ACTIVITY CHARACTERIZATION OF SPATIAL MODELS: APPLICATION TO DISCRETE EVENT SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS

By

Rajanikanth Jammalamadaka

A Thesis Submitted to the Faculty of the

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

In Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE
WITH A MAJOR IN COMPUTER ENGINEERING

In the Graduate College

THE UNIVERSITY OF ARIZONA

2 0 0 3

STATEMENT BY AUTHOR

SIGNED: _____

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

_____ _____

Bernard P. Zeigler Date

Professor of Computer Engineering

ACKNOWLEDGEMENTS

I would like to thank my parents and my brother for their love and continued support through out my life. I am what I am because of my parents.

I would like to thank my advisor Dr. Bernard Zeigler for his guidance and encouragement throughout this endeavor. Without him, this thesis would not have been a success. It was a very good experience working with him.

I would like to thank Dr. Salim Hariri and Dr. Kevin Mc Neill for reviewing my thesis and for serving on my thesis defense committee.

I would like to thank Jim Nutaro for patiently answering many of my questions.

I would like to thank Dr. Ernesto Kofman for reviewing my thesis.

I would like to thank my lab mates Dr. Sun Woo, Saehoon, Xiaolin, Cseo, Saurabh, Salil, Mahesh, and Narayanaswami for creating a friendly and entertaining working environment.

I would like to thank all my friends at the University of Arizona for making my stay memorable.

TO MY PARENTS

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

ABSTRACT

Recent research in modeling and simulation suggests that potential speed advantages with discrete event simulation are to be expected for partial differential equation models that are characterized by heterogeneity in their behavior in time and space. In such cases, discrete events are a natural way to focus attention on the portions of the solution that are exhibiting high activity levels at the moment. Theory also suggests a way to characterize the activity of solutions over time and space independently of the solution technique that might be employed. This activity measure, when divided by a quantum size, predicts the number of boundary crossings (computations) required by a DEVS simulator for the accuracy afforded by that quantum size. Where significant heterogeneity of activity exists, the number of discrete event computations may be significantly lower than that required by a uniform allocation of computational resources across both space and time.

In this thesis, the theory of activity is developed for the particular problem of diffusion and the execution times of DEVS and a standard discrete time method are evaluated and compared under the conditions of stability and accuracy.

1. INTRODUCTION

As we step into the 21st century, the need for computationally efficient methods is ever increasing. Discrete Event System Specification (DEVS) was one of the methods devised in order to meet these ever-increasing needs. In DEVS, we define a threshold called a quantum, and perform the processing only if this quantum is exceeded. The accuracy of the output depends upon this quantum. An important concept related to DEVS that is been recently developed is "Activity" (see chapter 2). Classical processing takes place in discrete time, where, in order to model any process, all of the cells need to be scanned at every time step even though there are no interesting events at most of the cells. In contrast DEVS concentrates its computational resources at the regions of high activity.

In this thesis, the problem of diffusion was modeled in DEVS and discrete time, and the resulting performances were compared. This chapter introduces the problem of diffusion and the initial conditions that were used to model the problem in DEVS and discrete time.

Presented Work

Chapter 1 discusses the initial conditions used to model the problem of diffusion. Chapter 2 introduces the theory of activity. Chapter 3 discusses the DEVS formalism. Chapter 4 discusses about the DEVS integrator. Chapter 5 discusses about the performance comparison of DEVS and discrete time. Chapter 6 presents the experimental results. Conclusion and scope for future work are presented in chapter 7.

1.1. Diffusion

Diffusion is the movement of heat molecules from a region of high concentration to a region of low concentration.

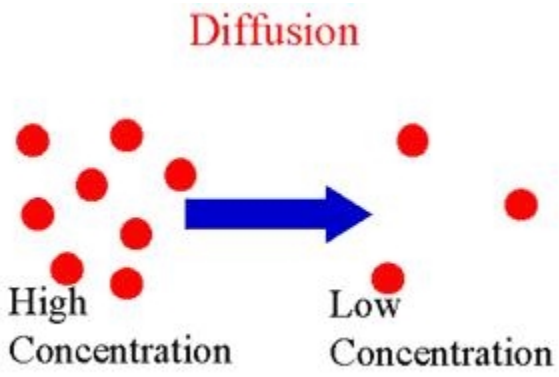

Figure 1 Definition of diffusion

1.2. Partial differential equations

Differential equations are those equations where the unknowns are functions of one or more variables and which contain not only these functions but their derivatives as well. If the unknowns are functions of several variables (not less than two), then the equations are called partial differential equations [4].

A partial differential equation containing derivatives of the unknown function u with respect to the variables $x_1,.....,x_n$ is said to be of Nth-order if it contains at least one Nth-order derivative and does not contain derivatives of higher orders, that is, the equation

$$\Phi\left(x_1,.....x_n,u,\frac{\partial u}{\partial x_1},.....,\frac{\partial u}{\partial x_n},\frac{\partial^2 u}{\partial x_1^2},\frac{\partial^2 u}{\partial x_1 \partial x_2},.....,\frac{\partial^N u}{\partial x_n^N}\right)=0 \qquad Eqn \quad 1.1$$

Equation 1.1 is said to be linear if Φ, as a function of $u,\frac{\partial u}{\partial x_1},.....,\frac{\partial^N u}{\partial x_n^N}$, is linear.

1.3. Method of Lines

In one class of method for solving partial differential equations, one of the variables, say x is discretized, while the other variable t is left continuous. When suitable finite-difference expressions are substituted for the x derivatives, the partial differential equation is converted into a coupled system of ordinary differential equations in the independent variable t, i.e., a difference-differential equation. This method is called the method of lines [5].

The 2-d diffusion equation is:

$$u_t = -au_{xx} - bu_{yy} \qquad Eqn \quad 1.2$$

where u_t is the first order partial differential of the state variable u with respect to time, a, b are the diffusion constants and u_{xx}, u_{yy} are the second order partial derivatives with respect to space (x and y directions respectively).

Using the method of lines equation 1.2 can be written as

$$u_t = -a\left(\frac{u(x+\Delta x)-2u(x)+u(x-\Delta x)}{(\Delta x)^2}\right)-b\left(\frac{u(y+\Delta y)-2u(y)+u(y-\Delta y)}{(\Delta y)^2}\right) \qquad \text{Eqn 1.3}$$

where $\Delta x, \Delta y$ are the spatial resolutions in the x and y directions respectively.

Using the FTCS (Forward Time Centered Space) method, this equation can be approximated as follows:

$$u_{m,n}^{i+1} = u_{m,n}^{i} - \Delta t\left(a\left(\frac{u_{m+1,n} - 2u_{m,n} + u_{m-1,n}}{(\Delta x)^2} \right) - b\left(\frac{u_{m,n+1} - 2u_{m,n} + u_{m,n-1}}{(\Delta y)^2} \right) \right) \quad Eqn \quad 1.4$$

where $u_{m,n}^{i+1}$ updates the value of u for each iteration.

For simplicity, assume a = b = c, $\Delta x = \Delta y = h,$ then equation 1.4 becomes

$$u_{m,n}^{i+1} = u_{m,n}^{i} - \left(\frac{c\Delta t}{h^2} \right)\left[u_{m+1,n} + u_{m-1,n} + u_{m,n+1} + u_{m,n-1} - 4u_{m,n} \right] \quad Eqn \quad 1.5$$

In the equation 1.5, care must be taken so that the COURANT-FRIEDRICH-LEWY condition [7] $\frac{c\Delta t}{h^2} < 0.5$ is satisfied. This means that for a given spatial cell size, h, the time step must be small enough to avoid instability. Actually, as the cell size h decreases, the time step must decrease as the square of h to maintain stability.

So, conceptually, the 2-D diffusion equation is equivalent to saying that each cell tries to reduce the difference (of the quantity which is diffused) between itself and its neighbors.

The following diagrams figuratively illustrate the 2-D diffusion process:

$$u_t = -cu_{xx} - cu_{yy}$$

$$\xleftarrow[\Delta y \to 0]{\Delta x \to 0} \frac{du(x)}{dt} = -c/h^2 * [avg(u(x-\Delta x), u(x+\Delta x), u(y-\Delta y), u(y+\Delta y)) - u(x,y)]$$

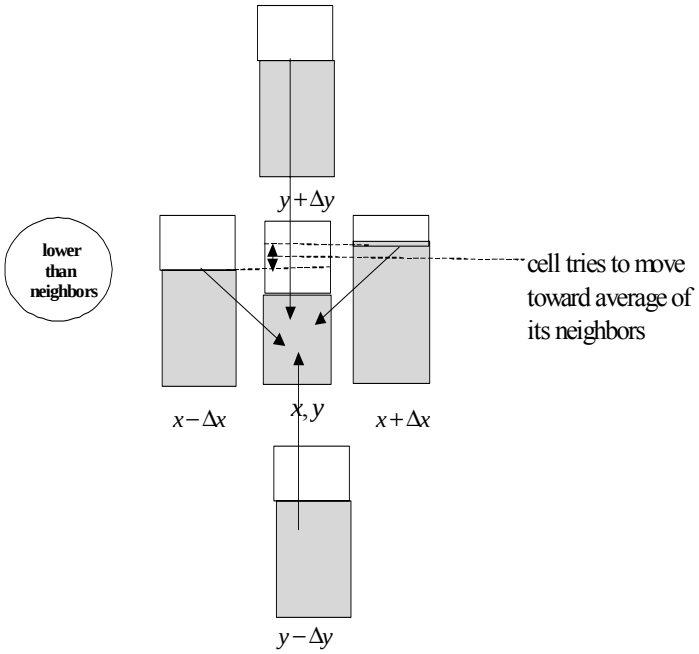

Figure 2 Illustration of two-dimensional diffusion 1

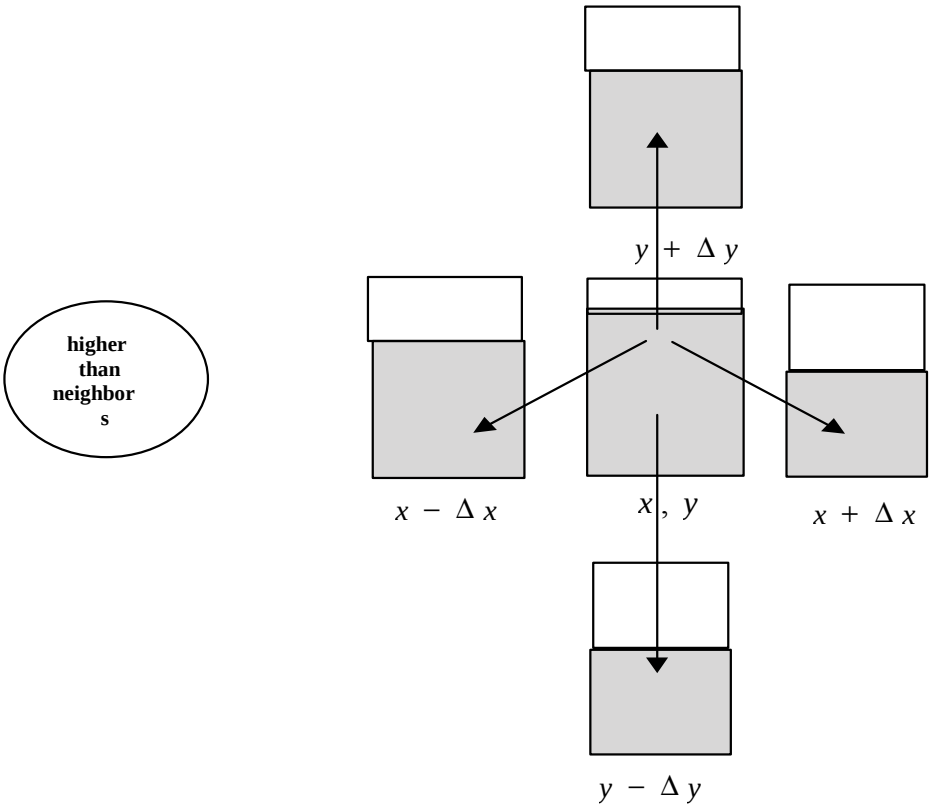

Figure 3 Illustration of two-dimensional diffusion 2

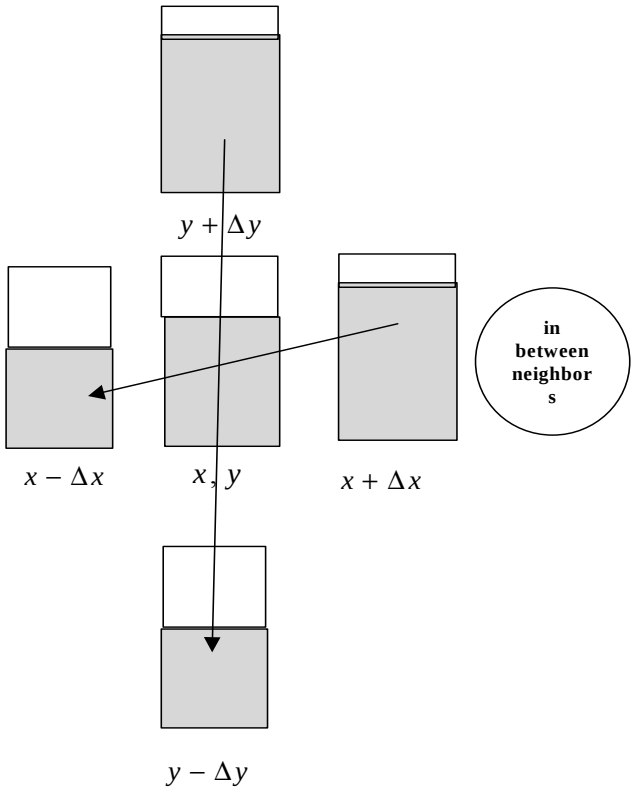

Figure 4 Illustration of two-dimensional diffusion 3

1.4. Initial Conditions

To start with, a plate that has a high value of heat at its center is constructed. This is as shown below:

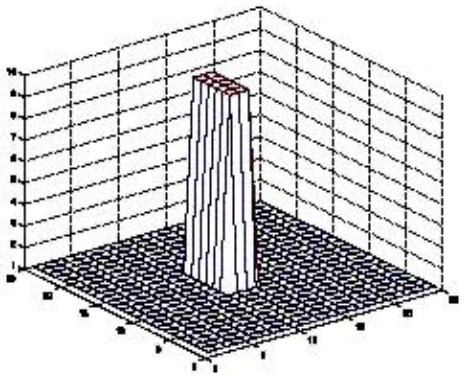

Figure 5 Pulse for 2-D diffusion

Since we are starting from a sharp pulse at the center of the plate, all the derivatives will be infinitely large.

Therefore, running the simulation for an interval of 2.4 seconds smoothed the initial pulse and this Gaussian pulse was used as the initial data. The smoothed pulse is as shown below:

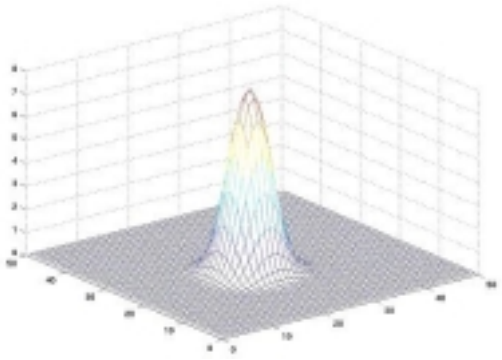

Figure 6 Initial condition for two-dimensional diffusion

1.5. Boundary Conditions

To make the diffusion equation well posed, we need to specify the boundary conditions at $x = y = 0$ and $x = y = N$ (where N is the number of cells in each direction). The two main cases of interest are

•Constant boundary conditions

$u(0,0,t) = u(\Delta x, \Delta y, t)$ *and* $u(N\Delta x, N\Delta y, t) = u((N-1)\Delta x, (N-1)\Delta y, t)$, where the cells 1, 2… N-1 are in the interior of the computational region, and the cells 0 and N are at the boundaries. In this case there is no loss in the net heat value i.e. the heat energy is conserved in this case.

