An initial value formulation of the CSDT method of solving partial differential equations

by VENKATESWARARAO VEMURI

Purdue University
Lafayette, Indiana

INTRODUCTION

Numerical methods of solving partial differential equations (PDEs) using analog or hybrid computers fall into three broad categories. Assuming, for concreteness, that one of the independent variables is time and the rest are spatial, the continuous-space and discrete-time (or CSDT) methods envisage to keep the space-like variable continuous and discretize the time-like variable. Similarly, the terms discrete-space and continuous time (DSCT) and discrete-space and discrete-time (DSDT) approximations are self-explanatory. For a one-space dimensional PDE, for instance, both the CSDT and DSCT approximations yield a set of ordinary differential equations while the DSDT approximations lead to a set of algebraic equations. Because of the inherent need to handle a continuous variable, both CSDT and DSCT approximations lend themselves well for computation on analog or hybrid computers. Indeed, several analog and hybrid computer implementations of all these three methods are currently in vogue each method claiming to be superior in some respect to the others. However, it was the CSDT method that showed great promise and produced little results. The purpose of this paper is to present another alternative to this problem.

One of the fundamental advantages of the CSDT method over others is its ability to handle moving boundaries. This can be readily achieved by controlling the analog computer's integration interval since the problem space variable is represented by computer-time. A second advantage is that the analog hardware requirements of the CSDT method are very modest because a relatively small analog circuit is time-shared to solve the entire problem. With the advent of modern high-speed iterative analog and hybrid computers the above promises of the CSDT method appeared to be almost within the reach.[1,2]

In practice, however, considerable difficulties were encountered in obtaining dependable results using the CSDT method.[2,3] The major difficulty is that the CSDT methods are inherently unstable. Methods that were proposed to circumvent this stability problem are either conceptually wrong or impose additional computational burdens making their efficiency debatable. A second difficulty with the CSDT methods is that the basic spatial sweep from boundary to boundary, at each discrete time level, yields a two point boundary value problem (TPBVP) which in turn has to be solved iteratively. It is not clear, at the outset, whether any advantage gained by time-sharing of the analog hardware is really tangible when compared to the price paid in solving a TPBVP. A third disadvantage is that the CSDT method is essentially limited to handle problems in one space dimension only.

This paper suggests a new alternative which still adopts the basic CSDT procedure but results in an initial-value problem. By this formulation the first two difficulties cited in the preceding paragraph are eliminated. This paper still treats a one-space-dimensional problem and no attempt was made here to extend the concept to higher dimensions. However, it is not quite inconceivable to extend this technique to higher dimensions by using this in conjunction with an alternating direction iterative method.

STATEMENT OF THE PROBLEM

Consider the simple diffusion equation

$$\frac{\partial^2 U}{\partial x^2} = \frac{\partial U}{\partial t}\,; \quad U = U(x,t) \tag{1}$$

with the initial conditions

$$U(x,0) = U_0(x) = f(x)\,; \quad 0 \leq x \leq 1 \tag{2}$$

and without loss of generality[5,7] with the homogeneous boundary conditions

$$\left.\begin{aligned} U(0,t) &= 0 \\ U(1,t) &= 0 \end{aligned}\right\} \tag{3}$$

A CSDT approximation to (1), (2) and (3) can be written, as usual, by using a backward difference approximation for the time-derivative. Specifically, at time $t = t_k$, using a simple difference scheme, Eq. (1) can be approximated by

$$LU_k(x) = \frac{1}{(\Delta t)}\left[U_k(x) - U_{k-1}(x)\right]; \quad 0 \leq x \leq 1 \tag{4}$$

where $L \triangleq d^2/dx^2$ is a differential operator and (Δt) is the size of the time step taken. With this approximation, the auxilary conditions (2) and (3) take the form

$$U_0(x) = f(x) \tag{5}$$

$$\left.\begin{aligned} U_k(0) &= 0 \\ U_k(1) &= 0 \end{aligned}\right\} \tag{6}$$

