow
mikula@agh.edu.pl, kos@agh.edu.pl

Abstract

The paper presents a methods of using Green's Functions as a useful tool for partial differential equations solving. A short history and field of use the Green's function is presented. The mathematical foundation of solving the heat equation using the Green's function is presented in detail. Implementation with use of the GNU Scientific Library with detailed algorithm and computation approximations is presented. Execution time is compared with the Matlab software.

1. Introduction

The Green's functions, named after the English mathematician and physicist, George Green (1773-1841), are very powerful tools for obtaining solutions of transient and steady-state linear heat conduction problems. They can also be applied for the solution of some convection problems and to many other phenomena which are described by the same type of equations. These problems usually involve solution of diffusion-type partial differential equations. A Green's function (G_F) is a basic solution of a specific differential equation with homogeneous boundary conditions. It is a building block from which many useful solutions may be constructed. For transient heat conduction, a G_F describes the temperature distribution caused by an instantaneous, local energy pulse.

The G_F method is related to other methods for solving heat conduction problems. The classic methods in heat conduction, including the method of separation of variables and the Laplace transform, are used to derive the G_F . Approximate methods of finding G_F developed by Haji-Sheikh [7] extends usability of the Green's Function on a numerical side.

Green functions are also a useful tool in condensed matter theory, where they allow the resolution of the diffusion equation and in quantum mechanics, where the Green function of the Hamiltonian is a key concept, with important links to the concept of density of states. The primary use of Green's functions in mathematics is to solve inhomogeneous boundary condition problems. In physics, Green's functions are also usually used as propagators in Feynman diagrams (and the phrase "Green's function" is often used for any correlation function). In addition to solution procedures, the GF method also provides greater understanding of the nature of diffusion processes, including heat conduction flow in porous media.

2. Thermal computation with the Green's function

A Green's function is an integral kernel that can be used to solve an inhomogeneous differential equation with boundary conditions. It serves roughly an analogous role in partial differential equations as does Fourier analysis in the solution of ordinary differential equations.

The heat conduction equation, which presents the temperature in the specific point of the chip is described as [1]:

$$k \nabla^2 T = c_v \frac{\partial T}{\partial t} \quad (2.1)$$

Thermal conductivity is defined by $k = \rho c_p \kappa$, where c_p is the heat capacity, and κ is the thermal diffusivity. The main goal of our computations is to compute the temperature in the specific point, in the three dimensional space (x, y, z) , in the defined time moment t :

$$T = T(x, y, z, t) \quad (2.2)$$

After transformations, the equation, which is used in this work is given as expression:

$$\frac{\partial^2 T}{\partial x_1^2} + \frac{\partial^2 T}{\partial y_1^2} + \frac{\partial^2 T}{\partial z_1^2} = \frac{\partial T}{\partial t_1} \quad (2.3)$$

$$\text{where } x_1 = \frac{x}{c} \quad y_1 = \frac{y}{c} \quad z_1 = \frac{z}{c}, \text{ and } t_1 = \frac{t}{\frac{c_v c^2}{k}}$$

The dimension values are normalized by the chip thickness. The time value is normalized by the expression $\frac{c_v c^2}{k}$. The temperature distribution in an infinite one-dimensional, constant-property body [7] which has an initial temperature distribution $F(x)$, and volumetric energy generation $g(x, t)$ (with units of W/m^3), is described by:

$$\frac{\partial^2 T}{\partial x^2} + \frac{1}{k} g(x, t) = \frac{1}{\alpha} \frac{\partial T}{\partial t}, \quad -\infty < x < \infty, \quad t > 0 \quad (2.4)$$

$$T(x, 0) = F(x) \quad (2.5)$$

The temperature $T(x, t)$, the solution of (2.4) and (2.5), is called a Green's function solution equation.

$$T(x, t) = \int_{t'=-\infty}^{\cdot} G(x, t | x', 0) F(x') dx' + \frac{\alpha}{k} \int_{\tau=0}^{\cdot} \int_{t'=-\infty}^{\cdot} G(x, t | x', \tau) g(x', \tau) dx' d\tau \quad (2.6)$$