•No-flux boundary conditions

$$\left(\frac{\partial u}{\partial x}\right)_{x=0} = \left(\frac{\partial u}{\partial y}\right)_{y=0} = \left(\frac{\partial u}{\partial x}\right)_{x=N} = \left(\frac{\partial u}{\partial y}\right)_{y=N} = 0.$$

Such boundary conditions correspond to having no flux out of the domain.

If simulated for enough time (500 seconds), the pulse will approach a flat plate, which is as shown below:

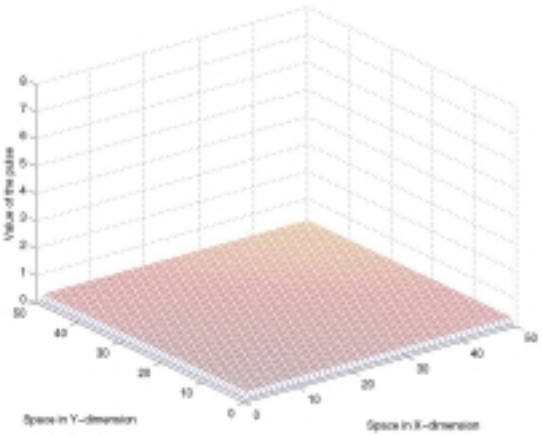

Figure 7 Final state of two-dimensional diffusion

2. ACTIVITY

Introduction

The following is a brief introduction to the concept of activity. As we discussed in chapter one, we define a quantum or threshold for processing in DEVS. A cell is said to be most active if the value of the cell crosses the quantum more times than any of the other cells. In the particular case of diffusion, the most active cells are at the center of the plate since the whole of the Gaussian pulse is concentrated at the center of the plate. As the Gaussian pulse diffuses, the activity moves from the center of the plate to its boundaries because then there will be more cells that are active at the boundaries than the cells at the center of the plate. DEVS has the advantage over discrete time processing when there is high heterogeneity in the time and space phenomena.

2.1. Definition

The following is the definition of activity for a continuous segment.

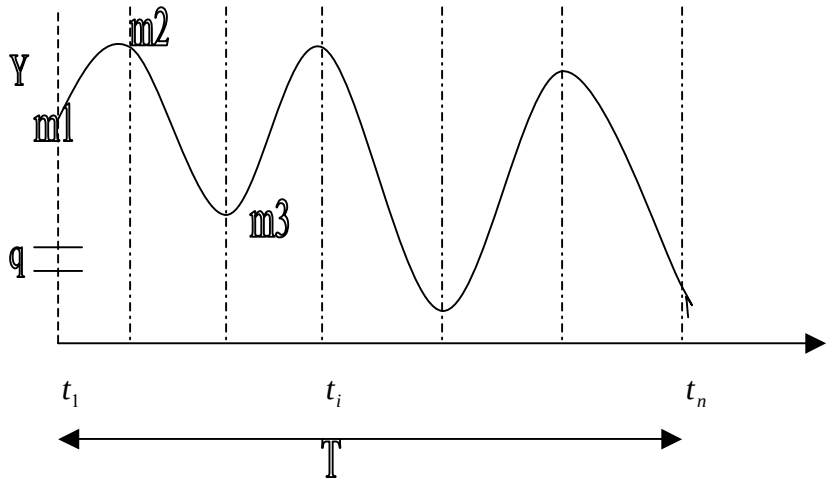

Figure 8 Definition of Activity

In the above figure, q corresponds to the quantum (the minimum threshold below which no processing occurs) and m_i corresponds to the maxima and minima of the curve, where the first and last m_i are the values of the function at the initial and final end points.

The *activity* in an interval [0, T] is defined as

$$A(T) = \int_0^T \left| \frac{\partial Y}{\partial t} \right| \partial t$$

Proposition 1. The activity in an interval [0, T] can be calculated by summing the differences between the adjacent maxima and minima, i.e.

$$Activity(T) = \sum_i \left| m_{i+1} - m_i \right|$$

Proof:

In an interval between successive turning points the derivative of the function does not change sign. If the sign is positive in the interval we have

$$\begin{aligned} A(t_i, t_{i+1}) &= \int_{t_i}^{t_{i+1}} \left| \frac{\partial Y}{\partial t} \right| \partial t \\ &= \int_{t_i}^{t_{i+1}} \frac{\partial Y}{\partial t} \partial t \\ &= Y(t_{i+1}) - Y(t_i) \\ &= m_{i+1} - m_i \end{aligned}$$

A similar argument holds for an interval in which the derivative is always negative. ■

The average activity in an interval [0, T] is given by:

$$AvgActivity(T) = Activity / T$$

Proposition 2. The number of threshold crossings in an interval of length T for threshold levels that are equally spaced by quantum size, q, is:

$$NumberOfThresholdCross(T,q) = Activity(T)/q$$

Proof: We will make use of the intermediate value theorem [6] in order to prove the proposition. The intermediate value theorem (IVT) states that:

"If a function f is continuous on a closed interval [a, b] and C is any number between f (a) and f (b), inclusive, then there is at least one number x in the interval [a, b] such that f (x) = C".

Consider the following figure

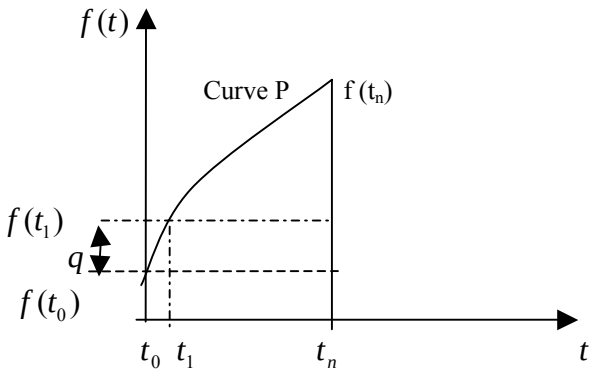

Figure 9 Proof for proposition 2

From the intermediate value theorem, we have

$$f(t_1) = f(t_0) + q \qquad \text{Eqn 2.1}$$

We will prove that equation 2.1 holds in the general case using mathematical induction i.e. we will prove that

$$f(t_n) = f(t_{n-1}) + q \qquad \text{Eqn 2.2}$$

For the case of n=1 equation 2.2 is obvious (from equation 2.1).

Let us assume that equation 2.2 is true for some arbitrary k greater than one i.e. let us assume

$$f(t_k) = f(t_{k-1}) + q \qquad \text{Eqn 2.3}$$

We have to prove that

$$f(t_{k+1}) = f(t_k) + q \qquad \text{Eqn 2.4}$$

The right hand side of equation 2.4 is equal to

$$f(t_k) + q = f(t_{k-1}) + q + q \quad \text{(from equation 2.3)}$$

⇨ R.H.S of equation 2.4 is equal to $f(t_{k-1})+2q$ which is equal to $f(t_{k+1})$

Since we assumed that the threshold levels are equally spaced by quantum size q.

Therefore, if we traverse along the curve P from $f(t_0)$ to $f(t_n)$ we will have to cross each quantum level of size q. In fact, the number of times we have to cross a quantum q (number of threshold crossings) is equal to $\left(\frac{f(t_n) - f(t_0)}{q} \right)$ which is equal to $A(t_n)/q$ where $A(t_n)$ is the activity over a time interval t_n. ■

Proposition 3. The average activity in an interval is related to the average absolute value of the derivative in the interval

$$Average \quad Derivative(t_i, t_{i+1}) = \left| m_{i+1} - m_i \right| / (t_{i+1} - t_i)$$

Proof: follows directly from the definition of activity and Proposition 1.

2.2. Significance of the activity concept for DEVS simulation of PDEs

The concept of partial differential equations was introduced in section 1.2 and the concept of activity was introduced in section 2.1. This section discusses the significance of activity as it pertains to DEVS [see chapter 3] and discrete time processing.

In the particular example of diffusion (Figure 5), initially the total amount of heat is concentrated at the center of the plate and therefore the value of activity is also high at the center of the plate. As the heat diffuses to the boundary of the plate, the activity at the center of the plate decreases, and the activity moves towards the boundaries.

In the approximation to the method of lines using quantization supported by DEVS, we define thresholds (separated by quanta) and processing is done only when the change in the value at a particular cell exceeds this threshold. In other words, quantization is a method that allows the DEVS simulator to track the activity of cells in a method-of-lines solution. At any time, the time advances of cells are distributed over a range varying inversely to their activity rates (absolute derivatives) at that time. Thus the DEVS simulator uses a smaller time advance (equivalent to time step in discrete time) in the regions where there is high activity, so that processing is fast at those regions. Conversely, cells in low activity regions have relatively long time advances and therefore do not needlessly consume simulator resources. On the other hand, in discrete time, processing takes place at all the cells, at every time step, even though there are no interesting events at most of those cells. Also many methods, such as FTCS, use a uniform time step irrespective of the activity, which causes them to be less efficient when compared with a quantized DEVS implementation.

2.3. Derivation of Activity for various initial conditions

In this section we will illustrate the utility of the activity concept for predicting the number of transitions required by a quantized method-of-lines DEVS solution. Activities over intervals can be obtained directly by simulation or analytically as the following examples show. In these examples, various initial data are considered with closed boundary conditions so that heat is conserved.

2.3.1. Activity for a rectangular pulse as the initial data

Consider a pulse of height H, and width, W in a one-dimensional space of length L, as illustrated in Figure 10.

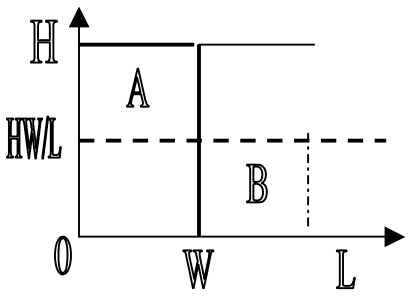

Figure 10 Initial condition as a rectangular pulse

The initial area (heat) under the pulse is HW. When the base of the pulse expands to L, this area remains the same (by conservation of heat), so at equilibrium the pulse has a uniform value of HW/L throughout the space. In this case, as time progresses, the height of the rectangle marked A goes down from H to HW/L and the height of the rectangle marked B increases from 0 to HW/L. An intermediate stage in this process is as shown in figure 11.

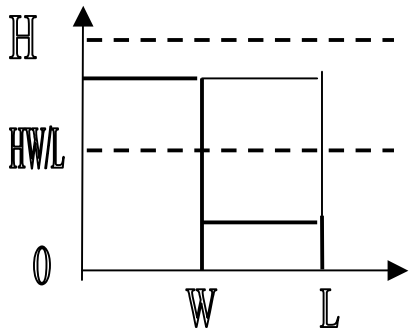

Figure 11 Intermediate condition as a rectangular pulse

Figure 12 shows the time plot of a representative cell, which is in rectangle marked A of figure 10 as it moves from H to HW/L.

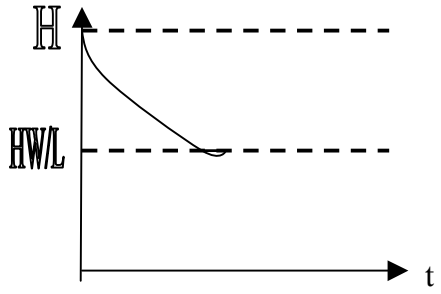

Figure 12 Time plot of a representative cell in rectangular pulse

Note that the curve monotonically approaches equilibrium so that the activity is just the difference between the initial height and the equilibrium value.

Let $\Delta x = \frac{L}{N}$ be the spatial resolution and N be the total number of cells.

At equilibrium, each cell will have a uniform value of HW/L. There are $\frac{W}{\Delta x}$ cells in a width of W, which move from H to HW/L and $\frac{L-W}{\Delta x}$ cells in a width of L-W, which move from 0 to HW/L.

Therefore, activity over the (infinite) period of evolution to equilibrium is given by

$$
\begin{aligned}
Activity &= \left| H - \frac{HW}{L} \right| \left(\frac{W}{\Delta x} \right) + \left| 0 - \frac{HW}{L} \right| \left(\frac{L - W}{\Delta x} \right) \\
&= \frac{2HNW\ (L - W)}{L^2} \\
&= 2HNw\ (1 - w) \quad (w = W\ /\ L) \qquad Eqn \quad 2.5
\end{aligned}
$$

Note that the total activity goes to zero as the width of the pulse relative to the size of the space under consideration goes to zero as well as to 1. In both extremes, the region of maximum activity is confined to a smaller and smaller region. Since the number of transitions in a DEVS solution is proportional to the activity, we see that it should be able to exploit such heterogeneity of activity, unlike a discrete time approach, which is not sensitive to the relative size of the pulse in the computation domain. Also note that the activity per cell (A/N) is invariant with respect to N, the number of cells employed in the simulation. This is a feature we will see in the ensuing examples as well.

2.3.2. Activity for a triangular pulse as the initial data

Consider an initial state in which heat increases linearly to a height H, over an interval of length, L.

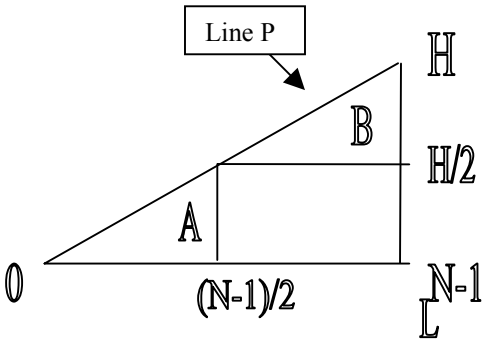

Figure 13 Initial condition as a triangular pulse

In this case, as time progresses, the height of the triangle B descends to the equilibrium and the height of the triangle A increases. At equilibrium, each cell will have a uniform value of H/2.

An intermediate stage in this process is as shown below.

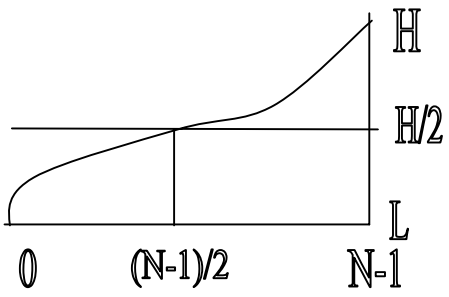

Figure 14 Intermediate condition as a triangular pulse

So, in the case of triangular pulse each cell travels a distance, H/2 to equilibrium.

From figure 13 we have the equation of line P as

$$y(i) = \left(\frac{H}{N-1}\right) i$$

where i = 0,1,2,… is the index of the cell (note that the length of the space does not enter in this formulation).

Therefore the total activity is given by

$$\begin{aligned} Activity &= \sum_{i=0}^{N-1} \left| \frac{H}{2} - \left(\frac{H}{N-1}\right) x \right| = \left|\frac{H}{2}\right| + \left|\frac{H}{2} - \frac{H}{N-1}\right| + \ldots\ldots + \left|\frac{H}{2} - H\right| \\ &= \frac{H}{2}\left(1 + 2\left(\frac{2}{N-1} + \frac{4}{N-1} + \ldots\ldots \frac{N-3}{N-1}\right)\right) \\ &= \frac{H}{2}\left(1 + \frac{4}{N-1}\left(1 + 2 + 3 + \ldots\ldots \frac{N-3}{2}\right)\right) \\ &= \frac{H}{2}\left(1 + \frac{N-3}{2}\right) = \frac{H}{4}(N-1) \qquad Eqn \quad 2.6 \end{aligned}$$

Note that the activity, in this case, is independent of L, the length of the space in contrast to the inverse dependence of the rectangular pulse's activity. This is explained by the fact that in this case the regions of maximum activity remain at the boundaries independently of the length. Also note as N increases, the activity per cell approaches a constant, as in the rectangular pulse example.

2.3.3. Activity for a Gaussian pulse as the initial data

We consider the case of a Gaussian pulse as the initial data for the case of open boundary conditions where the heat of the system tends to zero.