The classical method of implementing the CSDT method is to solve (4) on an analog computer with the initial condition (5) and the boundary conditions (6). However, equations (4), (5) and (6) constitute a TPBVP as such $U_k(x)$ for $0 \leq x \leq 1$ at any time level $t = t_k$ cannot be obtained in a single computer run; an iterative procedure is required to determine $U_k(x)$ at each $t = t_k$. This iterative procedure is often performed using either a trial and error procedure or by using a search technique such as the steepest descent method. Under such circumstances, scaling limitations of analog computers place severe restrictions on the region of search making them unattractive. Coupled with the inherent instability of the analog computer circuit solving (4), this necessity to solve a TPBVP at each time level is therefore the major drawback of the conventional CSDT method.

FORMULATION OF INTEGRAL EQUATION

The initial value formulation starts once again with equations (4), (5) and (6). Instead of solving them as a TPBVP, equations (4) through (6) are first transformed into an equivalent integral equation of the Fredholm type. The first step of this procedure, which can be found in any standard work,[4–7] is to determine a Green's function of the differential operator L in (4) that also satisfies the homogeneous boundary conditions in (6). Specifically, a Green's function for the operator $L = d^2/dx^2$ that satisfies the homogeneous boundary conditions in (6) can be written as

$$K(x,y) = \begin{cases} x(1-\eta); & 0 \leq x \leq \eta \leq 1 \\ \eta(1-x); & 0 \leq \eta \leq x \leq 1 \end{cases} \tag{7}$$

It is important to note that the Green's function has one form for $x < \eta$ and another for $\eta < x$ and that in each semi-interval it has a structure of the product of a function of x alone and a function of η alone. Such a structure is called semi-degenerate, which can greatly simplify the problem. If the Green's function $K(x, \eta)$ obtained is not degenerate or semi-degenerate, it can always be approximated, to any desired degree of accuracy, by a semi-degenerate kernel using standard techniques.[5–7] Therefore, the procedure presented here is not good for any nondegenerate kernel.

Solution of the TPBVP described by (4), (5) and (6) can now be written in terms of the Green's function (7) as

$$U_k(x) = \frac{1}{(\Delta t)}\int_0^1 K(x,\xi)\,U_{k-1}(\xi)\,d\xi - \frac{1}{(\Delta t)}\int_0^1 K(x,\xi)\,U_k(\xi)\,d\xi \tag{8}$$

or

$$U_k(x) = f_{k-1}(x) + \lambda\int_0^c K(x,\xi)\,U_k(\xi)\,d\xi \tag{9}$$

where $f_{k-1}(x)$ is the first term on the right hand side of (8) and can be explicitly evaluated because $U_{k-1}(x)$ represents a solution obtained at the preceding time level $t = t_{k-1}$. The terms λ and c in (9) are defined by

$$\lambda = -\frac{1}{(\Delta t)}; \quad c = 1$$

and are introduced merely for convenience and generality.

Equation (9) is a Fredholm integral equation of the second kind in its most familiar format. In (9), $f_{k-1}(x)$ is called the *free term*, λ the *parameter* and $U(x, \xi)$ is called the *kernel*. Without going into the details of a proof, let it be stated that for a well-posed problem, the solution $U(x, t)$ of the given PDE can be approximated by the sequence of functions $U_k(x)$ which are indeed the solution of the above integral equation.

This procedure of transforming the given PDE into an equivalent Fredholm type integral equation was apparently suggested also by Chan[8] in a recent paper but he adopts an *iterative* procedure to solve the integral equation.

SOLUTION OF THE INTEGRAL EQUATION

The next computational step is to solve the integral equation presented in (9) for $U_k(x)$. Classical methods of solving (9) are essentially iterative in nature[9] and so are not suitable for real-time operation. Furthermore, analog computers are ideally suited for solving problems with prescribed initial conditions. It, therefore, is logical to search for methods of transforming integral equations into sets of ordinary differential equations with prescribed initial conditions. Such a method was recently suggested by Kalaba.[10]

Kalaba's method is essentially one of treating the interval of integration $(0, c)$ as a variable rather than as a constant. By regarding the solution at a fixed point as a function of the interval of integration (now being treated as a variable), a set of ordinary differential equations with a complete set of initial conditions can be obtained. With a knowledge of the solution for one interval length, it is now easy to generate solutions for other interval lengths or for any interval length using this equation as a vehicle. Furthermore, the set of ordinary differential equations with prescribed initial conditions can be solved very easily on an analog computer.

