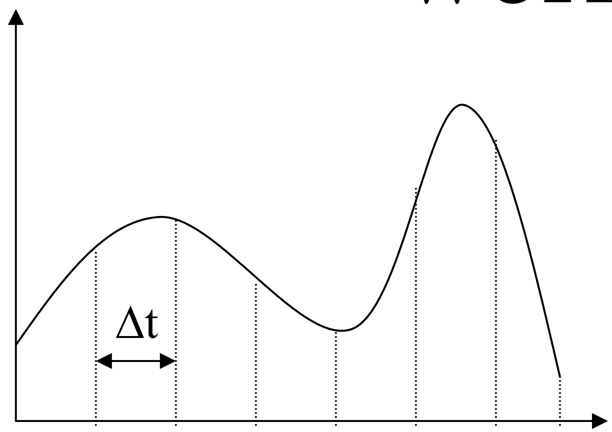
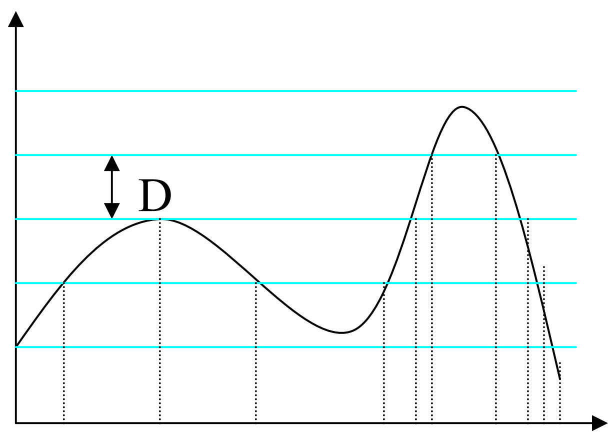
Discrete event approximations of continuous systems

Jim Nutaro
University of Arizona

Discrete time/discrete event world views



Discrete time approximations of continuous systems use discrete time and continuous states.



Discrete event approximations of continuous systems use discrete states and continuous time.

Discrete event models

Discrete event models describe systems in terms of significant changes to the system state.

Significant changes can occur at any time.

An event corresponds to a significant change of state. Internal events describe autonomous (input free) system behavior. External events describe the system’s input response.

A time advance function is used to schedule internal events.

An output function determines system outputs that coincide with internal events.

DEVS-Euler

Want to simulate time-invariant system

$$dy/dt = f(y)$$

Can approximate with backwards Euler formula

$$y(t+h) = y(t) + hf(y(t+h)),$$

where h is the integration time step.

Pick an integration quantum

$$D = |y(t+h)-y(t)|.$$

Approximate time for y(t) to change by D with

$$ta(y(t)) = D/|f(y(t))|$$

DEVS-Euler

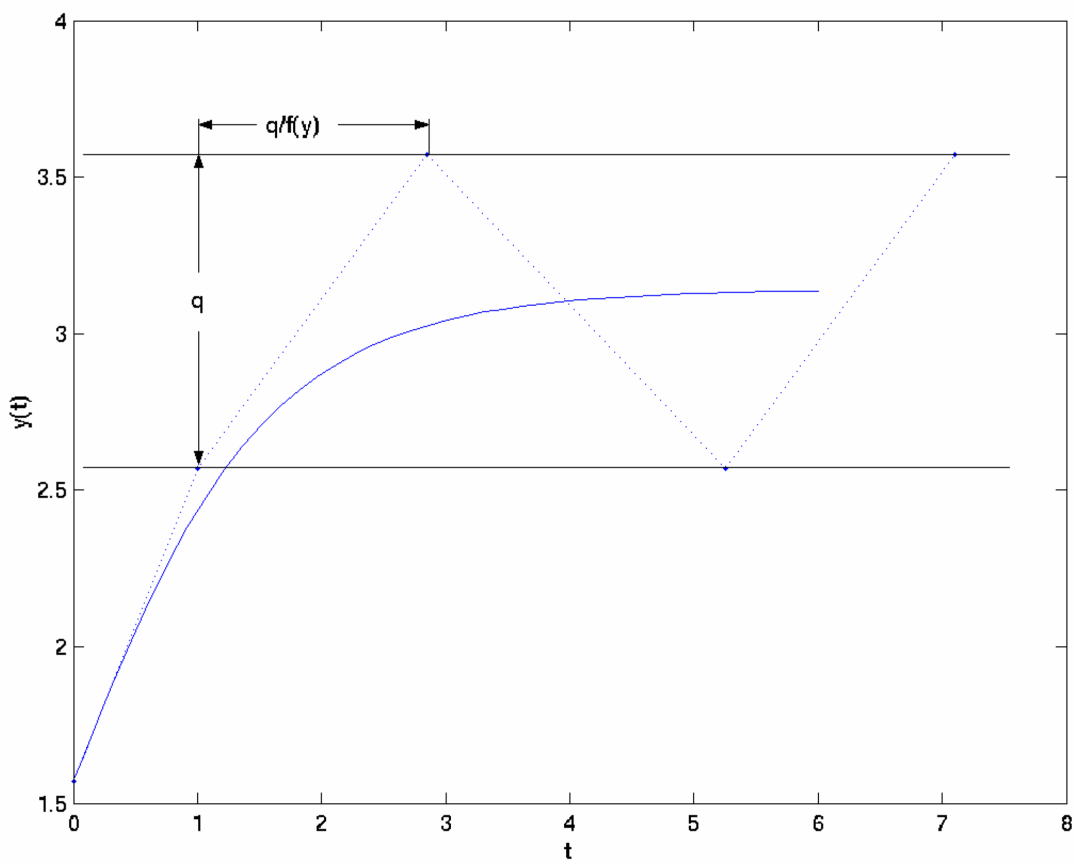
Gives a discrete event based simulation where