The diffusion equation for one dimension is expressed as

$u_t = -cu_{xx}$ where c is the diffusion constant. The solution to this equation is given by

$$u(x,t) = \left(\frac{H}{\sqrt{4*\pi*c*t}}\right)*\exp\left(\frac{-x^2}{(4*c*t\ )}\right) \quad \textit{where H is the Initial height of the pulse}$$

The partial derivative with respect to time is:

$$\frac{\partial u(x,t)}{\partial t} = \left(\frac{H}{\sqrt{4*\pi*c*t}}\right)*\exp\left(\frac{-x^2}{4*c*t}\right)*\left(\frac{x^2}{4*c*t^2}\right)-\left(\frac{Ht^{\frac{-3}{2}}}{\sqrt{16*\pi*c}}\right)*\left(\exp\left(\frac{-x^2}{4*c*t}\right)\right) \quad \textit{Eqn}\ \ 2.7$$

For Maxima or Minima at location, x,

$$\frac{\partial u(x,t)}{\partial t} = 0 \Rightarrow t^*(x) = \frac{x^2}{2*c}$$

At $t^*(x)$ $\frac{\partial^2 u(x,t)}{\partial t^2}$ *is negative, therefore*

$u(x,t)$ *is Maximum at* $t^*(x)$

Evaluating $u(x,t)$ *at* $t = t^*(x)$,

$$u_{\max}(x) = \frac{H}{x*\sqrt{2*\pi*\varepsilon}} \quad \textit{Eqn}\ \ 2.8$$

Now consider the following figure:

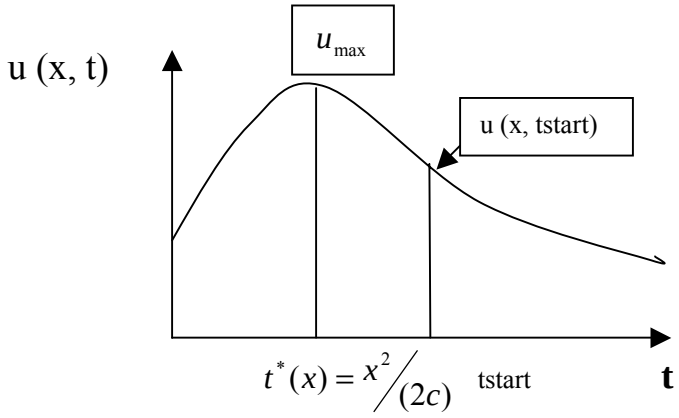

Figure 15 Case 1 - the cell sees no crest

In section 1.4, we discussed that the initial pulse was smoothened to get a Gaussian pulse and this pulse was used for the simulation runs. In fact, it is not possible to start with t equal to zero, since the Gaussian pulse becomes a Dirac delta function concentrated at the origin as t approaches zero. So in the following analysis, we start from a state in which t is equal to tstart, some time greater than zero.

We consider the cases where the crest of the wave at location x passes before, and after tstart, respectively. The first case (figure 15) plots the heat over time for the condition that tstart occurs after the pulse has reached its peak at position x. The activity in this case is given by u (x, tstart) because the pulse descends monotonically from u (x, tstart) to zero.

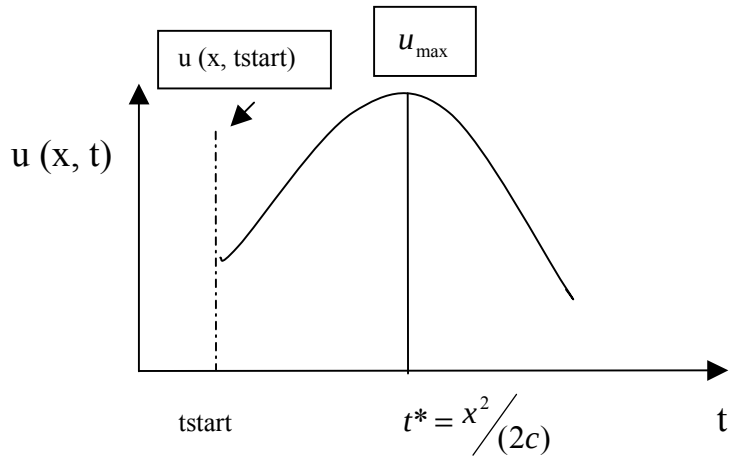

Figure 16 Case 2 - the cell sees a crest

The second case (figure 16) is for cells, for which tstart precedes the time when the pulse will reach it's maximum and then decay. The activity in this case is given by $\left|2u_{\max}(x) - u(x, tstart\ )\right|$ because the pulse has to rise from u (x,tstart) to $u_{\max}(x)$ and then fall from $u_{\max}(x)$ to zero.

The figures 15 and 16 are for a particular cell position x. The total activity is obtained by summing the activities of cells satisfying the conditions in figures 15 and 16. The cells in figure 15 are those for which tstart is greater than, or equal to, $t^* = x^2/(2c)$. Solving for the largest x for which this is true, we have $x_{max} = \sqrt{2 * c * tstart}$. The cells in figure 16 then range from $x = \sqrt{2 * c * tstart}$ to x = L where L is the length of the space.

Therefore, the total activity is given by:

$$A = \sum_{x=0}^{\sqrt{2*c*tstart}} u(x, tstart\ ) - \sum_{x=\sqrt{2*c*tstart}}^{L} u(x, tstart\ ) + 2 * \sum_{x=\sqrt{2*c*tstart}}^{L} u_{\max}(x) \qquad Eqn\ \ 2.9$$

$$A = \sum_{x=0}^{L} u(x,tstart) - 2* \sum_{x=\sqrt{2*c*tstart}}^{L} u(x,tstart) + 2* \sum_{x=\sqrt{2*c*tstart}}^{L} u_{\max}(x)$$

$$A = \sum_{x=0}^{L} u(x,tstart) + 2* \sum_{x=\sqrt{2*c*tstart}}^{L} (u_{\max}(x) - u(x,tstart))$$

$$A = N/L \ (\sum_{x=0}^{L} u(x,tstart)\Delta x + 2* \sum_{x=\sqrt{2*c*tstart}}^{L} (u_{\max}(x) - u(x,tstart))\Delta x)$$

As $N \to \infty$

$$A = N/L \ (\int_{0}^{L} u(x,tstart)dx + 2* \int_{\sqrt{2*c*tstart}}^{L} (u_{\max}(x) - u(x,tstart))dx$$

$$A = N/L \ (\int_{0}^{L} u(x,tstart)dx + 2* \int_{\sqrt{2*c*tstart}}^{L} u_{\max}(x)dx - 2*(\int_{0}^{L} u(x,tstart)dx - \int_{0}^{\sqrt{2*c*tstart}} u(x,tstart)dx) \ )$$

$$A = N/L \ (2* \int_{\sqrt{2*c*tstart}}^{L} u_{\max}(x)dx - \int_{0}^{L} u(x,tstart)dx + 2* \int_{0}^{\sqrt{2*c*tstart}} u(x,tstart)dx \ )$$

Using the error function,

$$erf\ (x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-z^2}\, dz$$

$$A = \frac{HN}{L}\left(\frac{2}{\sqrt{2\pi e}} \ln\left(\frac{L}{\sqrt{2 * c * tstart}} \right) - \frac{1}{2} erf\left(\frac{L}{\sqrt{4 * c * tstart}} \right) + erf(0.707) \right) \quad Eqn\ 2.10$$

$$\frac{A}{N} = \frac{H}{L}\left(\frac{2}{\sqrt{2\pi e}} \ln\left(\frac{L}{\sqrt{2 * c * tstart}} \right) - \frac{1}{2} erf\left(\frac{L}{\sqrt{4 * c * tstart}} \right) + erf(0.707) \right) \quad Eqn\ 2.11$$

This shows us that the average activity per cell when the initial condition is a Gaussian pulse tends to a constant as N tends to infinity.

Now $\frac{A}{N}$ decreases as *L* increases since the error function goes to a constant and the natural logarithm term grows more slowly than *L*.

2.4. The Behavior of Total Activity as N tends to Infinity

We conclude that for a Gaussian pulse the average activity per cell quickly approaches a constant as the number of cells increases where this constant is a decreasing function of the size of the computation domain

The following table compares the results obtained thus far as we let the number of cells employed in a simulation tend toward infinity.

TABLE 1. Activities for different initial conditions

	Activity	Activity/N	Activity/N as N→∞
Rectangular pulse	2HN(W/L)(1 –W/L)	2H(W/L)(1 –W/L)	2H(W/L)(1 – W/L)
Triangular pulse	(N-1)*H/4	(N-1)*H/ (4*N)	H/4
Gaussian pulse	$\frac{N*H}{L}\left(\frac{2}{\sqrt{2*\pi*e}}\ln\left(\frac{L}{\sqrt{2*c*tstart}}\right)\right.$ $\left.-\frac{1}{2}erf\left(\frac{L}{\sqrt{4*c*tstart}}\right)+erf(0.707)\right)$	$\frac{H}{L}\left(\frac{2}{\sqrt{2*\pi*e}}\ln\left(\frac{L}{\sqrt{2*c*tstart}}\right)-\right.$ $\left.\frac{1}{2}erf\left(\frac{L}{\sqrt{4*c*tstart}}\right)+erf(0.707)\right)$	Constant

In all cases, the activity is propositional to the initial height of the pulse, H.

In the case of the rectangular pulse the activity depends on the ratio, W/L of the width to the length of the space. The activity can easily be shown to be greatest where W = L/2 and to decrease to zero as the ratio approaches zero or 1. The behaviors of the Gaussian and rectangular pulses are consistent in that the results support the assertion that the

number of transitions in a DEVS quantized method-of-lines simulation should decrease as the initial heat is confined to a smaller and smaller part of the space. Both rectangular and Gaussian pulses contrast with the case of the triangular pulse, where the activity is independent of the length of the space, since the latter does not affect the size of the region in which the initial heat is confined.

3. DISCRETE EVENT SYSTEM SPECIFICATION

This chapter introduces the Discrete Event System Specification [1]. It also describes the hierarchical modular decomposition of DEVS.

3.1. Definition

A discrete event system specification (DEVS) is a structure

$$M = <X,S,Y,\delta int, \delta ext,\lambda,ta>$$

where

X is the set of input values

S is a set of states,

Y is the set of output values

$\delta int:S->S$ is the internal transition function

$\delta ext:Q\times X->S$ is the external transition function, where

$Q=\{(s,e)|s \in S, 0 \leq e \leq ta(s) \}$ is the total state set, e is the time elapsed since last transition

$\lambda: S->Y$ is the output function

ta is the set of positive reals with 0 and ∞

The interpretation of these elements is illustrated in the following two figures. At anytime the system is in some state, s. If no external event occurs, the system will stay in state s for time ta(s). ta(s) can also take on the values 0 and ∞. When ta(s) is zero, the stay in

state s is so short that no external events can intervene-we say that s is a transitory state. When ta(s) is ∞, the system will stay in s forever unless an external event interrupts its slumber. We say that s is a passive state in this case. When the resting time expires, i.e., when the elapsed time, e = ta(s), the system outputs the value, λ(s), and changes to state δint(s). Output is only possible just before internal transitions.

If an external event x ∈ X occurs before this expiration time, i.e., when the system is in total state (s,e) with e ≤ ta(s), the system changes to state δext(s,e,x). Thus, the internal transition function dictates the system's new state when no events occurred since the last transition. The external transition function dictates the system's new state when an external event occurs- this state is determined by the input, x, the current state, s, and how long the system has been in this state, e. In both cases, the system is then in some new state s' with some new resting time, ta(s'), and the same story continues.

3.2. Internal Transition/ Output Generation

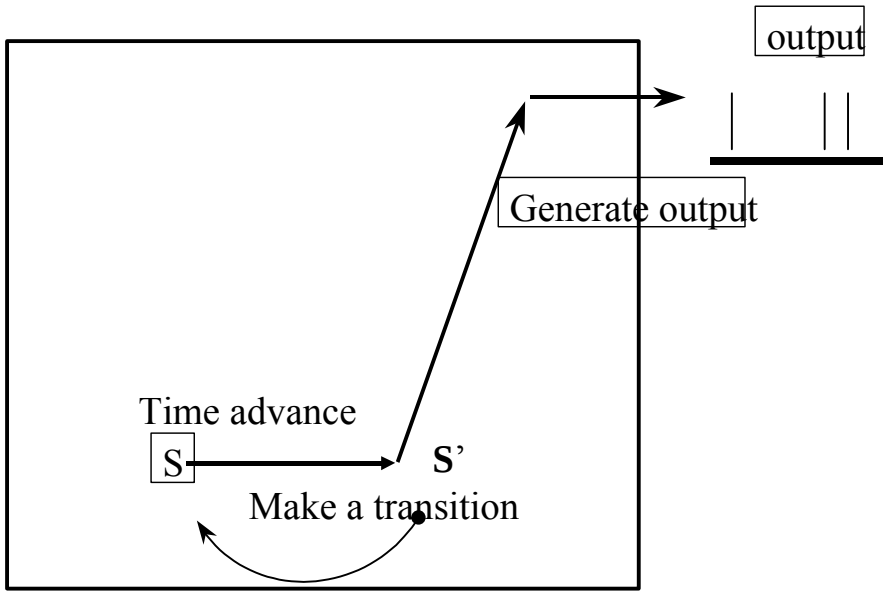

Figure 17 Internal Transition/ Output Generation in DEVS

3.3. Response to External Input

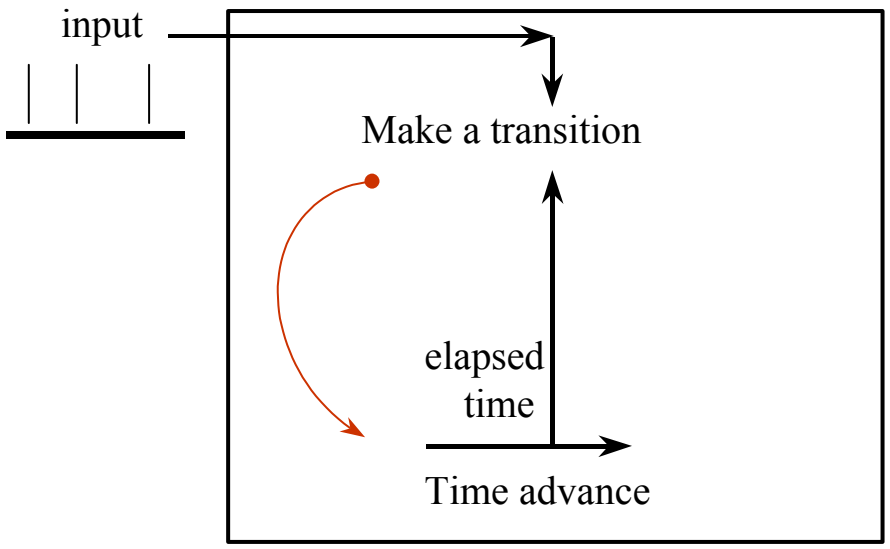

Figure 18 Response to External Input in DEVS

The behavior of DEVS can be depicted as in the figure 19.

Here the input trajectory is a series of events occurring at times such as t0 and t2. In between these event times may be those such as t1, which are times of internal events. The latter are noticeable on the state trajectory, which is a step like series of states that change at external and internal events (second from top). The elapsed time trajectory is a saw tooth pattern depicting the flow of time in an elapsed time clock that is reset to 0 at every event. Finally, at the bottom, the output trajectory depicts the output events that are produced by the output function just before applying the internal transition function at internal events.

