Study and Implementation of Variable Time Step Integration Using Gear’s Method for Transient Analysis

Romesh Kumar Nandwana
Department of Electrical and Computer Engineering
Oregon State University, Corvallis, OR, USA
Email: nandwanr@eecs.oregonstate.edu

Abstract—In this project variable time step integration method has been successfully implemented using Gear’s method and interfaced with an basic SPICE-like simulator myspice. The order of integration is user defined and can be set between 1 to 6. the integration method been tested with stiff circuits and relative performance comparison has been done between fixed time step trapezoidal method and different order’s gear’s method.

Keywords: Stiff-Circuits, Gear’s Method, BDF, Predictor-Corrector

I. INTRODUCTION

Transient analysis is one of the most popular simulation method and have wide applications. In every SPICE based or SPICE-like simulator it is an important tool for circuit analysis where we use some integration method and based on the previous time point values we can calculate the next time point values. In a normal spice simulation Trapezoidal method with fixed time step is used as default integration method which is very fast and efficient method for integration. It have good accuracy and stability relative to the other methods but have some limitations also. Trapezoidal ringing is one of them in which as we increase the time step values the method tends to ringing problem because the method is not an L-Stable method.

It creates problem when we try to solve the Stiff circuits with large time steps using TR method. Stiff circuits are the circuits which have widely varying time constants. In this type of circuits either the circuit elements have a vast difference between time differences or the circuit have only one time constant but the source in the circuit have varying time step. For example if have a circuit with time constant have large difference as shown in figure 1. Where \( \lambda_1 \) and \( \lambda_2 \) have large difference.

When we use TR method FOR the stiff problem with large time steps it creates ringing problem or if we choose very small time steps for capturing initial transient then it will take very long time to complete the simulation. So here it is the accuracy and stability trade-off for stiff circuits.

To solve Stiff circuits we need a method which can have small time step for initial transients and large time steps when the change in the circuit node voltages is very small. There are many methods available for solving stiff circuit equations like explicit and implicit Runge-Kutta Method, Forward Euler Method, Backward Euler Method, Trapezoidal with variable time step or Gear’s Method. In the above mentioned method some of them have some advantages and disadvantages as mentioned below:

- when using forward Euler on stiff circuits the time-step must be chosen small relative to the fastest time constant to avoid stability problems. For this reason, forward Euler is not efficient when applied to stiff circuits.
- Implicit Runge-Kutta Method have larger computation time relative to the linear multi-step methods.
- Backward Euler method have numerical damping issues for oscillator circuits.

Because of the above reasons we choose gear’s Method for solving stiff circuit problems. Gear’s Method is stiffly stable method up to 6th order and thus very useful for us and it is also easy to implement.

II. GEAR’S METHOD

For solving stiff problems gear has given an integration method which uses the basic linear multistep integration formula. A linear multistep method which uses the previous P time steps to calculate the current time point value \( x[n] \) can be written as

\[ \Sigma_{i=0}^{k} \alpha_i x_{n-i} + h \Sigma_{i=0}^{k} \beta_i x_{n-i} = 0 \]

This linear multistep method is said to be exact for order K, then we can say that it will give the exact solution for any polynomial which have the maximum degree up to K.
Here $\alpha$ and $\beta$ values are calculated from exactness constraints.

For gears method we put some additional constraints on this linear multistep formula as number of previous time points used in the integration are made equal to the the order of the integration i.e., $K=P$ and we put previous time point derivative coefficients equal to zero

$$\beta_1 = \beta_2 = \beta_3 = \ldots \beta_n$$

and the formula can be modified as

$$\Sigma_{i=0}^{k} \alpha_i x_{n-i} - h \dot{x}_n = 0$$

Here $h_n$ is the time step taken to go from time $t_{n-1}$ to $t_n$. And $\beta_0$ is considers as -1 and rest P+1 $\alpha$ values are calculated from p+1 exactness constraints and

$$\alpha_0 = h_n \Sigma_{i=1}^{k} \frac{1}{t_n - t_{n-i}} \text{ and for } 1 \leq j \leq k \text{ } \alpha_j = h_n \frac{1}{\Pi_{l=1}^{k}(t_n - t_{n-j})} \frac{t_{n-j} - t_{n-i}}{t_{n-j} - t_{n-i}}$$

the local error of the integration method can be given as

$$LE_n = C^{k+1} h_n^{k+1} \dot{x}^{k+1}(t_n)$$

Now from these alpha and beta values and with the help of previous time point values we can write the derivative value as

$$\dot{x}_n = \alpha x_n + \beta$$

this is stiffly stable method up to $6^h$ order and can be used in the circuit analysis.

For higher order integration method we need more previous time point values in compare to the lower order method. So we cant start directly from a higher order method. For the first time step first order integration method is selected and then gradually the order is increased in subsequent steps up to we reach to the user defined order.

### III. Time Step Control:

For using the gears method one important part is to how to determine the next step size. The time step should be selected in such a manner that we can minimize the number of time points for calculation. At the same step time size should be consistent so that the local error in the computation should remain smaller then a user specified limit.

In spice the limit is defined in the terms of $E_n$ and we need to put the constraint in such a way that

$$LE_n \leq E_n$$

where

$$LE_n = C^{k+1} h_n^{k+1} \dot{x}^{k+1}(t_n)$$

and

$$E_n = \epsilon_A + \epsilon_{R\text{MAX}} (|x_n|, |x_{n-1}|)$$

from here we can say that

$$h_n \leq \left( \frac{E_n}{|C_{k+1}| \dot{x}^{k+1}(t_n)} \right)^{\frac{1}{k+1}}$$

Here we know the values of $C_{k+1}$ and $E_n$ for any time point. And for calculating $\dot{x}^{k+1}$ divided difference formula can be used but it can be a little bit difficult to implement.