$$y(t+ta(y(t))) = y(t)+D\text{sgn}(f(y(t)))$$

$$ta(y(t)) = D/|f(y(t))|$$

If $y(t) = 0$, $ta(y(t))$ is infinite. Since system has stabilized, no more computation needs to be performed.

Example: dy/dt = sin(y)



Properties

- For a stable, time invariant system $dy/dt = f(y)$ there exists a quantum size small enough that the corresponding DEVS-Euler approximation is stable.
- The DEVS-Euler scheme is convergent.
- For a linear, stable, time invariant system $dy/dt = -ky$, the corresponding DEVS-Euler scheme is unconditionally stable.
- Truncation error is proportional to the quantum size.

Representation in DEVS

- DEVS is a systems formalism for describing discrete event systems
- DEVS models consist of atomic components and coupled components
- Coupled components are constructed from atomic and other coupled components
- For solving PDEs, atomic components will consist of discrete event integrators and coupling (e.g., flux computing) functions

Instantaneous function f(**x**)

<x1,x2,x3,…,xn>, ta = ∞

input x1’ → external event

<x1’,x2,x3,…,xn>, ta = 0

internal event

output f(x1’,x2,…,xn)

<x1’,x2,x3,…,xn>, ta = ∞

DEVS-Euler integrator

<q,ql,x,sigma>, ta = sigma

input x’