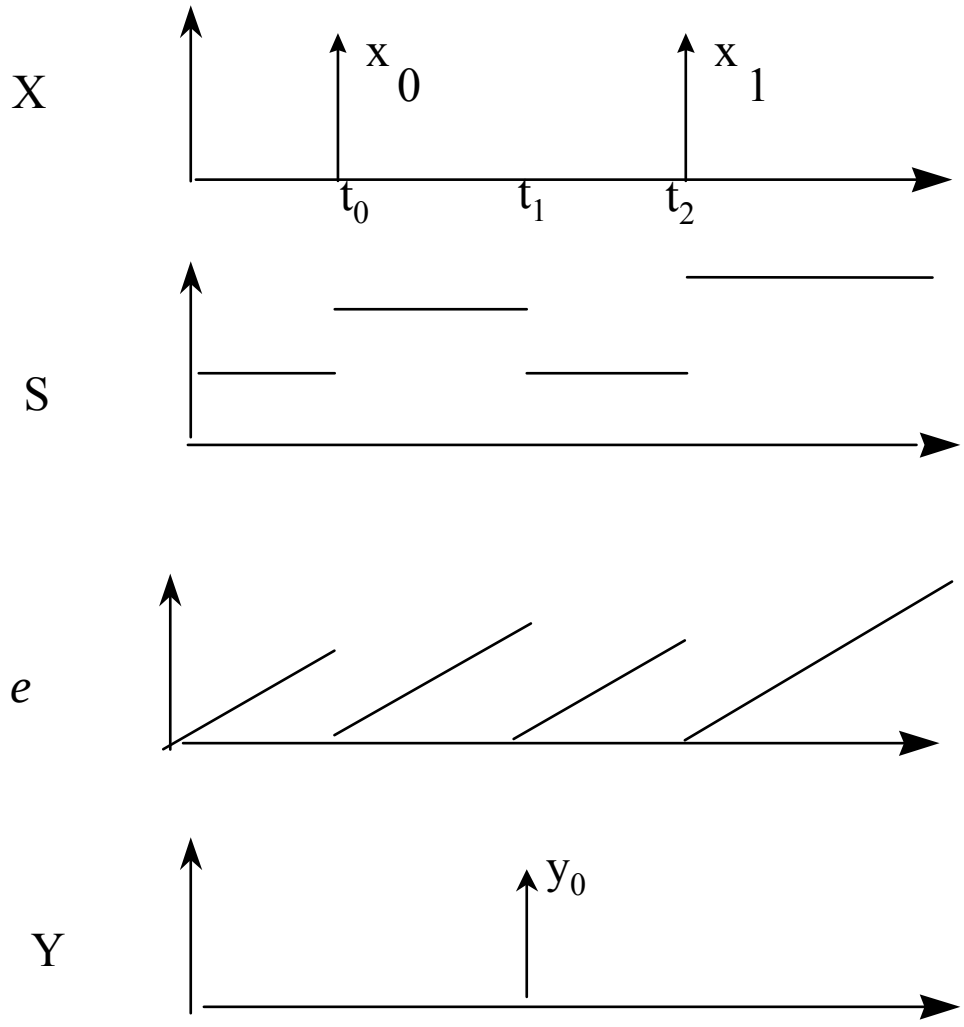

Figure 19 Behavior of DEVS

Figure 20 shows the hierarchical modular composition of DEVS.

Basic models may be coupled in the DEVS formalism to form a coupled model. A coupled model tells how to couple (connect) several component models together to form a new model.

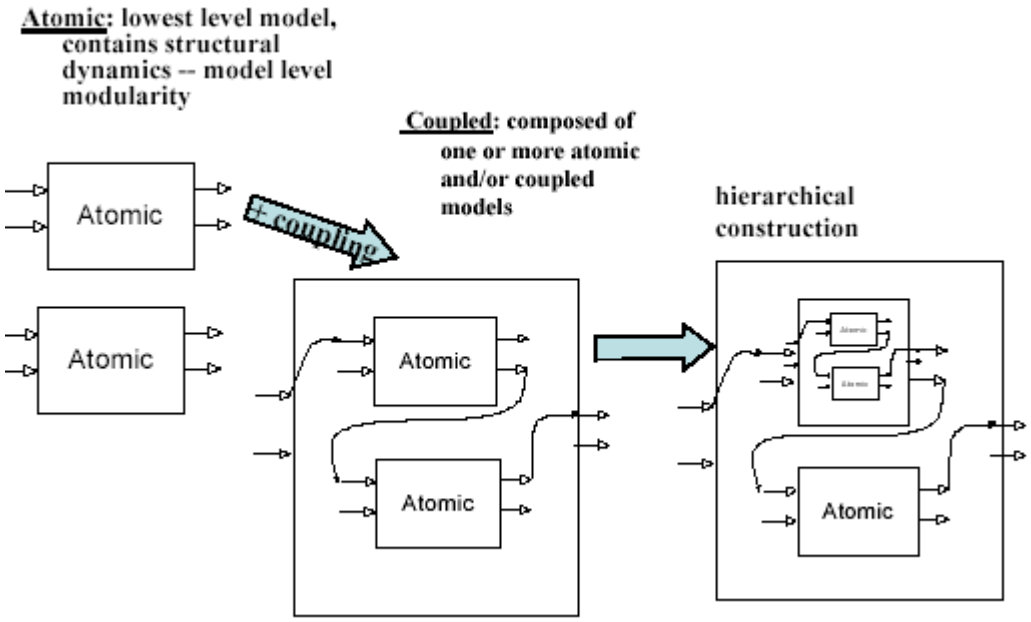

Figure 20 Hierarchical modular composition of DEVS

This latter model can itself be employed as a component in a larger coupled model, thus giving rise to hierarchical construction. A coupled model contains the following information:

- the set of components
- the set of input ports through which external events are received
- the set of output ports through which external events are sent
- the coupling specification consisting of:

➢ the external input coupling which connects the input ports of the coupled to model to one or more of the input ports of the components – this directs inputs received by the coupled model to designated component models,

➢ the external output coupling which connects output ports of components to output ports of the coupled model – thus when an output is generated by a component it may

be sent to a designated output port of the coupled model and thus be transmitted externally,

- the internal coupling which connects output ports of components to input ports of other components- when an input is generated by a component it may be sent to the input ports of designated components (in addition to being sent to an output port of the coupled model).

4. DEVS INTEGRATOR

This chapter introduces the theory of DEVS integrator [3] and its application to solve system of partial differential equations. It also explains the condition under which a system modeled using DEVS will reach equilibrium.

4.1. Quantum Based Integration Scheme

The following figure shows the DEVS integrator for a single cell used in DEVS implementation of the diffusion problem.

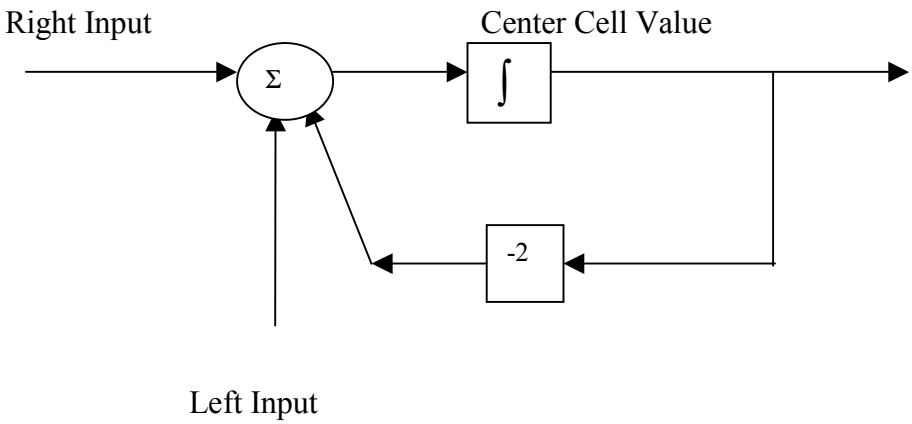

Figure 21 A system using DEVS Integrator

It takes two inputs, one is the left input, and other one is the right input.

Consider a stationary system described by the ordinary differential equation

$$\frac{dy}{dt} = f(y) \quad Eqn \quad 4.1$$

A simple numerical scheme for simulating this system is the explicit Euler formula

$$y_{n+1} = y_n + hf(y_n) \quad Eqn \quad 4.2$$

where h is the integration time step. The discrete time system described by equation 4.2 will produce approximate solutions to equation 4.1; assuming some particular assumptions concerning f(y) and h are met [10]. For example, the stable stationary system,

$$\frac{dy}{dt} = \sin(y) \quad Eqn \quad 4.3$$

whose actual solution and approximate solution, computed with 4.2 using a time step h = 0.1, are shown in figure 22

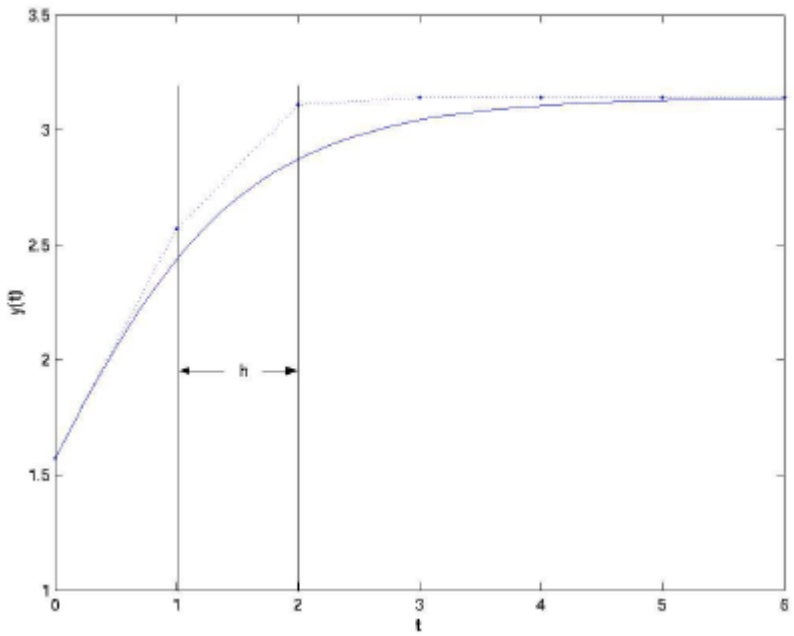

Figure 22 Soln to $\dot{y} = \sin(y)$ (solid) and its approx. soln by explicit Euler (dotted)

Suppose that, instead of generating approximate solutions at discrete points in time, we approximate the solution by looking at significant changes in the system state. The

magnitude of a change in solution that we consider to be significant is called an integration quantum (or quantum). We can define the integration quantum by

$$D = |y_n - y_{n+1}| \qquad Eqn \quad 4.4$$

so D is the desired change in the solution at each step of the computation. The time required for such a change to occur can be approximated by rearranging (4.2) to solve for h. Doing so, we find that

$$h = \frac{D}{|f(y)|} \quad Eqn \quad 4.5$$

If f(y) = 0, we let h =∞. Since we have assumed that y (t) is a stationary system, f(y) = 0 indicates that equilibrium has been reached and, consequently, the solution will not change. So no information is lost by setting h =∞.

We can construct a quantum based integration scheme by substituting 4.5 into 4.2 and keeping track of the sign of the derivative to ensure that the solution moves in the proper direction. This gives us the system

$$y_{n+1} = y_n + D\,\mathrm{sgn}(f(y_n)) \quad Eqn \quad 4.6$$

which approximates successive values of the continuous system. The formula

$$t_{n+1} = t_n + \frac{D}{|f(y_n)|} \qquad Eqn \quad 4.7$$

gives the time at which the approximate states occur. The result of using 4.6 and 4.7 to generate an approximate solution of 4.3 with D=1 is shown in figure 23.

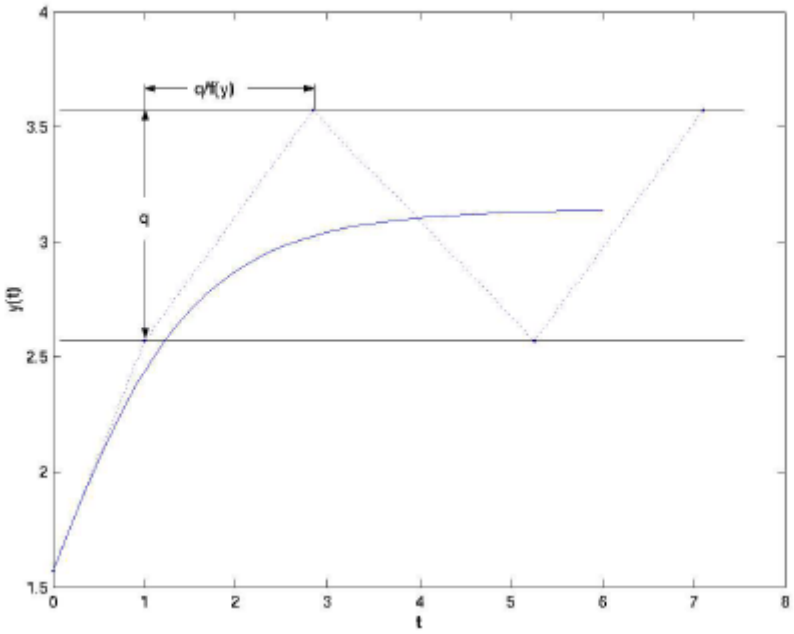

Figure 23. Soln to $\dot{y} = \sin(y)$ (solid), approx soln by quantized integ scheme (4.6), (4.7)

Figure 23 illustrates two aspects of quantized state simulation of continuous systems. First, a quantized state system that is stable will oscillate about equilibrium of the continuous system that it approximates, with the magnitude of the oscillation-approaching zero as the quantum size approaches zero. Second, for a quantized state system that is approximating a stable continuous system, it is always possible to find a quantum that is sufficiently small to ensure that the quantized state system is stable as well. In fact, as the quantum size approaches zero, the approximate solution approaches the exact solution.

4.2. Application of DEVS Integrator to solve system of Partial Differential Equations

Numerical solutions to partial differential equations with time and space components have two distinct and independent aspects. The first aspect of a numerical method is the approximation of spatial derivatives. After applying a spatial approximation technique, we are left with a system of coupled ordinary differential equations. The second aspect of the numerical method is then to select a technique for integration through time.

A set of coupled differential equations can be described as a multi-component system, with each component describing one differential equation or algebraic coupling equation. The spatial discretization of a partial differential equation will generate such a system. If we replace integration terms by their discrete event approximations (e.g., a DEVS Adams-Bashforth integrator), then we immediately arrive at a multi-component DEVS model that approximates the partial differential equation. This can be simulated using an appropriate DEVS simulator (parallel or otherwise) to generate numerical solutions.

The use of DEVS as a solution technique for partial differential equations is attractive due to the innate ability of the discrete event integrators to adjust their level of computational effort to match the rate at which the solution is changing. This results in a very efficient ‘front’ or ‘activity’ tracking technique. The technique is widely applicable since it relies on intrinsic properties of the time integration scheme, rather than on some aspect of the spatial discretization.

An example, the heat equation in one dimension will serve to illustrate the use of DEVS integrators in the simulation of systems of partial differential equations.

4.3. Adams-Bash forth type quantized state systems

The AB1 scheme is described by

$$y_{n+1} = y_n + hf_n \qquad Eqn \quad 4.8$$

Using a quantum D, this yields

$$\tau_1(D, f_0) = \frac{D}{|f_0|} \qquad Eqn \quad 4.9$$

$$\Delta_1(q, \sigma, f_0) = q + \sigma f_0 \qquad Eqn \quad 4.10$$

Therefore the internal transition function, the external transition function, the confluent function, the output function and the time advance function for the DAB1 are as follows:

$$\delta_{\text{int}}(q, q1, f_0, \sigma) = (\Delta_1(q, \sigma, f_0), \Delta_1(q, \sigma, f_0), f_0, \tau_1(D, f_0)),$$
$$\delta_{ext}((q, q_1, f_0, \sigma), e, x) = (\Delta_1(q, e, f_0), q_1, x, \tau_1(D - |\Delta_1(q, e, f_0) - q_1|, x)),$$
$$\delta_{con}((q, q_1, f_0, \sigma), x) = (\Delta_1(q, \sigma, f_0), \Delta_1(q, \sigma, f_0), x, \tau_1(D, x)),$$
$$\lambda(q, q_1, f_0, \sigma) = \Delta_1(q, \sigma, f_0), and$$
$$ta(q, q_1, f_0, \sigma) = \sigma$$

4.4. Application of DAB1 to finite difference schemes

The use of finite differences to approximate spatial derivatives in a partial differential equation gives rise to an input free continuous linear system that can be expressed as

$$\frac{dx(t)}{dt} = Ax(t) \qquad Eqn \quad 4.11$$

where the state vector has one element for each spatial grid point. For example, the heat equation in one dimension with coefficient of diffusion c is given by

$$\frac{\partial u}{\partial t} = c \frac{\partial^2 u}{\partial x^2} \qquad Eqn \quad 4.12$$

where u(x,t) is the temperature at time t and position x. The spatial derivative can be approximated using center differences, this giving

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{(\Delta x)^2} \qquad Eqn \quad 4.13$$

where Δx is the spatial resolution of the approximation. Substituting 4.13 in 4.12 gives a system of coupled ordinary differential equations

$$\frac{\partial u(x_i,t)}{\partial t} \approx c\frac{u(x_{i+1},t) - 2u(x_i,t) + u(x_{i-1},t)}{(\Delta x)^2} \qquad Eqn \quad 4.14$$

where the x_i are the positions $i(\Delta x)$ at which the approximate solution is computed.