The particular form of $G_F(\cdot)$ for an infinite one-dimensional body is a fundamental heat conduction solution (Cannon [6]).

$$G(x, t | x', \tau) = K(x - x', t - \tau) \begin{cases} [4\pi\alpha(t - \tau)]^{-\frac{1}{2}} \exp\left[-\frac{(x - x')^2}{4\alpha(t - \tau)}\right], & t - \tau \geq 0 \\ 0, & t - \tau < 0 \end{cases} \quad (2.7)$$

Based on the Green function described for infinite one dimensional solution and using principles described above, we could easily derive result for the multi-dimensional case. Let's assume existence of an area without boundary conditions. A Silicon is placed everywhere in the three dimensional space (x,y,z) and a rectangle heat source with sizes specified by coordinates (x₁, x₂), (y₁, y₂), (z₁, z₂). Let the arbitrary point X be placed in the heat source area described by coordinates (x', y', z').

The temperature in an arbitrary point in the space is specified by a formula:

$$T(x, y, z, t) = \int_{t_1}^{t_2} \int_{x_1}^{x_2} \int_{y_1}^{y_2} \int_{z_1}^{z_2} G_F(x-x', y-y', z-z', t-t') dz' dy' dx' dt' \quad (2.8)$$

The Z direction of the real heat source is normally greater than 0. In this work we assume, that the thickness of the heat source is much smaller than a thickness of the whole chip ($z' \ll c$). This assumption leads to omitting integral of the z' variable in the expression (2.8). Using the equation (2.7) which determines temperature for a single point, we are able to introduce a formula, which describes the temperature in a single point in a space caused by a square heat source:

$$T(x, y, z, t) = \int_{t_1}^{t_2} \left[\frac{1}{8\sqrt{\pi(t-t')}} e^{\frac{-z^2}{4(t-t')}} \int_{x_1}^{x_2} e^{\frac{-(x-x')^2}{4(t-t')}} dx' \int_{y_1}^{y_2} e^{\frac{-(y-y')^2}{4(t-t')}} dy' \right] dt' \quad (2.9)$$

Farther the equation will be applied to numerical calculations.

3. Using GNU Scientific Library for temperature computations

Equation (2.9) can be numerically computed. There is need to perform triple integration for each computing point in the space. Considering computing power or modern informatics systems this is not a big issue. There are although simple methods for equation complexity reduction. Approximation of the definite Riemann integrals of the X and Y dimensions using error function leads to the final solution. Numerical computation is performed using Gauss-Kronrod method with 41-point integration rule. For specific integration ranges (fig. 1) we use two integral computing methods based on Gauss-Kronrod integration. Algorithms are used from GSL toolkit [5].

The first algorithm (QAG) is an adaptive integration procedure. The integration region is divided into subintervals, and on each iteration the subinterval with the largest estimated error is bisected. The second algorithm (QAGS) combines adaptive bisection with the Wynn epsilon-algorithm to speed up the integration of many types of integrable singularities. This function applies the Gauss-Kronrod 21-point integration rule adaptively. The integrals for dimensions X and Y could be replaced by the appropriate error functions [2]. The basic formula for the error function is

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du \quad (3.1)$$

Finally we got the equation, which is going to be implemented numerically:

$$T(x,y,z,t) = \int_{t_1}^{t_2} \frac{1}{8\sqrt{\pi(t-t')}} e^{\frac{-z^2}{4(t-t')}} \cdot \left(\operatorname{erf}\left(\frac{x-x_2}{2\sqrt{t-t'}}\right) - \operatorname{erf}\left(\frac{x-x_1}{2\sqrt{t-t'}}\right) \right) \left(\operatorname{erf}\left(\frac{y-y_2}{2\sqrt{t-t'}}\right) - \operatorname{erf}\left(\frac{y-y_1}{2\sqrt{t-t'}}\right) \right) dt' \quad \text{where } z' = 0 \quad (3.2)$$

In order to compute the equation above and to minimize error introduced by the error function approximation the time integral (from t_1 to t_2) is divided and for each area error function is computed independently.