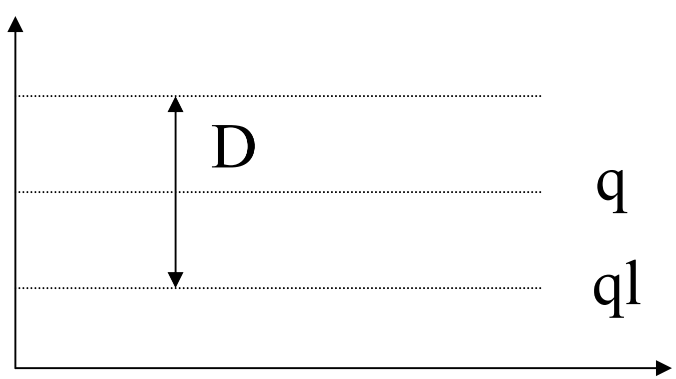
external event
with elapsed time e

<q+ex,ql,x’,|D–(q+ex-ql)|/|x’|>

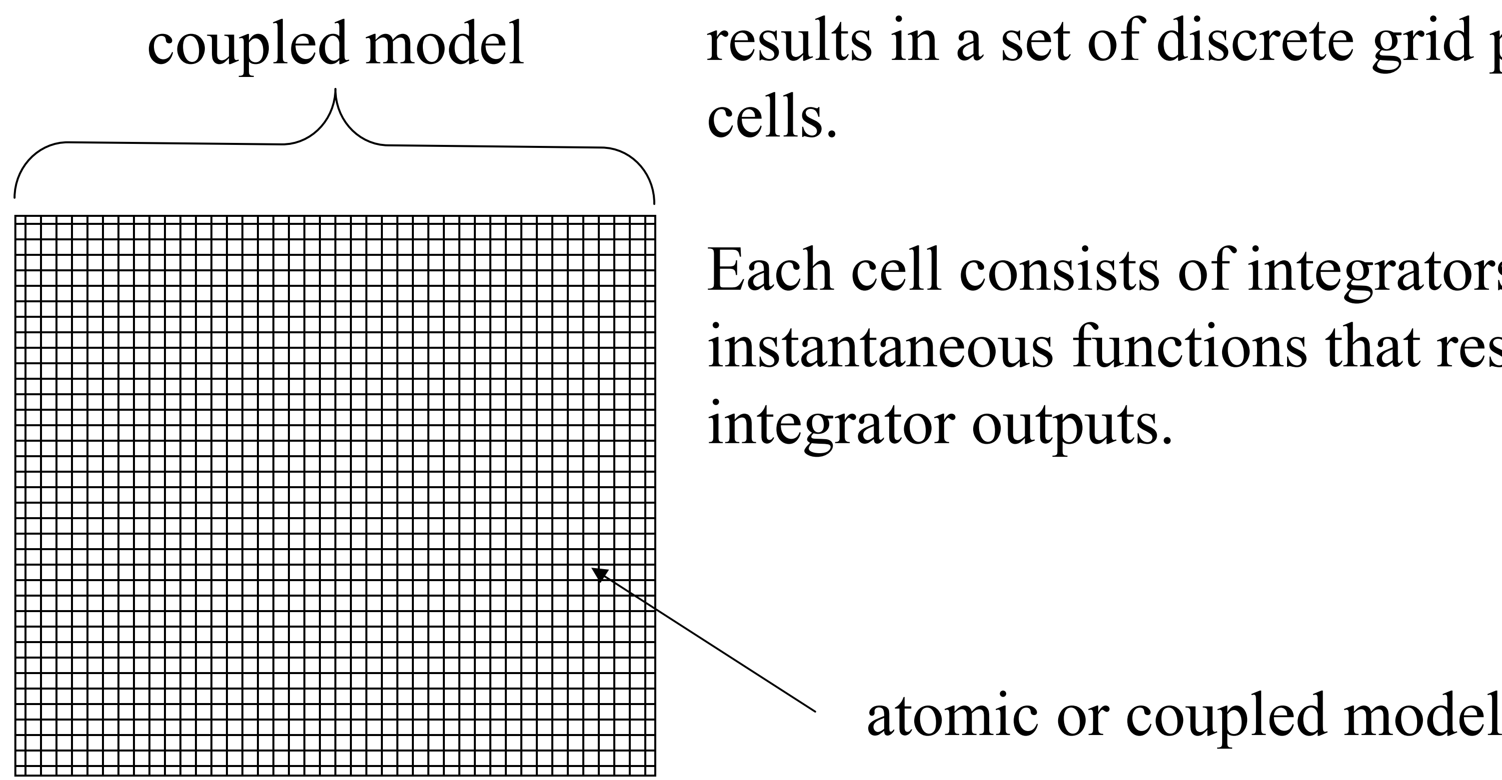
internal event

output ql+Dsgn(x)

<q+ta*x,q+ta*x,x,D/|x|>



Application to PDEs



Approximation of spatial derivative results in a set of discrete grid points or cells.

Each cell consists of integrators and/or instantaneous functions that respond to integrator outputs.

Activity

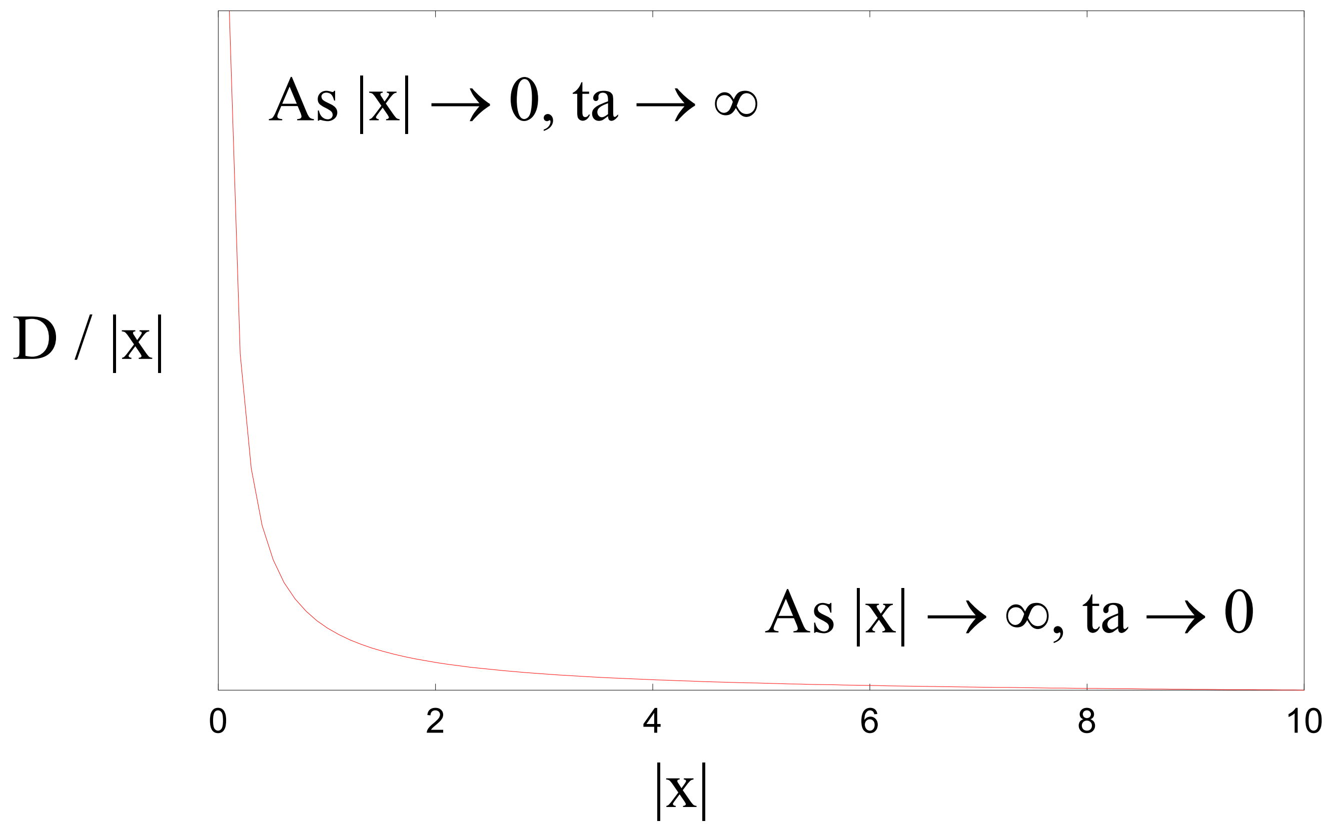
Observation: Most spatially distributed systems are not active everywhere all the time.