For the heat equation, it is necessary to specify boundary conditions so that the problem is well posed. When the boundary conditions are fixed (i.e., the edge cells are assumed to have a constant temperature) then 4.14 is a stable linear system and the unconditional linear stability of DAB1 ensures that the discrete event approximation of 4.14 is stable as well [9].

Another choice of boundary conditions that result in a marginally stable linear system is

$$u(0,t) = u(\Delta x,t) \quad and \quad u(N\Delta x,t) = u((N-1)\Delta x,t), \qquad Eqn \quad 4.15$$

where the cells 1,2,…..,N-1 are in the interior of the computational region, and the cells 0 and N are at the boundaries.

4.5. Explanation of Activity/q = DEVS transitions using DEVS Integrator

Activity at time τ is given by

$$A_i(\tau) = \int_0^T \left| \frac{dy_i}{dt} \right| dt$$

Therefore the total activity over a time T is given by

$$A(T) = \sum_i A_i(\tau) = \int_0^T \sum_i \left| \frac{dy_i}{dt} \right| * dt$$

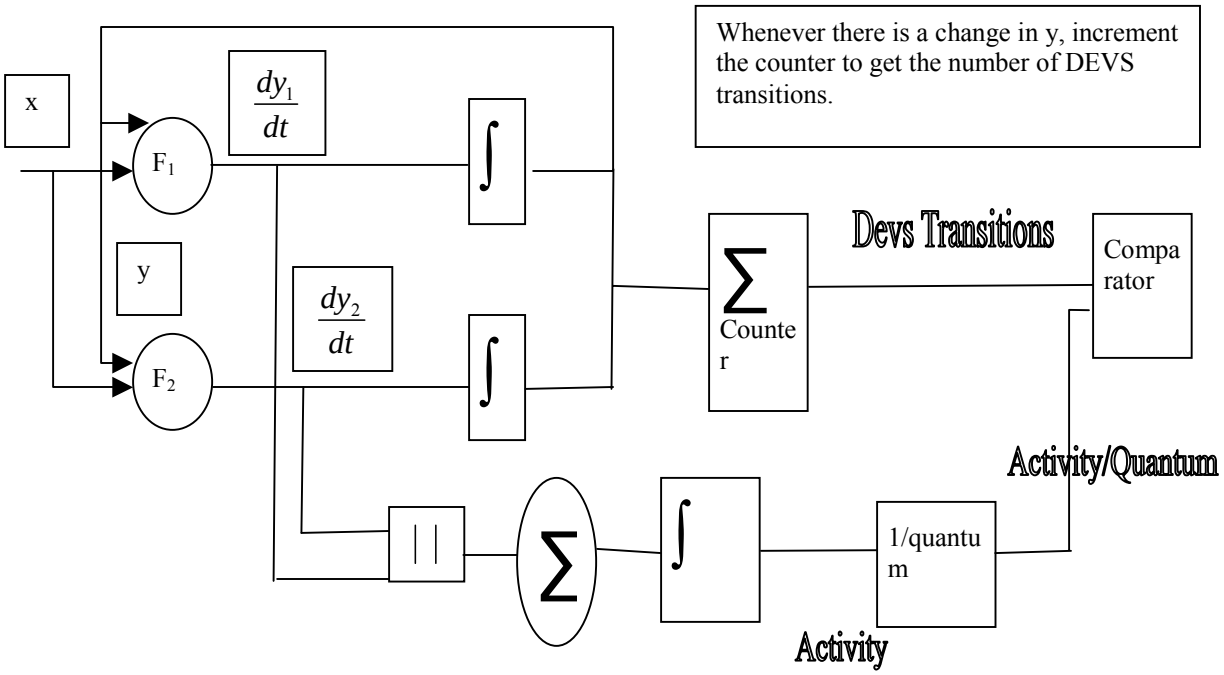

Figure 24 Explanation of Activity/q = DEVS transitions using DEVS Integrator

The above figure shows how two DEVS integrators can be combined to explain proposition 2, which was proved in chapter 2. The quantum chosen should be such that Activity/quantum is equal to the Number of DEVS transitions.

5. COMPARISON OF DEVS AND DISCRETE TIME PROCESSING

This chapter develops the comparison of DEVS and Discrete Time Processing analytically. A generalized formula for comparison between DEVS and DTSS is derived. This general formula is then applied to some particular cases of diffusion and the results are tabulated.

5.1. Generalized formula for comparison of DEVS and Discrete Time processing

We base a comparison between DEVS and DTSS on the smallest time advance required in a DEVS simulation. This approach has been found to result in the same accuracy for both simulation methods [1].

The smallest time advance for one DEVS integrator, illustrated in the figure, is given by dividing the quantum by the magnitude of the largest time derivative. We set the time step of a discrete time integrator to this value, i.e.,

$$\Delta t = \frac{q}{Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_i} \qquad \text{Eqn 5.1}$$

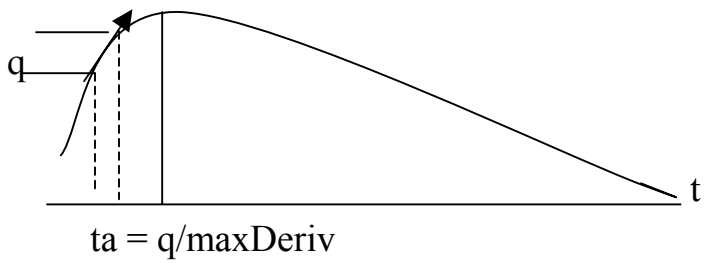

Figure 25 Comparison of DEVS and Discrete Time Processing

For a set of integrator components of a DTSS simulation of a differential equation, let the maximum absolute derivative occur at an integrator (cell) with index $\bar{i}$, i.e.

$$Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_{\bar{i}} \quad \geq \quad \left(\left|\frac{\partial Y}{\partial t}\right|\right)_{i} \quad \forall i$$

Then since all DTSS integrators have the same time step, they must employ the smallest allowed for the one with index $\bar{i}$, i.e.

$$\hat{\Delta}t = \mathrm{q}/\,Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_{\bar{i}} \qquad \text{Eqn 5.2}$$

The number of transitions (threshold crossings) done by N DEVS integrators is:

$$\#\,DEVS \quad = \quad \frac{A}{q} \quad = \quad \frac{\sum_{i}^{N} A(i)}{q} \qquad \text{Eqn 5.3a}$$

The number of transitions done by the same number of DTSS integrators employing a time step $\hat{\Delta}t$ is:

$$\#\,DTSS \quad = \quad \frac{T}{\hat{\Delta}t} \quad * \quad N \qquad \text{Eqn 5.3b}$$

The ratio of DTSS to DEVS transitions is:

$$\frac{\#\,DTSS}{\#\,DEVS} \quad = \quad \frac{q * T * N}{\hat{\Delta}t * A}$$

Applying the relation in Eqn 5.2

$$\frac{\#DTSS}{\#DEVS} = Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_{\bar{i}} \quad \frac{T * N}{A} \qquad \text{Eqn 5.4}$$

or in terms of the average activity per cell,

This gives the proof of

Proposition 4:

$$\frac{\#DTSS}{\#DEVS} = Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_{\bar{i}} \quad \frac{T}{A / N} \qquad \text{Eqn 5.5}$$

This is the fundamental relation that we will use to compare DEVS and DTSS for diffusion from various initial states in the next section.

We now show that the ratio $\frac{\#DTSS}{\#DEVS}$ is always greater than or equal to 1.

Proposition 5:

$\frac{\#DTSS}{\#DEVS} \geq 1$.

Proof:

Recall that the activity of a DEVS integrator, i over a time interval T is given by:

$$A_i = (\int_0^T \left|\frac{\partial Y}{\partial t}\right| \partial t)_i$$

and assuming the derivative is bounded from above over the interval, we have

$$A_i \quad \leq \quad T \quad * \quad Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_i$$

So from Eqn 5.3a

$$\#\,DEVS \quad \leq \quad T \qquad \frac{\sum_i^N Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_i}{q}$$

Using Eqn 5.3b

$$\frac{\#\,DTSS}{\#\,DEVS} \quad \geq \quad \frac{q}{\hat{\Delta} t} \qquad \frac{N}{\sum_i^N Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_i}$$

From Eqn 5.2, we have

$$\frac{\#\,DTSS}{\#\,DEVS} \quad \geq \quad Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_{\bar{i}} \qquad \frac{N}{\sum_i^N Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_i}$$

But

$$N \quad Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_{\bar{i}} \quad \geq \qquad \sum_i^N Max\left(\left|\frac{\partial Y}{\partial t}\right|\right)_i$$

So

$$\frac{\#\,DTSS}{\#\,DEVS} \quad \geq \quad 1 \qquad \blacksquare$$

5.2. Comparison for a rectangular pulse as the initial data

The time derivate is maximum at time t = 0 at the cell where the spatial discontinuity occurs, in this case the cell is (N-1)/2.

Therefore the maximum time derivative is given by

$$\mathrm{Re}\,ct_{MaxDeriv} = c\left(\frac{2x_i - x_{i-1} - x_{i+1}}{(\Delta x)^2}\right) = c\left(\frac{2H - H - 0}{(\Delta x)^2}\right) = cH\left(\frac{N}{L}\right)^2 \qquad Eqn \quad 5.6$$

where c is the diffusion constant.

From equations 2.5, 5.5 and 5.6,

We have

$$\frac{DTSS}{DEVS} = \frac{TcH\left(\frac{N}{L}\right)^2}{2Hw(1-w)} = \frac{Tc\left(\frac{N}{L}\right)^2}{2w(1-w)} \propto O(N^2) \qquad Eqn \quad 5.7$$

where w = W/L.

From equation 5.7, we can conclude that, for the rectangular pulse the performance of DEVS:

1. Increases as the square of the number of cells.
2. Increases as the time of simulation, T is increased.
3. Increases as the diffusion constant, c is increased.
4. Increases as the ratio of width to length goes to 0 or to 1.
5. Decreases as the length of the pulse is increased while keeping the width-to-length ratio constant. .
6. Is independent of the height of the pulse.

5.3. Comparison for a triangular pulse as the initial data

The time derivate is maximum at time t =0 at the cell where the spatial discontinuity occurs; in this case that cell is (N-1).

Therefore the maximum time derivative is given by

$$Triang_{MaxDeriv} = c\left(\frac{2x_i - x_{i-1} - x_{i+1}}{(\Delta x)^2}\right) = c\left(\frac{2H - H\left(\frac{N-2}{N-1}\right) - H}{(\Delta x)^2}\right) = c\left(\frac{H}{N-1} * \frac{N^2}{L^2}\right) \qquad Eqn \quad 5.8$$

where c is the diffusion constant.

From equations 2.6, 5.5 and 5.8, we have

$$\frac{DTSS}{DEVS} = \frac{Tc\left(\frac{H}{N-1}\right)\left(\frac{N^2}{L^2}\right)}{\frac{H(N-1)}{4N}} = \frac{4TcN^3}{L^2(N-1)^2} \approx \frac{4Tc}{L^2} N \propto O(N) \qquad \text{Eqn 5.9}$$

From equation 5.9, we can conclude that, for the triangular pulse the performance of DEVS:

1. Increases as the number of cells.
2. Increases as the time of simulation, T is increased.
3. Increases as the diffusion constant, c is increased.
4. Decreases as the square of the length of the space.
5. Is independent of the height of the pulse.

5.4. Comparison for a Gaussian pulse as the initial data

The maximum time derivative for a cell location x, depends on whether the cell does not see a crest as in figure 14 or does as in figure 15. In the first case, the maximum derivative that it experiences is that at time tstart.

This works out to

$$Gaussian^{1}{}_{MaxDeriv} = \frac{0.062H}{\sqrt{c*tstart^{3}}} \qquad \text{Eqn 5.10}$$

In the second case, the maximum derivative is obtained by differentiating equation 2.7 with respect to t, equating it to zero to solve for t, and substituting this value of t back in equation 2.7.

Doing so, we get

$$Gaussian^{2}{}_{MaxDeriv}(x) = \frac{1.83cH}{x^{3}} - \frac{0.206H}{c^{2}x^{3}} = \frac{H}{x^{3}}\left(\frac{1.83c^{3} - 0.206}{c^{2}}\right) \quad Eqn \quad 5.11$$

For the derivative to be largest over all cells seeing a crest, the denominator of equation 5.11 must be the smallest allowable value. This occurs for the smallest value of x, viz., $\sqrt{2*c*tstart} + \Delta x$, which is the first cell to experience a crest.

Neglecting Δx in comparison with $\sqrt{2*c*tstart}$, we obtain:

$$Gaussian^{2}{}_{MaxDeriv} = H\left(\frac{1.83c^{3} - 0.206}{c^{2}}\right)\left(\frac{1}{\sqrt{2*c*tstart}}\right)^{3} = \left(\frac{H}{tstart^{3/2}}\right)\lambda(c) \quad Eqn \quad 5.12$$

Comparing the two maximum derivatives we find that equation 5.12 gives the maximum derivative if c, the diffusion constant is greater than 0.5 else the maximum derivative is given by equation 5.10.

In both cases, the dependence on tstart is of the same form. Since we are interested in decreasing values of c we will use the case 1 version in the following:

From equations 2.11, 5.5 and 5.10,

We have,

$$\frac{DTSS}{DEVS} = \frac{\frac{0.062TH}{\sqrt{c*tstart^3}}}{\frac{H}{L}\left(\frac{2}{\sqrt{2\pi e}}\ln\left(\frac{L}{\sqrt{2*c*tstart}}\right) - \frac{1}{2}erf\left(\frac{L}{\sqrt{4*c*tstart}}\right) + erf(0.707)\right)} \qquad Eqn \quad 5.13$$

$$\therefore \quad \frac{DTSS}{DEVS} = \frac{0.062T}{f(L)\sqrt{c*tstart^3}} \propto O(N^0) \qquad Eqn \quad 5.14$$

where f(L) is a decreasing function of L i.e. it decreases as L increases. This is because the denominator in equation 5.13 is of the form f(log(L))/L which goes to zero as L tends to infinity.

From equation 5.14, we can conclude that, for the Gaussian pulse, the performance of DEVS

1. Does not depend on the number of cells.

2. Increases as the time of simulation, T is increased.

3. Is independent of the height of the pulse.

4. Increases with the length of the space.

5. Increases as the diffusion constant, c decreases.

Points 4 and 5 agree with the behavior found in the case of the rectangular pulse. In both cases, as the initial amount of heat is confined to a smaller and smaller region, the

advantage of the DEVS simulation increases. This is because the DEVS simulation, exploits the reduced activity that results from such concentration, while the discrete time method does not.

To summarize, the following table shows the ratio of DTSS to DEVS for the three conditions

TABLE 2. DTSS/DEVS for Various Initial Conditions

Type of initial data	DTSS/DEVS
Rectangular Pulse	$\frac{TcN^2}{2w(1-w)L^2}$ where w is the width to the length ratio
Triangular Pulse	$\frac{4TcN}{L^2}$
Gaussian Pulse	$\frac{0.062T}{f(L)(c*tstart^3)^{1/2}}$

6. VERIFICATION

This chapter validates the analytical derivations of the previous chapters through experimental results.

Measurement of Activity

The concept of activity was discussed in chapter 2. We now discuss the procedure by which activity and threshold crossings were measured for the following experimental results. For measuring the activity, the first cells' value was stored in a variable and the sum of the differences between all the consecutive cells was computed. This gives the total activity. In order to compute the threshold crossings, a counter was initialized to zero and the first cells' value was stored in a variable. The counter was incremented if the difference between any two consecutive cells exceeded the quantum.

6.1 Verification of Activity Formula when the initial data is a Triangular Pulse

In section 2.3.2, we obtained the formula

$$A = H (N-1)/4 \quad \text{Eqn 6.1}$$

for the total activity of a triangular pulse of initial height H, using N cells. We now discuss experimental results that confirm the correctness of this formula.

