Numerical Solution of Nonlinear Differential Equations in Musical Synthesis

David Yeh

Center for Computer Research in Music and Acoustics (CCRMA)
Stanford University

CHESS Seminar, UC Berkeley EECS
March 11, 2008

Nonlinear Differential Equations in Musical Synthesis

Nonlinear Differential Equations Describe Musical Physics

- Vibrating systems, equations of motion for mechanical components described by systems of Ordinary Differential Equations (ODE), nonlinear in general.
- Distributed phenomena such as strings, resonant tubes (or "bore") of wind instruments, membranes and plates are modeled by Partial Differential Equations (PDE).

Nonlinear Differential Equations Also Describe Musical Electronics

- Music in the electronic and recorded era utilizes electronics as a musical “filter.” Often the nonlinearities of these effects are an intrinsic part of the sound.
- Examples: vacuum tube mic preamps, dynamic range compressors, LP records, magnetic tape.
- Especially emblematic of the distorted electric guitar sound.





General Ordinary Differential Equations

- Differential Algebraic Equations (DAE), a special class of ODE, is a natural way to describe mechanical and circuit system equations.

$$M\dot{x} = f(t, x)$$

  where M ("mass matrix") in general is singular, x is the state vector, $f(t, x)$ is a nonlinear vector function.
- For linear constant-coefficient differential equations,

$$\dot{x}(t) = Ax(t) + Bu(t)$$

  Eigenvalues of A are poles of system
  These are digital filters, an efficient special case of ODEs.

Numerical Integration Methods

Solve ODE by integrating the derivative and solving for the state variable.

Different ways of approximating the derivative

$$\frac{\mathrm{d}y}{\mathrm{d}t} = \lim_{T \to 0} \frac{y(t+T) - y(t)}{T}$$

using a nonzero T

- Euler
- Trapezoidal Rule/Bilinear Transform/Tustin's Approximation
- Taylor series methods
- Polynomial approximations
- Runge-Kutta
- Extrapolation

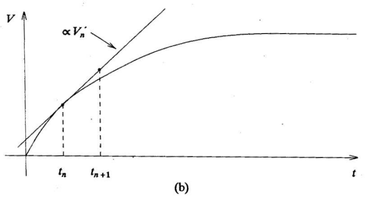
Euler methods

First order approximation of derivative as difference

T is stepsize/sampling period, x is system state.

Forward Euler

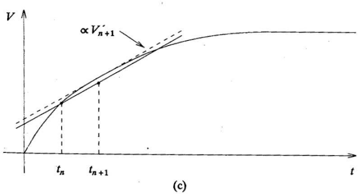
$$x[n] = x[n-1] + T\dot{x}[n-1]$$



Derivative at the previous time step
Explicit form.

Backward Euler

$$x[n] = x[n-1] + T\dot{x}[n]$$



Derivative at the current time step
Implicit computation.

Figures from McCalla (1988, Kluwer)

Implicit Trapezoidal Rule Integration

Average of derivative from current and previous time step

$$x[n] = x[n-1] + \frac{T}{2}\left(\dot{x}[n] + \dot{x}[n-1]\right),$$

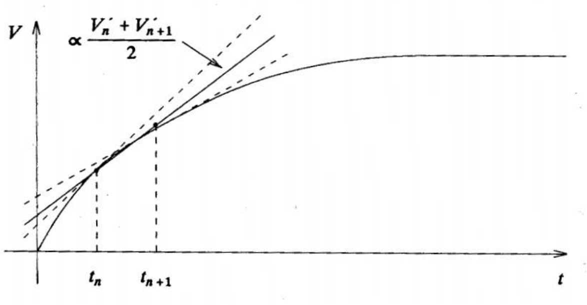


Figure from McCalla (1988, Kluwer)