Example: forest fires, shock waves, diffusive phenomena.

Areas where the system is not changing are inactive: characterized by small derivatives, large time advances.

Areas where the system is changing are active: characterized by large derivatives, small time advances.

Intrinsic activity tracking



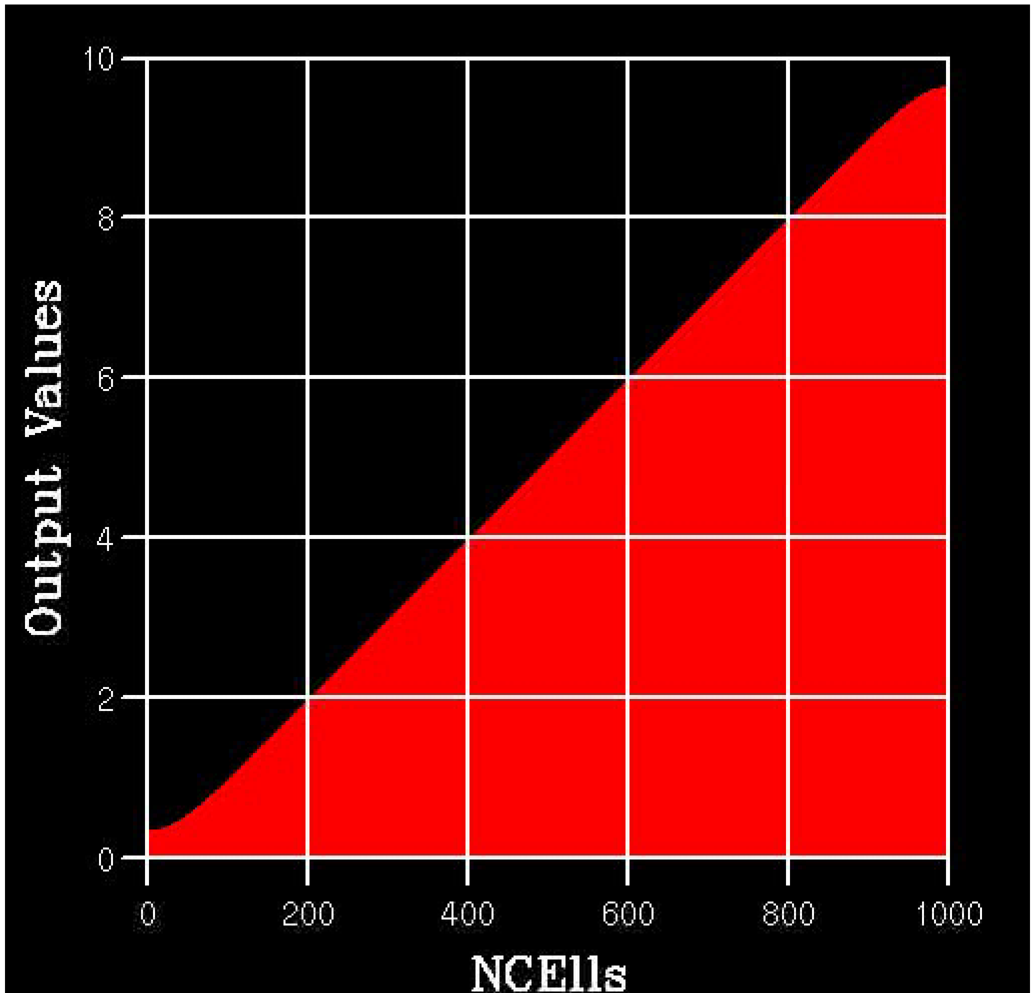
Intrinsic activity tracking

- Areas where the solution is changing will consist of cells with relatively small time advance, frequent state changes – frequent events
- Areas where solution is unchanging will consist of cells with relatively large time advance, infrequent state changes – infrequent events
- More computational resources go to areas that are active, less to areas that are not active