To measure activity of the one-dimensional triangular pulse, we employ a discrete time simulation with the following parameters values:

Diffusion constant, c = 1.0

Length of the space, L = 16.0

Initial height of the pulse, H = 10.0

Spatial Step, $\Delta x = L/N$, where N is the number of cells

Time step, $\Delta t = 0.5*(\Delta x^2)$

Tend (Total time of Simulation) = 100 seconds

Figure 26 shows the activities for different initial heights for a cell space of 100 cells.

Recall that the activity at each cell measures the number of quantum thresholds, or distance measured in quanta, that it travels from its initial state to the common equilibrium state. This distance is proportional to the distance of the cell from the center cell. In other words, the middle-most cell has zero activity whereas the cells at the edges of the pulse have the highest activity. This is verified in figure 26.

From equation 6.1, we see that the activity is directly proportional to the initial height of the pulse. This is verified by figure 26, where the total activity of each cell increases with initial height.

Table 3 compares the total activity measured in the simulation versus that predicted by Eqn 6.1. We see that the agreement is excellent.

TABLE 3. Verification of Activity for the Triangular Pulse

#Cells	Total Activity measured in simulation	Total Activity computed from Eqn 6.1
10	19.27	22.5
100	244.26	247.5
200	494.2	497.5
500	1244.0	1247.5
700	1744.0	1747.5
1000	2494.0	2497.5

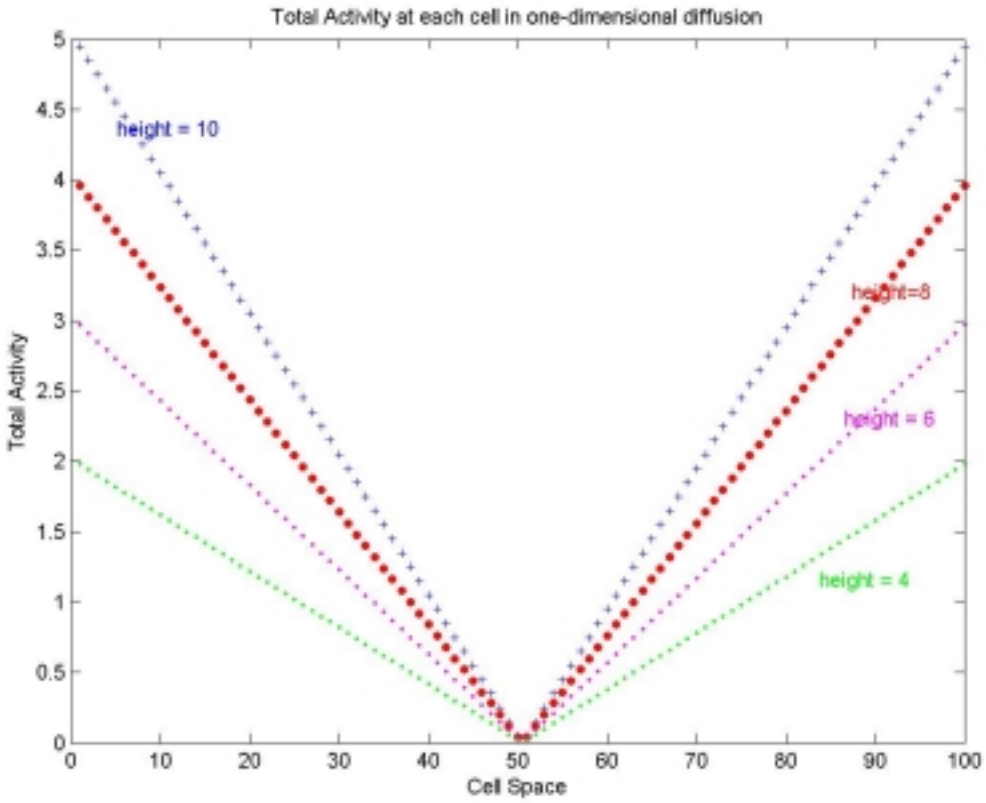

Figure 26 Total activity at each cell in 1-D diffusion starting from a triangular pulse of various heights for a cell space of 100 cells

6.2 Verification of Activity Formula when the initial data is a Gaussian Pulse

In section 2.3.3, we obtained the formula

$$A = \frac{NH}{L}\left(\frac{2}{\sqrt{2\pi e}}\ln\left(\frac{L}{\sqrt{2*c*tstart}}\right) - \frac{1}{2}erf\left(\frac{L}{\sqrt{4*c*tstart}}\right) + erf(0.707)\right) \qquad Eqn \quad 6.2$$

for the total activity of a Gaussian pulse of initial height H, using N cells. We now discuss experimental results that confirm the correctness of this formula. It should be noted that the experimental results below were done for two-dimensional diffusion for the Gaussian pulse. We are comparing them with equation 6.2, which is the equation of activity for one-dimensional diffusion of the Gaussian pulse since the derivation for two-dimensions should be similar to that for one dimension.

To measure activity of the two-dimensional Gaussian pulse, we employ a discrete time simulation with the following parameters values:

Diffusion constant, c = 0.1

Length of the space, L = 16.0

Initial height of the pulse, H = 10.0

Spatial Step, Δ = L/ROWS, where ROWS denotes the number of cells in either the x-axis or y-axis directions (assuming a square cell-size).

Time step, tstep = $2.2*(\Delta^2)$

Tend (Total time of Simulation) = 96 seconds.

Table 4 shows the variation of total activity per cell (total activity divided by the number of the cells) for two-dimensional diffusion when the cell space size (number of cells) is varied

TABLE 4. Total Activity per Cell for 2-D diffusion

Cell Space Size (Number of cells)	Total Activity for Time (t = 96 seconds)	Total Activity Per Cell (Total Activity/#Cells)
625	128.3	0.21
2500	1082.6	0.43
5625	1863	0.33
10000	3822.5	0.38
40000	15280	0.38

Figure 27 shows the plot of table 4.

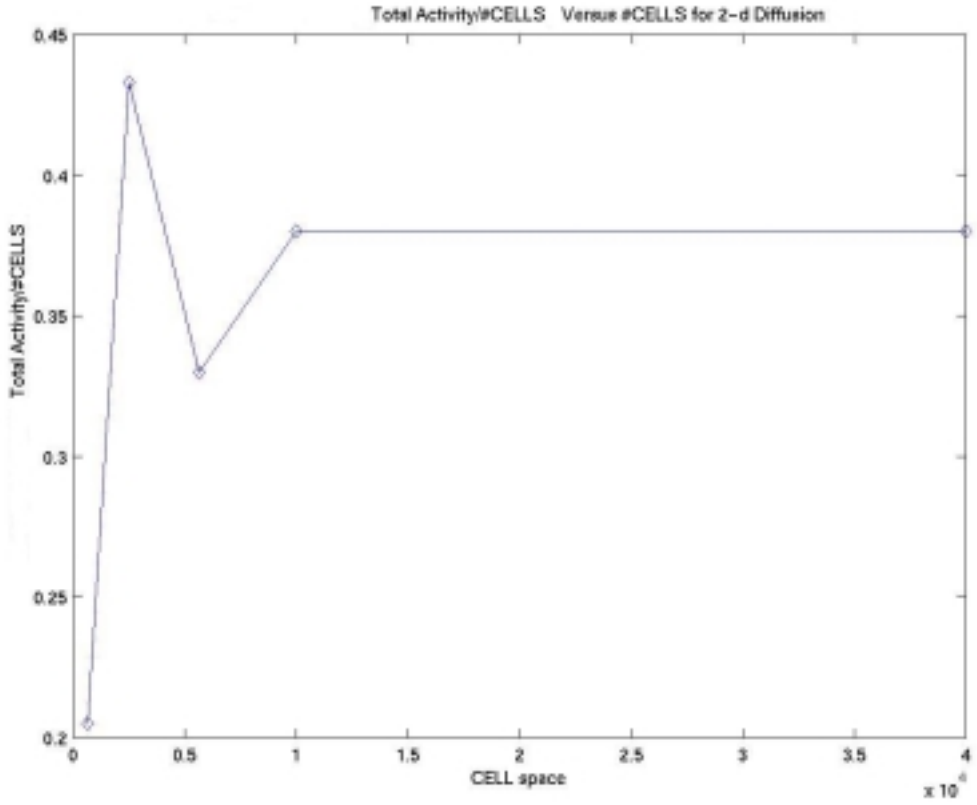

Figure 27 Total Activity vs. Cell Space Size for two-dimensional diffusion of Gaussian Pulse

From figure 27, we see that the activity per cell goes to a constant. This is as it should be as is evident from equation 6.2.

Table 5 shows the Normalized Activity (Total Activity divided by mass squared, where mass is the total value of the heat, which is conserved in this case) versus the total time for which the program was run

TABLE 5. Normalized Activity Vs. Tend for 2-D Diffusion

Initial Pulse Height	Time (Tend in sec's)	Normalized Activity
10	12	0.04
	24	0.11
	48	0.28
	96	0.35
	120	0.35
	140	0.35
5	12	0.07
	24	0.20
	48	0.38
	96	0.39
	120	0.39
	140	0.39
1	12	0.22
	24	0.37
	48	0.38
	96	0.38
	120	0.38
	140	0.38

Figure 28 shows the plot of table 5.

In figure 28, height denotes the initial height of the pulse before the diffusion starts. From figure 28, it can be concluded that, smaller the initial height of the pulse, sooner the activity reaches a constant and therefore this verifies that normalized activity converges in both time and spatial resolution.

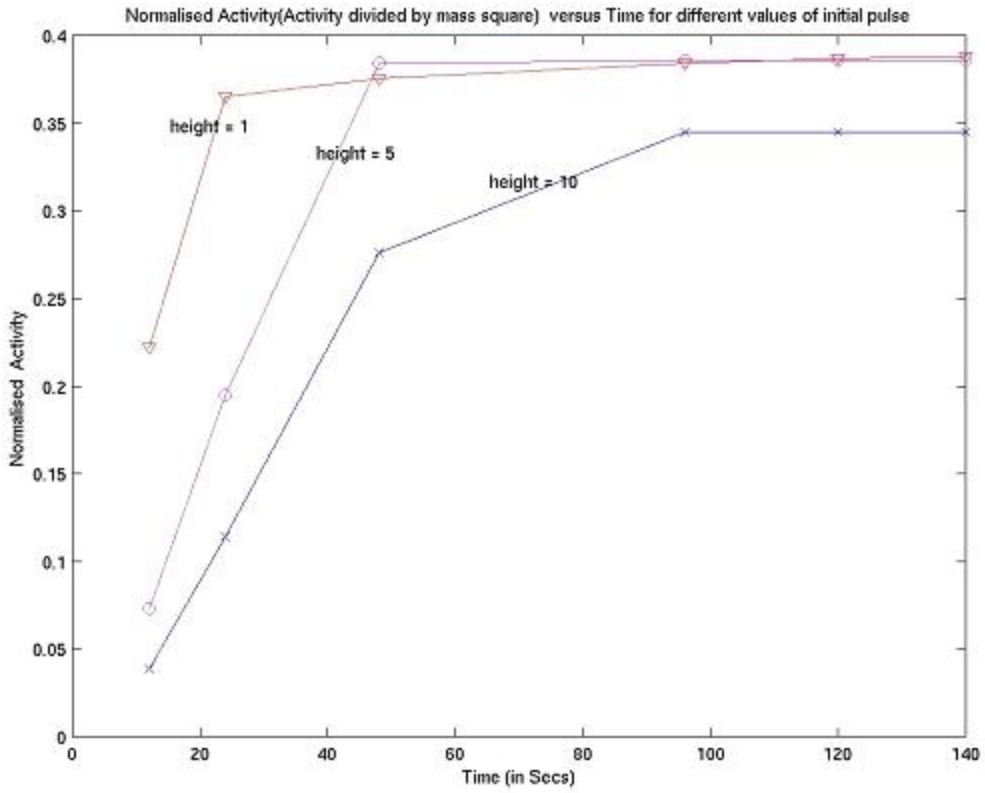

Figure 28 Normalized activity Vs tend for two-dimensional diffusion

In figure 29, the predicted activity A is plotted against the number of crossings determined from direct count of threshold crossing. Recall that from proposition 2, A/q should equal the number of crossings as q approaches 0. The plot shows increasing better agreement between A/q and number of crossings, as q gets larger. This anomaly can be explained by the fact that, in the implementation of the discrete time simulation; we did not change the time step as the quantum varied. Therefore, for a large quantum, using the same time step we may have under-estimated the number of threshold crossings. This verifies that activity can be used to predict the number of threshold crossings.

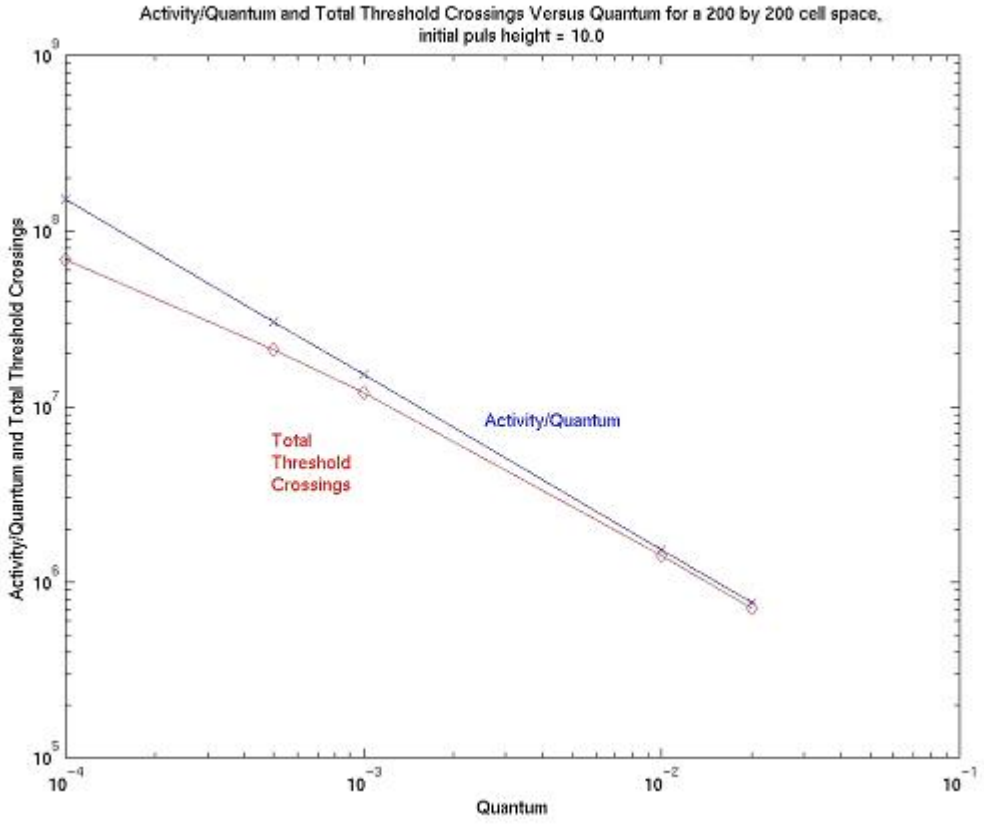

Figure 29 Activity/quantum and total threshold crossings Vs quantum for a 200 by 200 cell-space

6.3 Verification that the total activity of the Gaussian pulse decreases as the Length increases

From equation 6.2 we see that the total activity, A decreases as the length of the space, L increases. This is so since the error function goes to a constant and the natural logarithm term grows more slowly than *L*. This is confirmed as shown in figure 30. This is because the length of the space does not affect the size of the region in which the initial heat is confined. As the length increases, so does the number of cells that have very small activities. These are cells far from the initial pulse to which very little heat diffuses.