Equation (9) is the starting point for the formulation of the initial-value problem. Treating the interval $(0, c)$ as a variable, (9) can be rewritten as

$$U_k(x, \tau) = f_{k-1}(x) + \int_0^\tau K(x, \xi) U_k(\xi, \tau)\, d\xi; \qquad 0 \leq x \leq \tau \quad (10)$$

It is assumed that (10) has a solution for $\tau \leq c$. For τ sufficiently small and

$$0 \leq x \leq \tau \quad (11)$$

the solution $U_k(x, \tau)$ of (10) can be proved (see appendix) to be identical to the solution of the set of equations *defined* by (12) through (20).

$$G(\tau) \triangleq (1 - \tau) + \tau r(\tau) \quad (12)$$

$$\frac{dr(\tau)}{d\tau} \triangleq [G(\tau)]^2 \quad (13)$$

$$\frac{de(\tau)}{d\tau} \triangleq G(\tau)[f_{k-1}(\tau) + (1 - \tau)e(\tau)]; \qquad \tau > 0 \quad (14)$$

with the initial conditions at $\tau = 0$ given by

$$r(\tau = 0) = r(0) = 0 \quad (15)$$

$$e(\tau = 0) = e(0) = 0 \quad (16)$$

and

$$\frac{dJ(x, \tau)}{d\tau} = G(\tau)[\tau \cdot J(x, \tau)]; \qquad \tau > x \quad (17)$$

$$\frac{dU_k(x, \tau)}{d\tau} = [f_{k-1}(\tau) + \tau e(\tau)]J(x, \tau) \cdot \tau; \qquad \tau > x \quad (18)$$

with the initial condition at $\tau = x$ given by

$$J(x, \tau = x) = (1 - x) + xr(x) \quad (19)$$

and

$$U_k(x, \tau = x) = f_{k-1}(x) = xe(x) \quad (20)$$

COMPUTATIONAL PROCEDURE

Equations (12) through (20) can now be solved using an analog computer or a hybrid computer. The various computational stages are indicated below.

Step 1. Solve (13) and (14) on an analog computer over the interval $0 \leq \tau \leq x$ by treating τ as computer time. Initial conditions for this computer run are given by (15) and (16) respectively.

Step 2. After integrating until time $\tau = x$, the analog computer is placed in HOLD mode and the solutions r and e, at $\tau = x$, obtained in step 1 are used to evaluate the expressions in (19) and (20). These values will be useful as initial conditions while solving (17) and (18) in the next step.

Step 3. At time $\tau = x$, and after (19) and (20) are evaluated equations (17) and (18) are adjoined to the original set (13) and (14) and both sets are integrated over the interval $x \leq \tau \leq c$ by putting the analog computer back in COMPUTE mode. During this phase of integration, the *initial conditions* of the additional set are the values of J and U_k evaluated *not* at $\tau = 0$ but at $\tau = x$. This is precisely the reason and purpose of the computation in step 2.

Step 4. The output of the integrator solving equation (18) in $U_k(x, \tau)$ and this is the solution of (9) at the argument x. This is also the solution of the PDE (1) at a particular time level $t = t_k$.

DISCUSSION

Initial-value problems are conceptually simple, computationally easy to solve and are susceptible for simulation studies. Simulation inherently involves trial and error experimentation in which the validity of a model is verified; sensitivity to environment is explored and variation of performance due to parameter changes evaluated. Such problems come under the classical heading of inverse problems—that is, problems where a system's performance is known from a measured set

of observations and the nature of the system is to be determined. While solving such inverse problems by using such search techniques as gradient methods, it is often necessary to solve not only the dynamic equation of the system, such as (1), but also an additional equation called the derived equation. This is not a mere doubling of computational effort as it appears at first sight. The computational effort required in the evaluation of the gradient increases very fast if the derived equation is an adjoint equation posed as a final value problem. It is precisely in bottleneck situations like this that an initial-value formulation comes in handy.