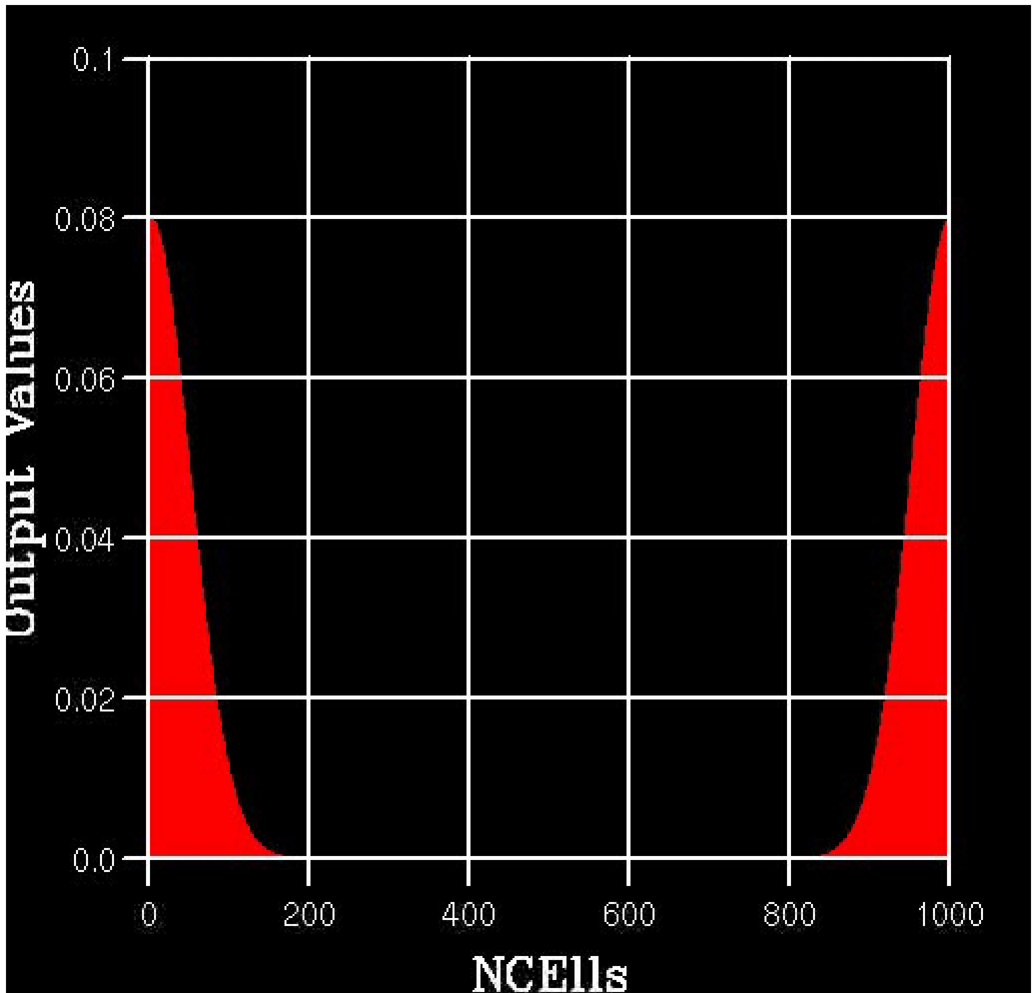
Demonstration of activity tracking

- Use heat equation as example since I had an animation for it
- Solution used center differences in space, DEVS-Euler to integrate through time at each cell
- Same effect can be seen looking at plots of computational effort as function of time and space for Godunov's method applied to Sod shock tube problem