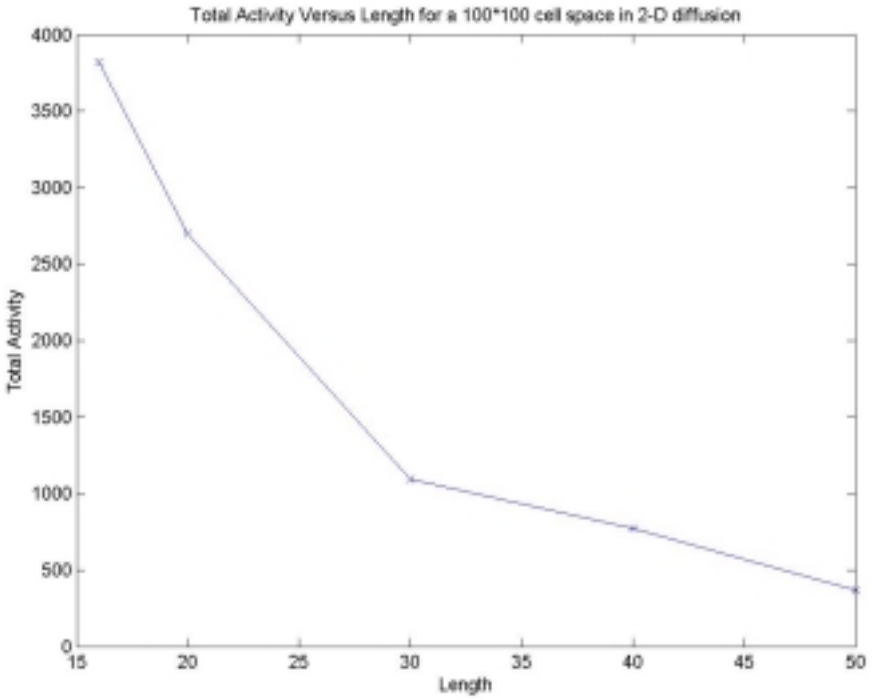

Figure 30 Total Activity Vs. Cell space in two-dimensional Diffusion

6.4 Verification of ratio formula when the initial data is a Gaussian Pulse

In section 5.4, we obtained the formula

$$\therefore \quad \frac{DTSS}{DEVS} = \frac{0.062T}{f(L)\sqrt{c*tstart^3}} \propto O(N^0) \qquad Eqn \quad 6.3$$

for the ratio of discrete time to DEVS transitions for a Gaussian pulse of initial height H, using N cells. We now discuss experimental results that confirm the correctness of this formula.

The total number of iterations the discrete time version took for a 50 by 50 cell space was 1038707 and for a 100 by 100 cell space the total number of iterations was 1.66*e 7 (total number of iterations = number of cells *Tend/ Δt).

Table 6 shows the ratio of number of iterations in discrete time to the total number of threshold crossings for different heights of initial pulse.

TABLE 6. Ratio of iterations in DTSS to Threshold Crossings in DEVS

Cell Space	Quantum (q)	Ratio for an Initial Height of 10 units	Ratio for an Initial Height of 5 units	Ratio for an Initial Height of 1 units
50 by 50	0.0001	1.2	1.3	2.6
	0.0005	1.8	2.6	8.1
	0.001	2.6	4.1	14.0
	0.01	14.0	25.5	125.0
	0.02	25.5	49.0	289.0
	0.1	125.0	289.0	3256.1
100 by 100	0.0001	1.7	2.5	7.5
	0.0005	4.5	8.0	27.7
	0.001	8.0	12.8	51.6
	0.01	51.5	99.2	550.0
	0.02	99.2	199.0	1350.5
	0.1	550.0	1350.5	14955.0

So, we see that for a larger quantum, we get the best performance, but, on the other hand, there will be fewer events and hence less accuracy, so we need to choose a quantum, which has an optimum value.

Figure 31 shows the ratio plot for a 50 by 50 cell space

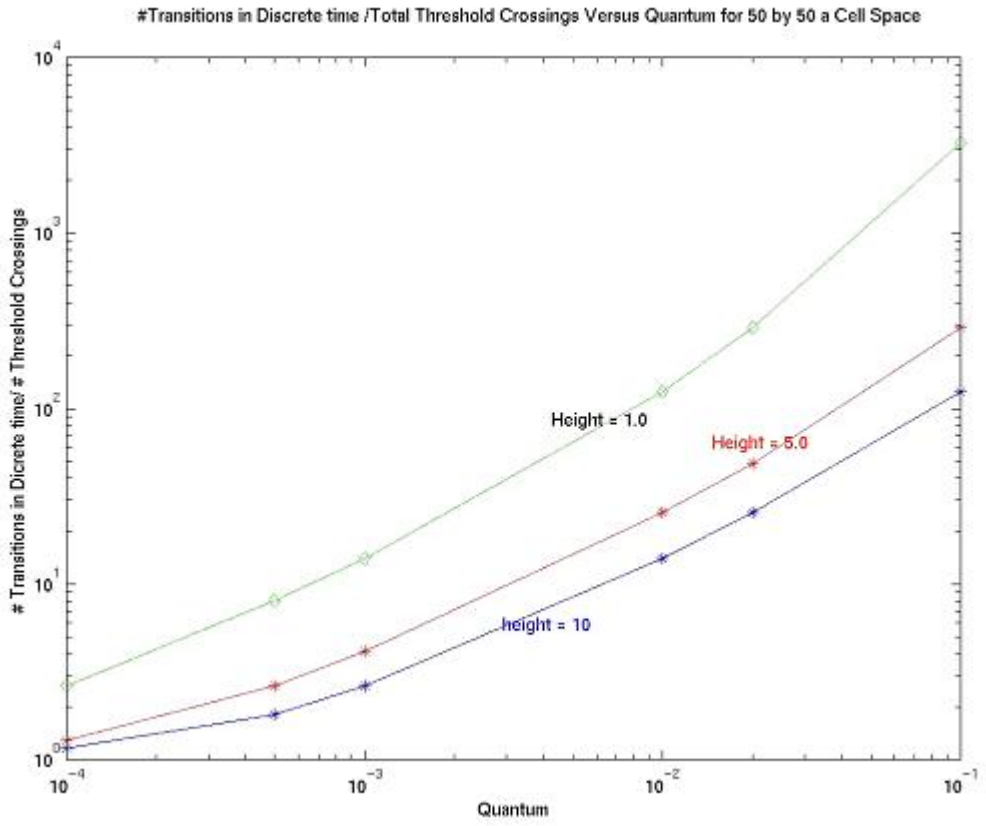

Figure 31 Transitions in DT/Transitions in DEVS versus quantum for different heights of initial pulse for a 50 by 50 cell space

Figure 32 shows the ratio plot for a 100 by 100 cell space

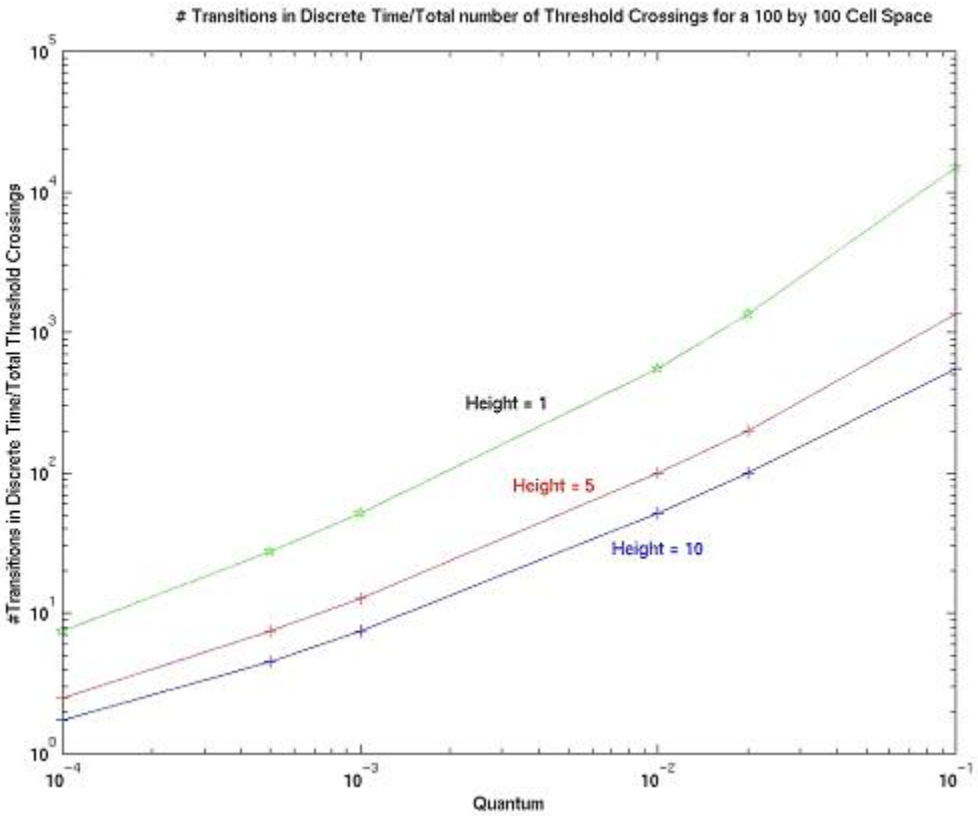

Figure 32 Number of cell transitions/number of threshold crossings versus quantum size for different heights of initial pulse height for a 100 by 100 cell space

Figures 31 and 32 verify that the advantage for DEVS potentially increases greatly with increasing spatial resolution (number of cells) and amplitude. However, it remains to show that this advantage can actually be exploited in a DEVS quantization-based simulation. We turn to this question next.

6.5 Determining the dependence of quantum on number of cells for the triangular pulse

We employed a DEVS quantization-based simulation such as discussed in Chapter 4. The following parameters were used for the simulation

Diffusion constant, c = 1.0

Length of the space, L = 16.0

Initial height of the pulse, H = 10.0

Spatial Step, $\Delta x = L/N$, where N is the number of cells

Time step, $\Delta t = 0.5*(\Delta x^2)$

Tend (Total time of Simulation) = 100 seconds

The Courant condition provides a guide for choice of time step to get stability as the number of cells increases. For the DEVS quantization-based method, we approached the problem of determining the dependence of quantum on number of cells as follows.

For one-dimensional diffusion, we have the equation

$$u_t = cu_{xx} \qquad \text{Eqn 6.4}$$

where *c* is the diffusion constant and *u* is the state variable.

Using the method of lines, which was mentioned in chapter 1, equation 6.4 can be written as

$$\frac{\partial u_i}{\partial t} = c\left(\frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2}\right) \qquad \text{Eqn 6.5}$$

If q is the quantum used [3], then

$$u_{i+1} = u_{i+1}^0 + n_{i+1}q$$
$$u_i = u_i^0 + n_i q$$
$$u_{i-1} = u_{i-1}^0 + n_{i-1}q$$

where u_i^0 denotes the initial state of the ith cell and n_i , n_{i-1}, and n_{i+1} are integers.

Therefore equation 6.5 can be written as

$$\frac{\partial u_i}{\partial t} = c\left(\frac{u_{i+1}^0 + n_{i+1}q - 2(u_i^0 + n_i q) + u_{i-1}^0 + n_{i-1}q}{(\Delta x)^2} \right) \quad \text{Eqn 6.6}$$

Now we will find the value of q, which solves the equation $\frac{\partial u_i}{\partial t} = 0$, so that the system is stable, i.e. the system will reach equilibrium.

This value of q is given by

$$q(n_{i-1} - 2n_i + n_{i+1}) \le 2u_i^0 - u_{i+1}^0 - u_{i-1}^0 \quad \text{Eqn 6.7}$$

Consider the case in which the triangular pulse of initial height H, number of cells N and space of length L, is the initial data, then equation 6.7 becomes

$$q(n_{i-1} - 2n_i + n_{i+1}) \le 2H - H - \left(H - \frac{H}{L}\Delta x \right) = \frac{H}{L}\Delta x = \frac{H}{N} \quad \text{Eqn 6.8}$$

where $\Delta x = \frac{L}{N}$ is the spatial resolution.

If we define

$$n_{i-1} - 2n_i + n_{i+1} = \alpha, \text{ equation 6.8 becomes}$$

$$q(N) \le \frac{H}{N\alpha} \quad \text{Eqn 6.9}$$

where α is an integer greater than or equal to 1.

This compares favorably with the dependence of the time step determined by the Courant condition, viz.

$$\Delta t(N) = (0.5/c)^* \Delta x^2$$

$$= (0.5/c)^*(L/N)^2$$

i.e., $$q(N)/\Delta t(N) \propto O(N) \quad \text{Eqn 6.10}$$

showing that, the quantum needed for accuracy in the DEVS case decreases by order N more slowly than the time step required by the Courant condition.

From chapter 2, the number of DEVS transitions is given by A/q where the activity A=NH/4.

Therefore, the number of DEVS transitions,

$$\text{DEVS} = \frac{A}{q} = \frac{NH/4}{H/N\alpha} = \frac{\alpha N^2}{4} \quad \text{Eqn 6.11}$$

If we denote the total time of simulation as T, the total number of cells as N and the time step as Δt, then the total number of discrete time transitions is equal to

$$\text{DTSS} = \frac{T}{\Delta t} N \quad \text{Eqn 6.12}$$

For stability, from the COURANT-FRIEDRICH-LEWY condition, we have

$$\Delta t = \frac{0.5}{c}(\Delta x)^2 = \frac{0.5}{c}\left(\frac{L}{N}\right)^2$$

Therefore, equation 6.12 becomes

$$\text{DTSS} = \frac{2TcN^3}{L^2} \quad \text{Eqn 6.13}$$

Therefore, from equations 6.12 and 6.13, we have the ratio of DTSS to DEVS as

$$\frac{DTSS}{DEVS} = \frac{8TNc}{\alpha L^2} \quad \text{Eqn 6.14}$$

In section 5.3, we found that the ratio of number of transitions of DTSS to DEVS in the case of the triangular pulse was equal to

$$\frac{DTSS}{DEVS} = \frac{4Tc}{L^2} N \quad \text{Eqn 6.15}$$

Both equations 6.14 and 6.15 predict that DEVS should have advantage over discrete time as the cell space, N increases.

Equation 6.14 is based on stability whereas equation 6.15 is based on accuracy. Therefore, from equations 6.14 and 6.15, we have

$$\alpha \geq 2 \quad \text{Eqn 6.16}$$

Equation 6.16 gives a lower bound on α for stability.

To measure the accuracy of a simulation run we compare the number of transitions actually employed to the number predicted by the activity computed from Eqn. 2.6. This means we are taking as a measure of accuracy of the simulation how close the activity of the simulation run matches the predicted activity. For a given number of cells, we adjusted the quantum employed in the simulation to find the largest value for which the ratio is close to one. We call this the optimum quantum to get matching activity.

The results are displayed in Table 7.

TABLE 7. Optimum Quantum to get matching Activity

#Cells	Optimum quantum to get matching activity	#DEVS transitions	#DEVS transitions * quantum	Computed Total Activity (2.5*(N-1)) where N is the total number of cells and H =10.0
10	0.01	2405	24.05	22.5
100	10^{-4}	$2.6*10^{6}$	$2.6*10^{2}$	247.5
200	10^{-4}	$6.53*10^{6}$	$6.53*10^{2}$	497.5
500	10^{-4}	$3.6*10^{7}$	$3.6*10^{3}$	1247.5
700	10^{-5}	$2.3*10^{8}$	$2.3*10^{3}$	1747.5
1000	$5*10^{-6}$	$6.9*10^{8}$	$6.9*10^{3}$	2497.5

We see that in each case the actual number of transitions required by the DEVS quantization method exceeds the predicted number of transitions. However, for the choices of optimal quantum shown, the difference is less than an order of magnitude. Further work on DEVS quantization-based methods may be able to reduce this discrepancy.

The different dots in figure 33, show the number of transitions for the DEVS quantization-based simulation, when an optimum quantum is not used for different cell numbers. From figure 33, we can see that when an optimum quantum is not used, the activity by quantum is not equal to the number of DEVS transitions. This justifies using the ratio of actual activity to predicted activity as the measure of accuracy for a simulation.