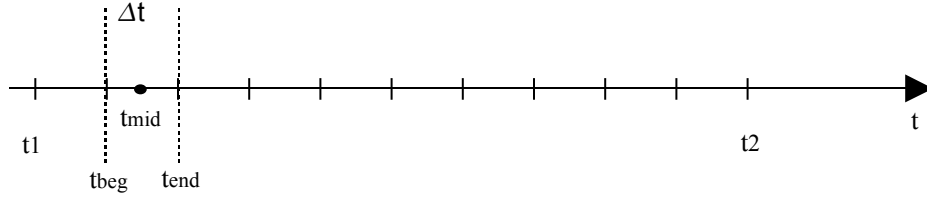


Fig. 1. Computation time division

For each area expression (3.2) is computed. The error function variable t' is always equal to the centre of the currently computed time area Δt :

$$t_{mid} = \frac{t_{end} - t_{beg}}{2} = \frac{\Delta t}{2} + t_{beg} \quad (3.3)$$

The final result is the addition of results computed for all time areas:

$$T = \sum T_k \quad (3.4)$$

It is a very simple computer operation.

4. Detailed computation algorithm

The computation method was based on the equation (3.2). Computation were made using numerical software. Calculations are executed during the heat module activity time (in this example from t_1 to t_2). This period is divided to many sub-periods in order to calculate error functions of the X and Y integrals.

Steps needed by the algorithm with short description of each stage are presented on the figure 2. Computation were made using numerical software. Calculations are executed during the heat module activity time (in this example from t_1 to t_2). This period is divided into many sub-periods in order to calculate error functions of the X and Y integrals.