So we use the alternative approach for time step prediction. We can define a new factor since

$$LE_n \leq E_n$$

so

$$r = \frac{LE_n}{E_n}$$

where

$$LE_n = C^{k+1} h_n^{k+1} \dot{x}^{k+1}(t_n)$$

and

$$E_n = C^{k+1} h_n^{k+1} \text{allowable} \dot{x}^{k+1}(t_n)$$

from here we can say that

$$r = \left( \frac{h_n}{h_{\text{allowable}}} \right)^{\frac{1}{k+1}}$$

and from here

$$h_{\text{allowable}} = h_n x_{rLE}$$

where

$$r_{LE} E = r^{\frac{k+1}{k+1}}$$

From here we can calculate the maximum allowable time step. If we have more then energy storage elements in the circuit then we calculate $r$ for all of them and by taking the norm we can get an optimum value that can be used for calculation.

### IV. Implementation

Implementation of the gears method it is very similar to a practical spice like simulator.

At first the circuit description is read by the simulator which also include the transient simulation options.

The user gives us TSTART, TSTOP, TMAX and order of integration.

Then the simulator set up the different data structure for all the components which are present in the circuit. After the setup the DC operating point is calculated at starting time point using Newton- Raphson method.

Now we define some terms with the help of user defined transient simulation options as

- DELMAX= tstop-tstart/50;
- DELMIN=10^-9*DELMAX;

Now for the first time point we choose the time step as DELMIN and then calculate the integration coefficients for first order. From here the value of derivative for energy storage elements is considered as
\[ x_n = \alpha x_n + \beta \]

And now this is stamped into the circuit matrix and solved iteratively with the help of Newton's method to linearise the nonlinear components. Then the circuit equations are assembled using Modified Nodal Analysis and solved with LU factorization.

If the circuit node values converge then we compute the local error \( LE_n \) and then check whether it is in the specified limit or not.

If local error is within the acceptable limit then we predict the next time step value. The next time step prediction and local error calculation is done in single step using Predictor-Corrector Method. The local error is calculated using the formula

\[ LE_n = \frac{h_n}{t_n - t_{n-k-1}} (x^n_n - x^n_{n-k}) \]

where \( x^n_n \) is the predicted value at time point \( n \) and \( x^n_{n-k} \) is the corrected values using the Newton iteration at time point \( n \).

Using \( r_{LE} \) value we check if \( r_{LE} \leq 0.9 \) then the time step \( h_n \) is rejected and we reiterate the whole process using \( h_{n+1} = r_{LE} * h_n \). If \( r_{LE} \geq 0.9 \) then \( h_n \) is accepted and we predict the next time step using

\[ h_{n+1} = \text{MIN} (r_{LE} * h_n, 2 * h_n, \text{DELMAX}) \]

After this we predict the value of the \( x_n \) at next time point using the predictor polynomial of order \( K \) where \( K \) is the order of integration and for prediction we use previous \( K+1 \) time points.

The predictor Lagrange polynomial is the given by Lagrange interpolation/extrapolation polynomial

\[ x^n_{n+1} = \sum_{i=1}^{k+1} \gamma_i x_i \]

where

\[ \gamma_i = \prod_{j=1, j \neq i}^{k} \frac{t_n - t_{n-i}}{t_n - t_{n-j}} \]

Now from predicted time step and predicted values the time point is moved further and we again calculate the coefficients of integration and calculate the solution until we reach to the stop time.

V. RESULTS

The algorithm has been successfully implemented and tested on a typical stiff circuit shown in figure-2. In this circuit we can have two widely varying time constants with by putting the values of \( C_1 = 1 \mu F, C_2 = 100 \mu F, R_1 = R_2 = 1 \mathrm{K\Omega} \) and \( V_1 = 5 \mathrm{V} \).

![Figure 2. test circuit Schematic](image)

The circuit has been tested for 5sec time period with fixed time step \( TR \) and gears method of order 1 to 6 and voltages at node 1 and 2 has been observed. Table-1 shows the number of steps required to achieve the acceptable error Local error and Figure-3 shows the voltage variation at Node-1 in initial fast transient and figure-4 shows the voltage variation at Node-2 in the slow transient.

<table>
<thead>
<tr>
<th>Method</th>
<th>Step Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>TR</td>
<td>5000</td>
</tr>
<tr>
<td>Gear-1</td>
<td>2500</td>
</tr>
<tr>
<td>Gear-2</td>
<td>1693</td>
</tr>
<tr>
<td>Gear-3</td>
<td>1293</td>
</tr>
<tr>
<td>Gear-4</td>
<td>1057</td>
</tr>
<tr>
<td>Gear-5</td>
<td>902</td>
</tr>
<tr>
<td>Gear-6</td>
<td>793</td>
</tr>
</tbody>
</table>

![Figure 3. Voltage at Node-1 using different integration Methods](image)

![Figure 4. Voltage at Node-3 using different integration Methods](image)

VI. CONCLUSION

The variable time step method has been successfully implemented and tested on a stiff problem and Step count and the error in the integration has been observed from the results. The method can be further extended to variable time step
variable order method where order will not be user defined and simulator can itself find the optimal order.

REFERENCES