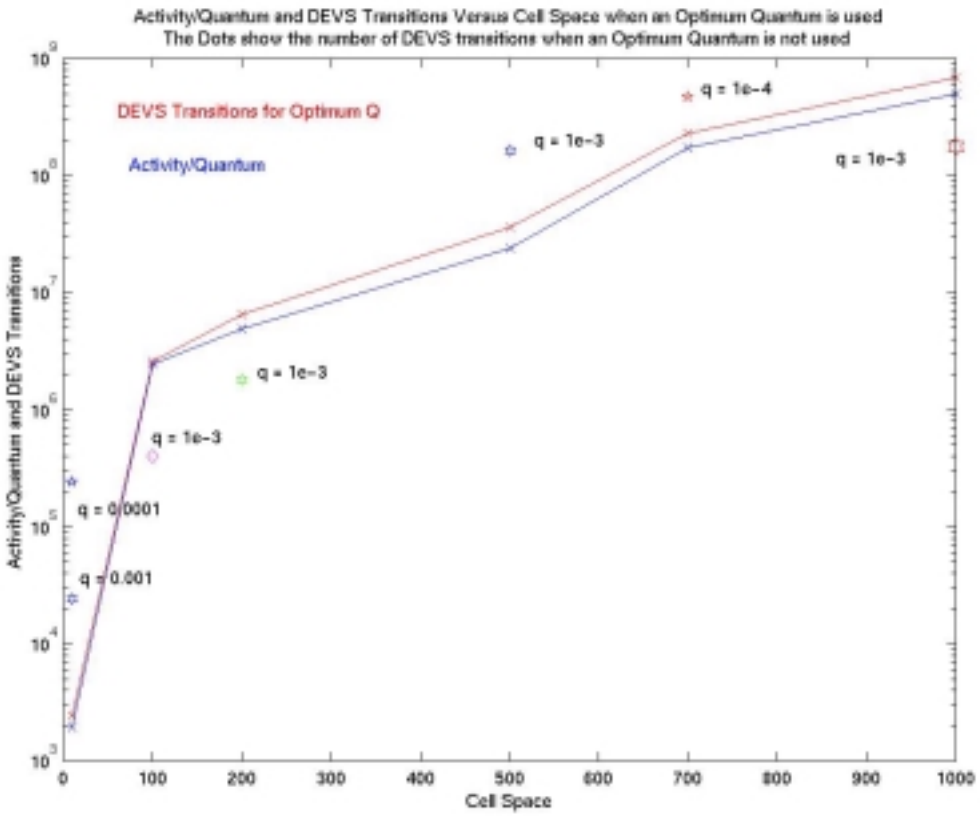

Figure 33 Importance of using the right quantum in DEVS

6.6 Ratio of Execution times

Now to estimate the ratio of execution times of the DTSS and DEVS simulation methods, we will estimate the ratio of their execution times per cell transition. If we let R denote this ratio then we have

$$\frac{DTSS}{DEVS}execution = R\frac{DTSS}{DEVS} \qquad \text{Eqn 6.17}$$

We proceed to estimate the transition time per cell transition

Table 8 shows the number of cell transitions (computed) and the measured execution time for the discrete time simulation. Note that the execution time is measured for calibration purposes using a time step of 1sec. This time step is much greater than required by the Courant condition since employing the latter would result in prohibitive execution times. According to the table, we estimate the execution time required by one cell for one time step of a discrete time simulation as approximately $3 * 10^{-6}$ seconds.

TABLE 8. Cell Transitions and measured Execution time for DTSS

#Cell Transitions	Execution time (in seconds)	Ratio(Seconds per transition)
$8.7*10^5$	2	$2.3*10^{-6}$
$8.0*10^6$	18	$2.25*10^{-6}$
$1.5*10^8$	346	$2.3*10^{-6}$
$4.3*10^8$	1102	$2.56*10^{-6}$
$1.3*10^9$	4836	$3.7*10^{-6}$

Table 9 shows the number of cell transitions and the execution time for the DEVS quantization-based simulation employing the optimum quantum for increasing cell spaces.

TABLE 9. Cell Transitions and measured Execution time for DEVS

#Cells	Optimum quantum to get matching activity	#DEVS transitions	Execution Time (in seconds)	Ratio (Seconds per transition)
100	10^{-4}	$2.6*10^{6}$	65	$2.5*10^{-5}$
200	10^{-4}	$6.5*10^{6}$	247	$3.8*10^{-5}$
500	10^{-4}	$3.6*10^{7}$	1555	$4.3*10^{-5}$
700	10^{-5}	$2.3*10^{8}$	7075	$3*10^{-5}$
1000	$5*10^{-6}$	$6.9*10^{8}$	$2*10^{4}$	$2.8*10^{-5}$

From this table, we estimate the execution time required by one cell for one time step of a DEVS quantization-based simulation as approximately $3 * 10^{-5}$ seconds.

Thus we estimate the ratio

$$R = 3*10^{-6} / 3*10^{-5}$$

$$= 0.1$$

Thus a single DEVS transition takes an order of magnitude more time than a single DTSS transition. From Eqn 6.17 we have

$$\frac{DTSS}{DEVS} execution = .1 \frac{DTSS}{DEVS} \qquad \text{Eqn 6.18}$$

Since, from equations 6.14 and 6.15, the ratio $\frac{DTSS}{DEVS}$ increases at least as fast as order (N), we see that with N sufficiently large, the ratio will exceed unity.

We now consider the ratio of execution times under different conditions.

6.7 Ratio of execution times when both Discrete Time and DEVS are required to be stable

In this section, we will derive the ratio of execution times when the discrete time method is stable and DEVS is also stable. In other words we are just looking for stability in both cases.

Substituting the values of L = 16 and T = 100 seconds in equation 6.13 gives the number of DTSS transitions as

$$DTSS = 0.8N^3 \qquad \text{Eqn 6.19}$$

Therefore, the execution time of discrete time processing is given by

$$DTSS_{exec} = 3*10^{-6}*0.8N^3 = 2.4*10^{-6}N^3 \text{ seconds} \qquad \text{Eqn 6.20}$$

Table 10 shows the values of α for increasing cell-spaces in order to get the optimum quantum for DEVS.

TABLE 10. Values of α to get Optimum Quantum for DEVS

#Cells	Optimum quantum to get matching activity for DEVS	Value of α, so that $(H/(\alpha N))$ equals the optimum quantum for DEVS
100	10^{-4}	1600
200	10^{-4}	800
500	10^{-4}	320
700	10^{-5}	2000
1000	$5*10^{-6}$	3200

From table 10, the average value of α, is approximately 1600. This value of α is needed to make DEVS both stable and accurate.

However, in order for DEVS to be just stable, the value of α estimated from table IX is too large.

So, we take value of α=2 which is given by equation 6.16.

Substituting the value of α = 2 in equation 6.11 gives the number of DEVS transitions as

$$DEVS = 0.5N^2 \qquad \text{Eqn 6.21}$$

Therefore, the execution time of DEVS is given by

$$DEVS_{exec} = 3*10^{-5}*0.5N^2 = 1.5*10^{-5}N^2 \text{ seconds} \quad \text{Eqn 6.22}$$

Now, the ratio of execution times becomes,

$$\frac{DTSS_{exec}}{DEVS_{exec}} = 0.16N \text{ seconds} \quad \text{Eqn 6.23}$$

The value of N, after which DEVS will have an advantage over discrete time, is given by

$$0.16\,N > 1$$

$$\Rightarrow N > 6$$

Table 11 shows the computed values of the execution times of DEVS and DTSS in days for increasing values of cell space.

TABLE 11. Execution Times when DTSS and DEVS are Stable

#Cells	DTSS Execution time (in seconds) from equation 6.20	DEVS Execution Time (in seconds) from equation 6.22
5	$3*10^{-4}$	$3.75*10^{-4}$
10	$2.4*10^{-3}$	$1.5*10^{-3}$
100	2.4	0.15
200	19.2	0.6
300	65	1.35
400	154	2.4
500	300	3.75
600	518	5.4
700	823	7.35
800	1229	9.6
900	1750	12
1000	2400	15
5000	$3*10^{5}$(3.5 days)	375
10000	$2.4*10^{6}$(28 days)	1500

We see that, although for small cell spaces DTSS has slight advantage, as the number of cells exceeds 6, the DEVS quantized-based approach is increasingly more efficient.

6.8 Ratio of execution times when Discrete Time is stable and DEVS is accurate

In section 6.7, we derived the ratio of execution times when both Discrete Time and DEVS are required to be stable.

In this section, we will derive the ratio of execution times when the discrete time method is just stable and DEVS is both stable and accurate.

The formula for the execution time of discrete time method will remain the same.

Therefore, we have,

$$DTSS_{exec1} = 3*10^{-6}*0.8N^3 = 2.4*10^{-6}N^3 \text{ seconds} \quad \text{Eqn 6.24}$$

In section 6.7, we found that, the value of α should be equal to 1600 in order for DEVS to be both stable and accurate.

Therefore, substituting the value of α = 1600 in equation 6.11 gives the number of DEVS transitions as

$$DEVS = 400N^2 \quad \text{Eqn 6.25}$$

Therefore, the execution time of DEVS is given by

$$DEVS_{exec1} = 3*10^{-5}*400N^2 = 0.012N^2 \text{ seconds} \quad \text{Eqn 6.26}$$

From equations 6.24 and 6.26, we have,

$$\frac{DTSS}{DEVS} execution = 2*10^{-4} N \quad \text{Eqn 6.27}$$

Therefore, the value of N, after which, DEVS will have an advantage over discrete time, is given by

$$2*10^{-4} N > 1$$

$$\Rightarrow N > 5000$$

Table 12 shows the computed values of the execution times of DEVS and DTSS in days for increasing values of cell space.

TABLE 12. Execution Times when DTSS is stable and DEVS is accurate

#Cells	DTSS Execution time (in seconds) from equation 6.24	DEVS Execution Time (in seconds) from equation 6.26
2000	$1.9*10^4$	$4.75*10^4$
3000	$6.5*10^4$	$1.1*10^5$(1.25 days)
4000	$1.5*10^5$ (1.77 days)	$1.9*10^5$(2.2 days)
5000	$3*10^5$(3.47 days)	$3*10^5$(3.47 days)
6000	$5.2*10^5$ (6 days)	$4.3*10^5$(5 days)
7000	$8.2*10^5$ (9.5 days)	$5.9*10^5$(6.8 days)
8000	$1.2*10^6$(14.2 days)	$7.6*10^5$(8.8 days)
9000	$1.75*10^6$(20.25 days)	$9.7*10^5$(11.25 days)
10000	$2.4*10^6$(28 days)	$1.2*10^6$(14 days)

We see that although for small cell spaces DTSS has slight advantage, as the number of cells exceeds 5000, the DEVS quantized-based approach is increasingly more efficient.

6.9 Ratio of execution times when both DEVS and DTSS are required to be of the same accuracy

In this section we compute the ratio of execution times when both DEVS and DTSS are required to be of the same accuracy.

In section 5.3, we found that the ratio of number of transitions of DTSS to DEVS in the case of the triangular pulse is equal to

$$\frac{DTSS}{DEVS} = \frac{4Tc}{L^2} N \quad \text{Eqn 6.28}$$

Therefore, from equation 6.1

$$\frac{DTSS}{DEVS} execution = \frac{0.4Tc}{L^2} N \quad \text{Eqn 6.29}$$

Substituting the values of L = 16, c = 1, and T = 100 seconds, equation 6.29 becomes

$$\frac{DTSS}{DEVS} execution = 0.1563N \quad \text{Eqn 6.30}$$

Therefore, the value of N, after which, DEVS will have an advantage over discrete time, is given by

$$0.1563\ N > 1$$

$$\Rightarrow N > 6$$

For discrete time to be at least as accurate as DEVS, we have from equations 5.1, 5.8, and 6.9,

$$\Delta t = \frac{q}{Max.Derivative} = \frac{\frac{H}{N\alpha}}{\left(\frac{cHN}{L^2}\right)} = \frac{L^2}{cN^2\alpha} \quad \text{Eqn 6.31}$$

Since we want DEVS to be accurate, we have from equation 6.26

$$DEVS_{exec2} = 3*10^{-5}*400N^2 = 1.2*10^{-2}\ N^2 \text{ seconds Eqn 6.32}$$

From equations 5.3b and 6.31, we have the number of discrete time transitions as

$$DTSS = \frac{T}{\Delta t} N = \frac{TN}{\frac{L^2}{cN^2\alpha}} = \frac{\alpha TN^3 c}{L^2}$$

which on substituting the values of c=1, T=100 seconds, α=1600 and L = 16 becomes

$$DTSS = 625N^3 \qquad \text{Eqn 6.33}$$

Therefore, we have the approximate execution time of DTSS as

$$DTSS_{exec2} = 3*10^{-6}*625N^3 = 2*10^{-3}\ N^3 \text{ Eqn 6.34}$$

Table 13 shows the computed values of the execution times of DEVS and DTSS in days for increasing values of cell space.

TABLE 13. Execution Times when DTSS and DEVS are required to be of the same accuracy

#Cells	DTSS Execution time (in seconds) from equation 6.34	DEVS Execution Time (in seconds) from equation 6.32
5	0.26	0.3
10	2.0	1.2
100	2000	121
200	$1.7*10^4$	484
300	$5.4*10^4$	1080
400	$1.3*10^5$(1.48 days)	1728
500	$2.6*10^5$(3 days)	3456
600	$4.3*10^5$(5 days)	4320
700	$6.9*10^5$(8 days)	6048
800	10^6(12 days)	7776
900	$1.5*10^6$(17 days)	8640
1000	$2*10^6$(23 days)	$1.2*10^4$
5000	8 years	$3*10^5$(3.5 days)

We see that, although for very small cell spaces DTSS has slight advantage, as the number of cells exceeds 6, the DEVS quantized-based approach is increasingly more efficient.

7. CONCLUSION

In this thesis, we applied the theory of activity (which was discussed in chapter 2) to the problem of diffusion. We proved that the activity measure when divided by the quantum (in DEVS) gives the threshold crossings of the DEVS simulator. We derived the activity for various initial conditions of the diffusion problem, and thereby, found the ratio of the number of transitions in discrete time and DEVS. We then proved that this ratio is of the order of N, where, N is the total number of cells. This proves that DEVS will have increasing advantage over discrete time as the cell-space, N, increases. Finally we derived the ratio of execution times for DTSS to DEVS under the conditions of stability and accuracy.

In section 6.7, we found that, in order for both DEVS and DTSS to be stable, the cell size should be at least 6 in order for DEVS to out-perform DTSS.

In section 6.8, we found that, in order for discrete time method to be just stable and DEVS to be both stable and accurate, the cell size should be at least 5000 in order for DEVS to out-perform DTSS.

In section 6.9, we found that, in order for both DEVS and DTSS to be accurate, the cell size should be at least 6 in order for DEVS to out-perform DTSS.

Therefore, from sections 6.7, 6.8 and 6.9, we conclude that the breakeven cell size for stability is approximately 6. When accuracy is also required, the breakeven cell size is between 6 and 5000, which is quite a broad range. So, further research needs to be done to compare DTSS and DEVS for equal accuracy.

8. REFERENCES

[1] B.P. Zeigler, T.G. Kim, H. Praehofer. Theory of Modeling and Simulation. New York, NY, Academic Press, 2000.

[2] J. Nutaro, B.P. Zeigler, R. Jammalamadaka, S. Akrekar. "Discrete Event Solution of Gas Dynamics within the DEVS Framework: Exploiting Spatiotemporal Heterogeneity", International Conference for Computational Science, Melbourne, Australia, 2003.

[3] J. Nutaro. "Parallel Discrete Event Simulation with Applications to Continuous Systems", PH. D. dissertation.

[4] V.P. Mikhailov. Partial differential equations. Mir Publishers, 1978.

[5] G.E. Forsythe, W.R. Wasow. Finite-Difference Methods for Partial Differential Equations. New York, NY, John Wiley and Sons INC, 1960.

[6] H. Anton. Calculus. New York, NY, John Wiley and Sons INC, 1988.

[7] W.E. Scheisser. The Numerical Method of Lines. San Diego, California, Academic Press, 1991.

[8] S.J. Farlow. Partial Differential Equations for Scientists and Engineers. Mineola, New York, Dover Publications, 1993.

[9] E. Kofman. "A Second Order Approximation for DEVS Simulation of Continuous Systems", Simulation, 78(2): 76-89, 2002.

[10] A. Ralston, P. Rabinowitz. A First Course in Numerical Analysis. Dover Publications, 2001.