Second order method

Explicit 4th Order Runge-Kutta (RK4)

$$
\begin{aligned}
k_1 &= Tf(n-1, v_{n-1}),\\
k_2 &= Tf(n-1/2, v_{n-1} + k_1/2),\\
k_3 &= Tf(n-1/2, v_{n-1} + k_2/2),\\
k_4 &= Tf(n, v_{n-1} + k_3),\\
f(n,v) &= \dot{v}(n,v) = \frac{V_i[n] - v}{RC} - 2\frac{I_s}{C}\sinh(v/V_t),\\
v_n &= v_{n-1} + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}
\end{aligned}
$$

- Highly popular due to 4th order accuracy
- Simplicity due to single-step, explicit method
- Expensive: Requires 4 function evals of $f(t, v)$ per step
- Note: Input is 2x upsampled relative to output – not desirable because of bandwidth expansion in nonlinear systems.

Stability Depends on Eigenvalues of System

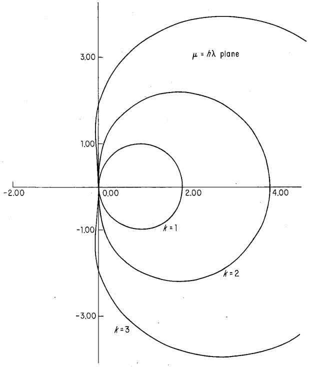


Fig. 11.6 Regions of absolute stability for stiffly stable methods of orders one through three. Methods are stable outside of closed contours.

Figure from Gear (1971, Prentice Hall)

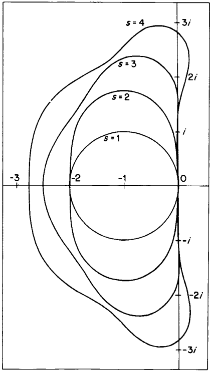


Figure 351 Explicit Runge–Kutta stability boundaries.

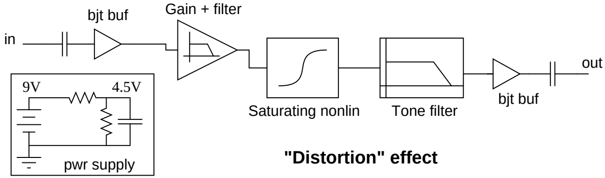
Figure from Butcher (1987, Wiley)

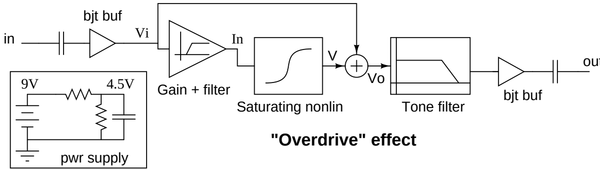
- Implicit methods stable outside region
- s=1 is Backward Euler
- Unstable λ (right half plane) may become stable

- Explicit methods stable within region
- s=1 is Forward Euler, s=4 is RK4
- Places limit on max stable λ given T

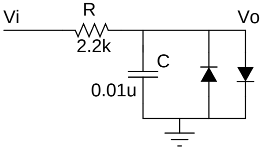
Overview of some guitar distortion effects

Boss DS-1 distortion and Ibanez Tube Screamer TS-9 use building blocks common to many guitar effects circuits
Linear filters and saturating nonlinearities can be used in a simplified digital implementation





Diode Clipper Ordinary Differential Equation



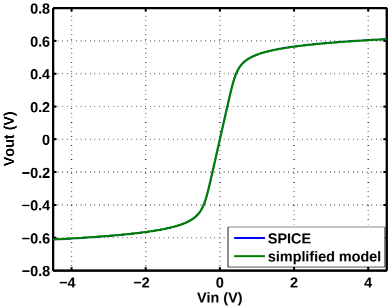
$$\frac{V_i - V_o}{R} = C\frac{\mathrm{d}V_o}{\mathrm{d}t} + I_s(e^{V_o/V_t} - 1) - I_s(e^{-V_o/V_t} - 1)$$

$$\frac{\mathrm{d}V_o}{\mathrm{d}t} = \frac{V_i - V_o}{RC} - 2\frac{I_s}{C}(\sinh(V_o/V_t))$$

- Diode equation: good description of diode behavior

$$I_d = I_s(e^{V/V_t} - 1)$$

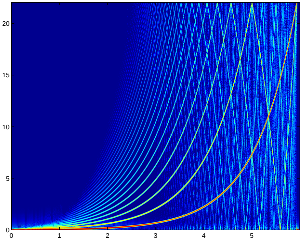
- Low pass with diode limit
- SPICE N914 diode model uses $I_s = 2.52\text{nA}$ and $V_t = 45.3\text{mV}$
- Saturating clip: power supply limits signal to 4.5 V.