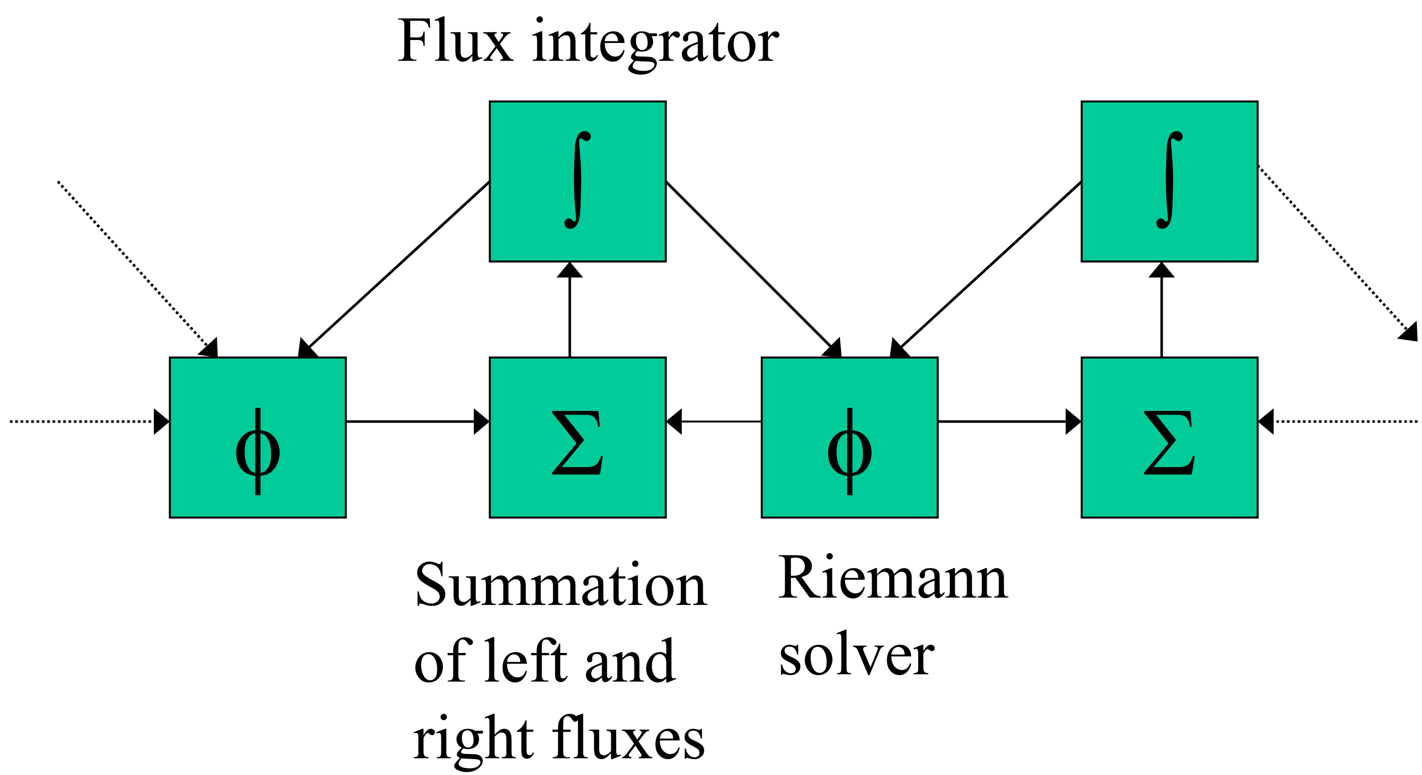
Simulation of diffusion



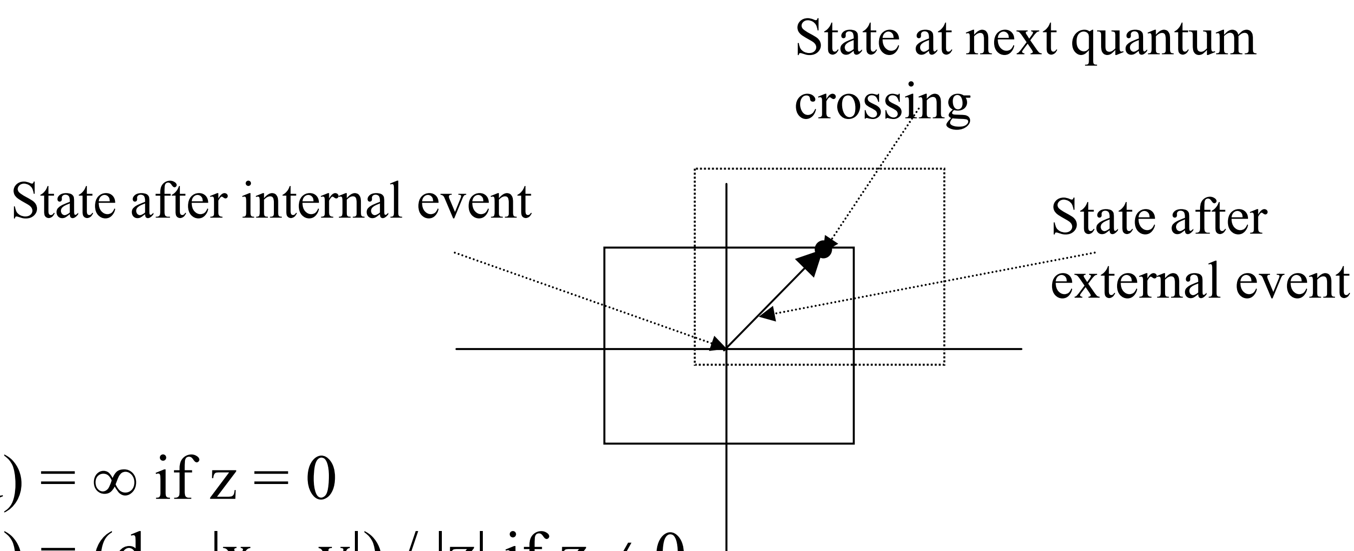
Allocation of computational resources



Structure of Godunov scheme



Time advance for multi-dimensional state



$T(x,y,z,d) = \infty$ if $z = 0$
$T(x,y,z,d) = (d - |x - y|) / |z|$ if $z \neq 0$.

Define a function $\Gamma(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{d}) = \min \{ T(x_i,y_i,z_i,d_i) \}$ where **x**, **y**, **z**, and **d** are vectors with N elements and the index i ranges over N.