A second possible application of this method would be in on-line control or identification of distributed parameter systems.

Implementation of this method, particularly when the kernel has no simple structure requires some degree of sophistication in the analog system. If the Green's function (or Kernel) contains, or is approximated by, expressions that are sums of products of a large number of terms then the analog circuit generally contains a large number of multipliers. This may make the scaling a little more difficult. Finally, computation from step 2 to step 3 requires a degree of sophistication in the analog switching system. Many present generation hybrid computer systems can indeed handle most of these requirements.

No attempt was made in this paper to present a procedure that can be applied to any partial differential equation. Similarly no assumptions were made that would restrict the procedure to the simple case presented. In the general case, an easy procedure is required to obtain equations (12) through (20) from (10). Material filling these gaps and results supporting this procedure will be presented in a subsequent paper.

ACKNOWLEDGMENTS

The author wishes to express his thanks to Professor R. Kalaba who first introduced the procedure presented in the appendix, to solve an integral equation. He also wishes to express his gratitude to Professor George A. Bekey of the University of Southern California who provided support during the initial stages of this work. provided support during the initial stages of this work from AFOSR–1018–67C.

REFERENCES

1 H S WITSENHAUSEN
Hybrid solution of initial value problems for partial differential equations
MIT Electronics Systems Lab DSR 9128 Memo No 8 August 1964

2 R VICHNEVETSKY
A new stable computing method for the serial hybrid computer integration of partial differential equations
Proc Spring Joint Computer Conference Vol 33 Pt 1 pp 565–574 1968

3 H H HARA W J KARPLUS
Application of functional optimization techniques for the serial hybrid computer solution of partial differential equations
Proc Fall Joint Computer Conference Vol 33 pt 1 pp 565–574 1968

4 R E BELLMAN
Introduction to the mathematical theory of control processes
Academic Press 1967 Vol 1

5 F B HILDEBRAND
Methods of applied mathematics
Prentice Hall 1952

6 H H KAGIWADA R E KALABA
A practical method for determining Green's function using Hadamard's variational formula
J Optimization Theory Appl Vol 1

7 F G TRICOMI
Integral equations
Interscience 1963

8 S-K CHAN
The serial solution of the diffusion equation using nonstandard hybrid techniques
IEEE Transactions on Computers Vol 1-8 No 9 pp 786–799 September 1969

9 G A BEKEY W J KARPLUS
Hybrid computation
Wiley 1968

10 H H KAGIWADA R E KALABA
An initial value theory for Fredholm integral equations with semi-degenerate kernels
Rand Corporation Memorandum RM-5602-PR April 1968

APPENDIX

Outline of the initial-value formulation

Step 1: The proof starts with a realization that if $\Phi(x, \tau)$ is a solution of the integral equation

$$\Phi(x, \tau) = K(x, \tau) + \int_0^\tau K(x, \xi)\Phi(\xi, \tau)\, d\xi, \qquad 0 \leq x \leq \tau \quad \text{(A1)}$$

then

$$W(x, \tau) \triangleq \Phi(x, \tau)U(\tau, \tau) \quad \text{(A2)}$$

is a solution of the equation defined by

$$W(x, \tau) \triangleq K(x, \tau)u(\tau, \tau) + \int_0^\tau K(\lambda, \xi)W(\xi, \tau)\, d\xi \quad \text{(A3)}$$

A proof of this statement can be found in any standard book on integral equations.[5,7]

Step 2: To prove that the integral equation (10) is equivalent to the set of differential equations (12)

through (20), equation (10) is first differentiated with respect to τ. Denoting derivatives with respect to τ by primes, this differentiation yields

$$U_k'(x,\tau) = K(x,\tau)U_k(\tau,\tau) + \int_0^\tau K(x,\xi)U_k'(\xi,\tau)\,d\xi \tag{A4}$$