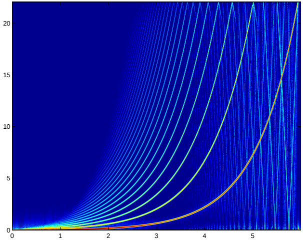
Newton's method solution for V_o when $\frac{\mathrm{d}V_o}{\mathrm{d}t} = 0$

Choose a Sampling Rate

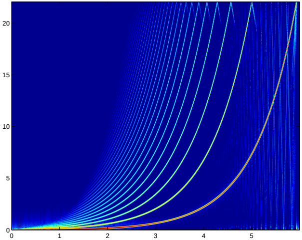
No oversampling



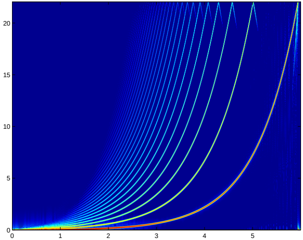
Oversample by 2



Oversample by 4

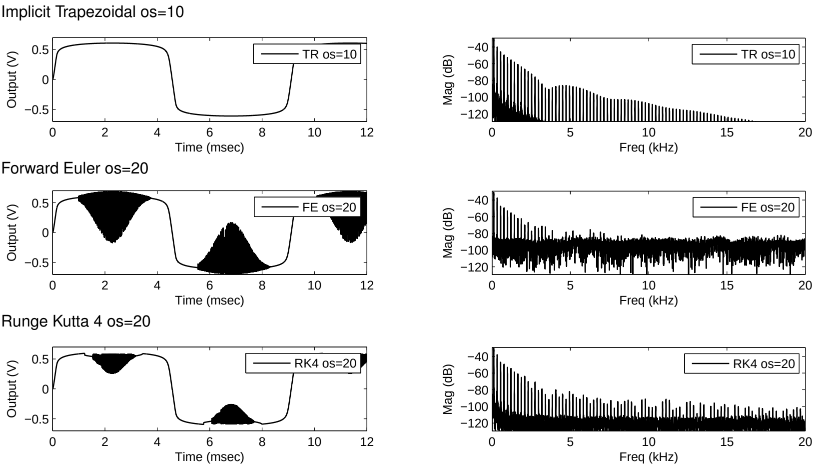


Oversample by 8



Compare Methods

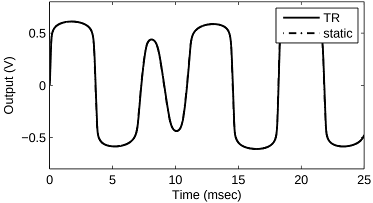
Explicit methods are unstable for a pole that moves in and out of audio band.



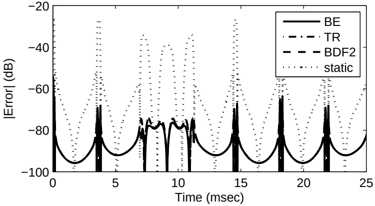
Compare Methods

Implicit methods are about the same when oversampled to avoid aliasing.