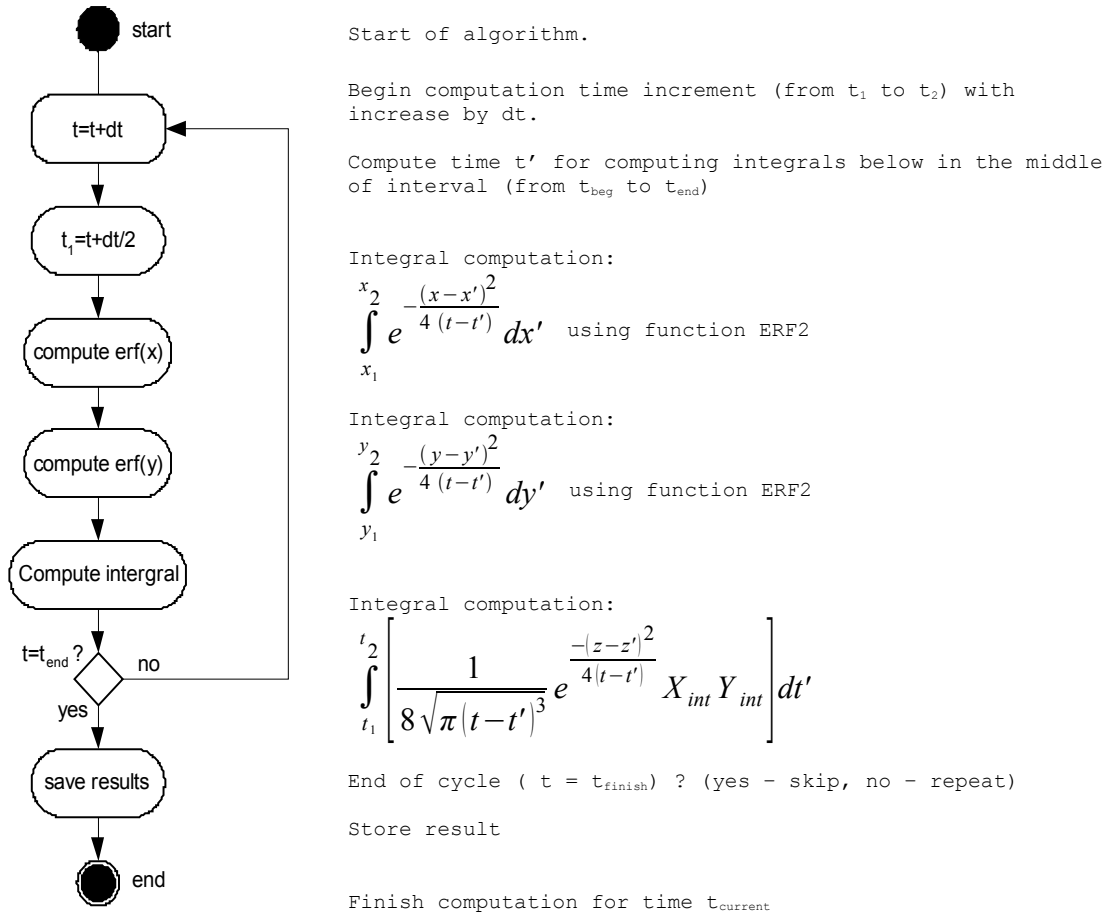


Fig. 2. Algorithm with explanation of each step

5. Algorithm time execution comparison

Presented algorithm was implemented in application used for chip surface temperature computation with optimization of maximal temperature based on scheduling algorithms for functional blocks of researched micro-processor [9]. There is a main focus of this article, to present usefulness of GNU Scientific library and to compare execution time with popular scientific software.

The test element, is a chip of the size of each side equal to 3mm. Chip thickness is default to 0.3 mm. Heat source is located centrally on the chip, with the dimesion 2mm x 2mm. The activity time is set to 2 seconds.

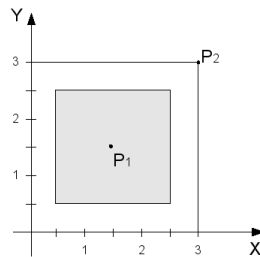


Fig. 3. Test element

Temperature was measured in two points, one placed centrally in the heat center, and second outside the heat source. In the same coordinates relative to the center of the chip. The temperature was measured in 10 steps, with 1 second interval after the chip activity time. Computation time was measured with the 10 us accuracy (in case of application compute time, time was rounded to 10 us). Computation was performed 5 times and the mean values were compared. Results are presented in the table 1.

	<i>Matlab 6.5</i>	<i>GNU Scientific Library</i>
Point P ₁	avg. 0,55 [s]	0,02 [s]
Point P ₂	avg. 0.69 [s]	0.03 [s]

Table.1. Test results

Computer used in computations: Pentium III 1,13 GHz, 384MB RAM. Application used with comparison: Matlab 6.5.0.180913a Release 13 (GNU/Linux Version) and application using Gnu Scientific Library version 1.6 (Ubuntu Linux libgsl-1.6-2). For Matlab, standard triple integral function was used – triplequad – with default option values (tolerance = 1e-6 and default quadrature function (QUAD).

Conclusions

The solution, considering a heat equation as an partial differential equation, was presented based on the Green's function method. Methods of integral approximation with detailed algorithm was developed and implemented on the GNU/Linux platform. The comparison of GNU Scientific Library and the approach of compilation using high level languages with the popular mathematical software, gives good results. Computation time on the same machine was much faster. By usage this library, user is not bound only to one operating system environment or proprietary software solutions.

Bibliography

- [1] Kos, Andrzej, De Mey, Gilbert "Thermal Modelling and optimization of power microcircuits", Electrochemical Publications, Bristol, England, 1997
- [2] Abramowitz, Milton and Stegun, Irene A. (Eds.) "Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables"; 9th printing; New York: Dover, 1972
- [3] Stevens, W. Richard "Advanced Programming in the Unix Environment", Addison Wesley Longman, Reading, MA 1993
- [4] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery "Numerical Recipes in C. The Art of Scientific Computing. Second Edition", Cambridge University Press (1988-1992).
- [5] Galassi, Mark; Davies, Jim; Theiler, James; Gough, Brian; Jungman, Gerard; Booth, Michael; Rossi, Fabrice "GNU Scientific Library Reference Manual Edition 1.5", for GSL Version 1.5; 19 July 2004
- [6] Cannon, John Rozier, "The one-dimensional heat equation", Encyclopedia of Mathematics and its Applications, 23. Addison-Wesley Publishing Company, 1984
- [7] Beck J.V., "Heat Conduction Using Green's Function", (Series in Computational Methods and Physical Processes in Mechanics and Thermal Sciences), Taylor & Francis, 1992