Courant condition

Some insight can be gained by looking at a similar system: consider up-winding scheme for solving

$$ut + aux = 0,$$

which is

$$u(t+h,x) = u(t,x) + (a\Delta t / \Delta x)(u(t,x)-u(t,x-\Delta x)).$$

Stability requires

$$a\Delta t / \Delta x <= 1.$$

Setting $D = |u(t+h,x)-u(t,x)|$, solve for Δt, and substitute…

Courant condition

Sufficient (not necessary) condition for stability is

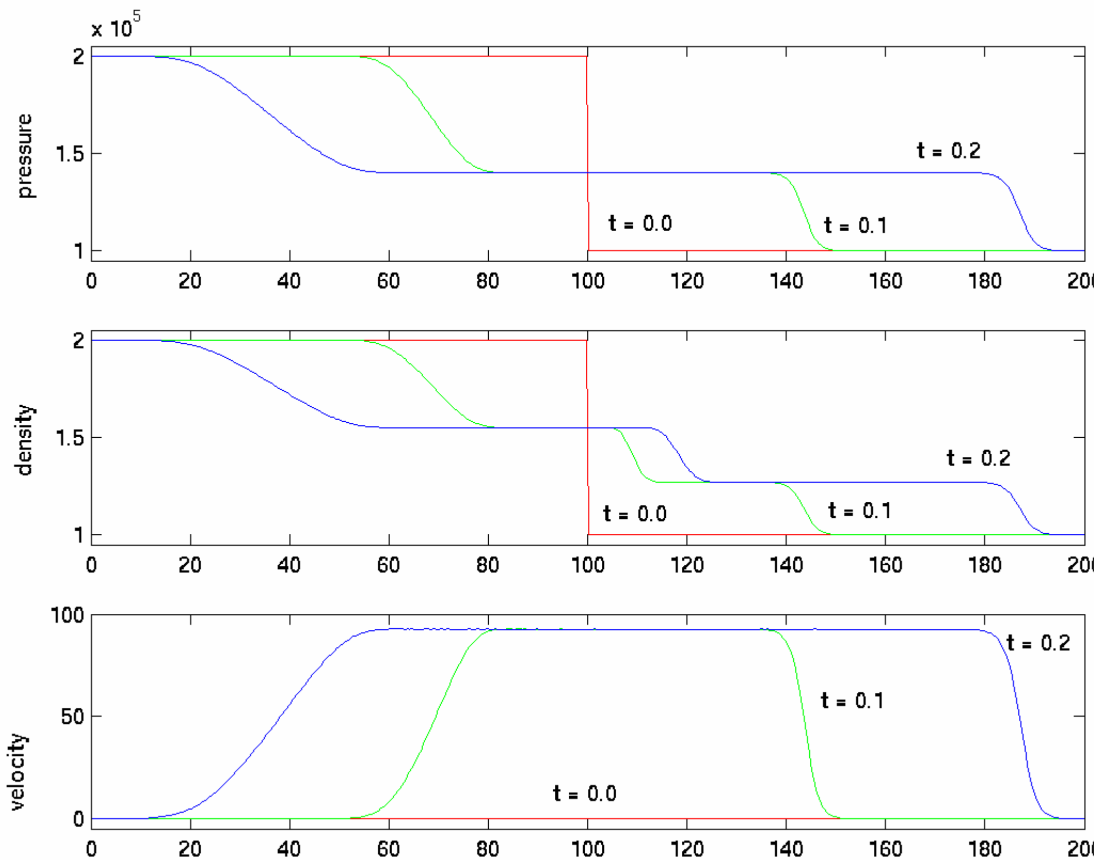
$$D <= |u(t,x)-u(t,x- \Delta x)|.$$

So system tends towards equilibrium until system nears equilibrium.

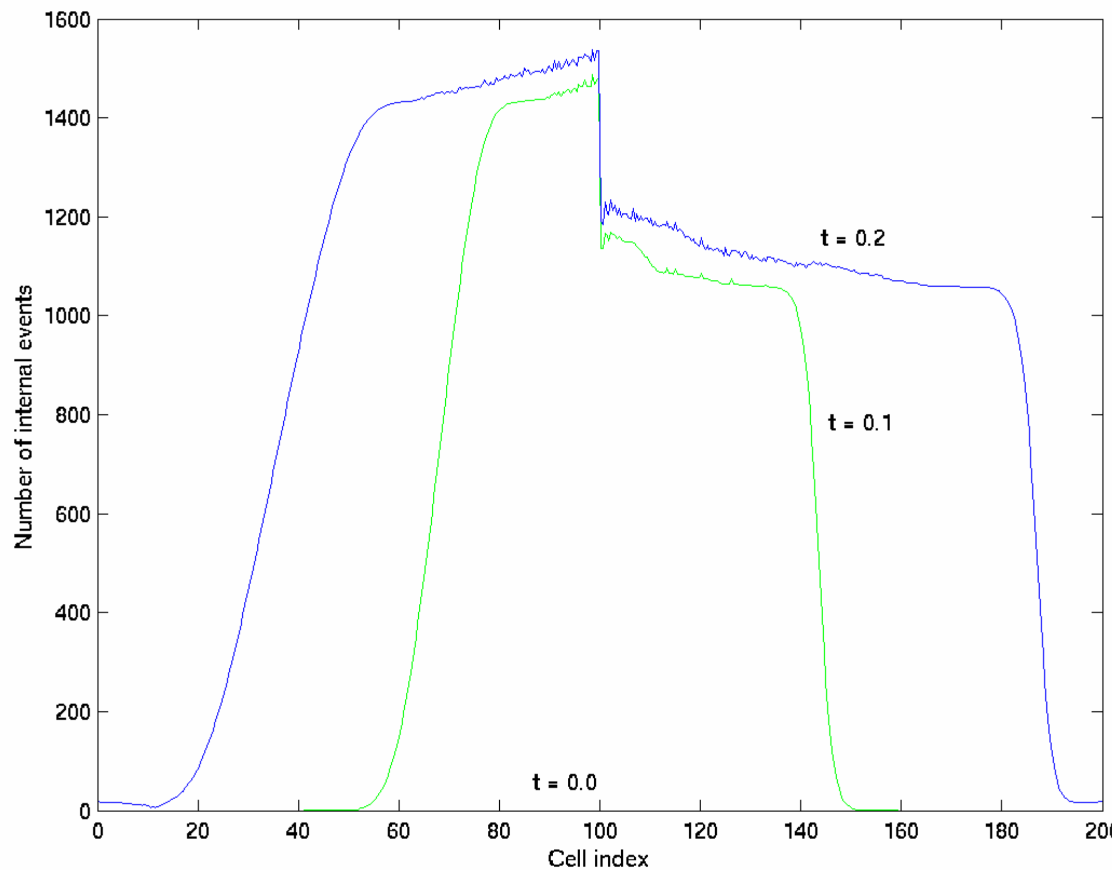
Condition states upper bound on approximation of a zero derivative. Can not reach zero due to discrete state space. Might not include zero.