If $U_k'(x,\tau)$ is identified with $W(x,\tau)$, equation (A4) is identical to (A3). Therefore, the solution $U_k'(x,\tau)$ of Eq. (A4) can be written as

$$U_k'(x,\tau) = \Phi(x,\tau)U(\tau,\tau);\quad 0 \le x \le \tau \tag{A5}$$

where $\Phi(x,\tau)$ is the solution of (A1).

Step 3: Directing attention once again on (A1) and replacing the kernel $K(x,\tau)$ by its semi-degenerate approximation, namely

$$K(x,\tau) = \begin{cases} x(1-\tau); & 0 \le x \le \tau \\ (1-x)\tau; & 0 \le \tau \le x \end{cases} \tag{A6}$$

equation (A1) can be written as

$$\Phi(x,\tau) = \tau(1-x) + \int_0^\tau K(x,\xi)\Phi(\xi,\tau)\,d\xi;\quad x \le \tau \tag{A7}$$

$$= \tau J(x,\tau); \tag{A8}$$

where $J(x,\tau)$ is defined by the integral equation

$$J(x,\tau) \triangleq (1-x) + \int_0^\tau K(x,\xi)\Phi(\xi,\tau)\,d\xi \tag{A9}$$

Step 4: If $J(x,\tau)$ can be determined, then using (A8) the function $\Phi(x,\tau)$ can be obtained which in turn will aid in getting $U_k(x,\tau)$ from (A10). The procedure to get $J(x,\tau)$ is very similar to the one used to get (A5).

Differentiating (A9) with respect to τ and using the same principle indicated in Step 1, one gets

$$J'(x,\tau) = \Phi(x,\tau)J(\tau,\tau) \tag{A10}$$

Step 5: In order to get $J(\tau,\tau)$, one goes back to (A9), from which

$$J(\tau,\tau) = (1-\tau) + \int_0^\tau K(x,\tau)J(\xi,\tau)\,d\xi; \tag{A11}$$

$$= (1-\tau) + \int_0^\tau x(1-\xi)J(\xi,\tau)\,d\xi \tag{A12}$$

$$= (1-\tau) + x \cdot r(\tau) \tag{A13}$$

where $r(\tau)$ is defined by

$$r(\tau) \triangleq \int_0^\tau (1-\xi)J(\xi,\tau)\,d\xi;\quad 0 \le \tau \tag{A14}$$

Step 6: The value of $r(\tau)$ can be determined by using, once again, a procedure similar to that used in Step 2. Differentiating (A14) with respect to τ

$$r'(\tau) = (1-\tau)J(\tau,\tau) + \int_0^\tau (1-\xi)J'(\xi,\tau)\,d\xi, \tag{A15}$$

Substituting (A10) in (A15)

$$r'(\tau) = (1-\tau)J(\tau,\tau) + \int_0^\tau (1-\xi)\Phi(\xi,\tau)J(\tau,\tau)\,d\xi \tag{A16}$$

The value of $\Phi(\xi,\tau)$ from (A8) can now be substituted in (A16).

$$r'(\tau) = (1-\tau)J(\tau,\tau) + \tau\int_0^\tau (1-\xi)J(\xi,\tau)J(\tau,\tau)\,d\xi \tag{A17}$$

using the definition of $r(\tau)$ from (A14)

$$r'(\tau) = (1-\tau)J(\tau,\tau) + \tau J(\tau,\tau)r(\tau) = [G(\tau)]^2 \tag{A18}$$

where $G(\tau)$ is defined in (12)

Thus, the differential equation for $r(\tau)$ is obtained. The initial conditions for this differential equation can be obtained readily from (A14) as

$$r(\tau = 0) = r(0) = 0,$$

The procedure to obtain other equations in the text is similar. A more rigorous and elaborate proof can be found in Reference 10.