Ensure stability by 'clamping' derivative to zero when reach closest point to zero allowed by choice of quantum size.

Sod shock tube simulation



Computational effort



Comparison with discrete time solution

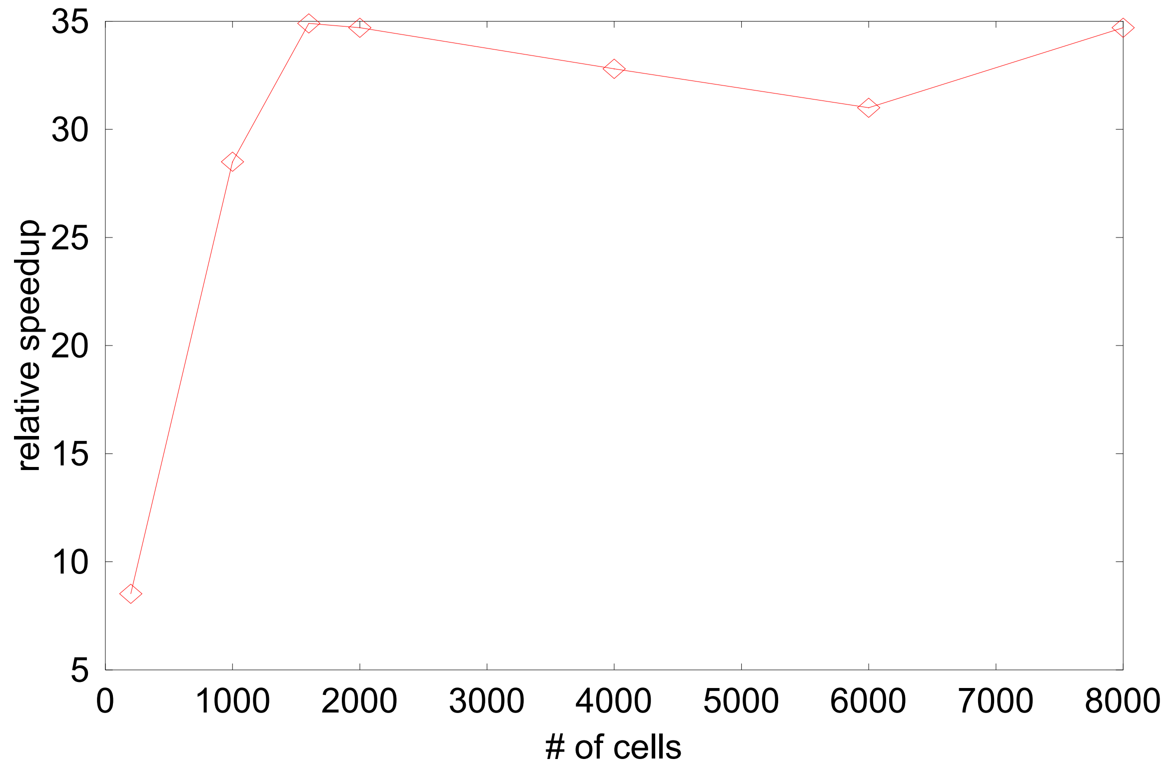
Assumed that discrete time scheme would require a time step equal to the smallest time advance of the discrete event system to resolve the same set of events.

Fixed quantum size and performed computation with a number of different cells.

Computed relative speedup as

DTSS exec. time / DEVS exec. time

Relative speedup



Time advance