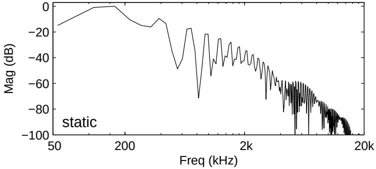
Trapezoidal versus DC nonlinearity



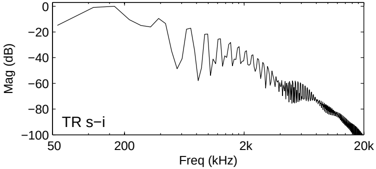
Compare error of BE, TR, DBF2, DC nonlinearity



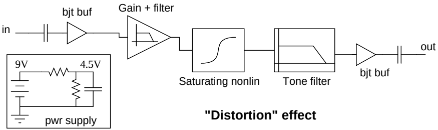
Spectral peaks using DC nonlinearity



Spectral peaks using Trapezoidal



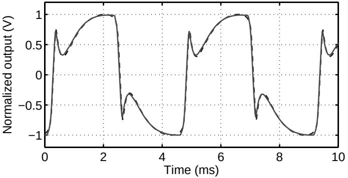
Compare with Measurement

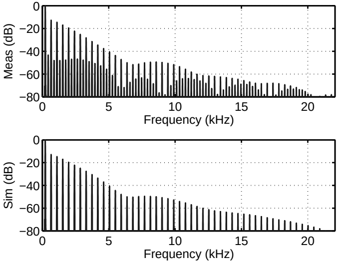


"Distortion" effect

VS

A simple model including only the dominant nonlinearity using trapezoidal integration at OS=8 (dashed line) comes very close to the real thing (solid line)





The K-Method Approach

Borin et al. (2000), Fontana et al. (2004)

- Solution to system of ODEs and nonlinear relations
- K-Method because it operates on K-variables; “K” for Kirchhoff
  - State-space description of system.
  - Implicit method discretization for stability
  - Solve implicit equation to make it explicit
  - Result: state update equations
- Nonlinear formulation of state space

K-Method formulation

- Express state derivative as linear combination of state, input, and output from nonlinearity
  - State is x
  - Input is u
  - Output of nonlinearity is v

$$\dot{x} = Ax + Bu + Cv,$$

where

$$v = f(w), \qquad w = Dx + Eu + Fv$$

- Write $\dot{x}$ as sx

$$sx = Ax + Bu + Cv$$

- Discretize by an implicit method for stability
  - Backward Euler: $s = \alpha(1 - z^{-1})$
  - Bilinear Transform: $s = \alpha \frac{1-z^{-1}}{1+z^{-1}}$

K-Method formulation computational summary

Linear combination of variables -> nonlinearity -> linear combination for state output

- Per sample given input u and state x do

$$p_n = DH(\alpha I + A)x_{n-1} + (DHB + E)u_n + DHBu_{n-1} + DHCv_{n-1}$$
$$v_n = g(p_n)$$
$$x_n = H(\alpha I + A)x_{n-1} + HB(u_n + u_{n-1}) + HC(v_n + v_{n-1})$$

where $H = (\alpha I - A)^{-1}$, $v_n = g(p_n)$ is some implicitly defined transfomation of $f(w)$ due to the discretization.

- Output y is generally expressed as

$$y_n = A_e x_n + B_e u_n + C_e v_n$$

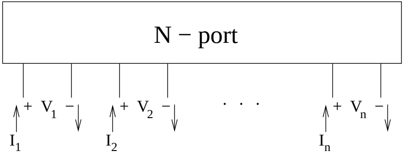
Wave digital filters model equivalent circuits.

Overview

- Fettweis (1986), Wave Digital Filters: Theory and Practice.
- Wave Digital Filters (WDF) mimic structure of classical filter networks.
- Element-wise discretization and connection strategy
- Modeling physical systems with equivalent circuits.
  - Piano hammer mass spring interaction
  - Real time model of loudspeaker driver with nonlinearity
  - Multidimensional WDF solves PDEs
- Ideal for interfacing with digital waveguides (DWG).

Classical Network Theory

N-port linear system is basis of WDF formulation.



- Describe a circuit in terms of voltages (across) and current (thru) variables
- General N-port network described by V and I of each port
- Impedance or admittance matrix relates V and I

$$\begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_N \end{pmatrix} = \underbrace{\begin{pmatrix} Z_{11} & Z_{12} & \dots & Z_{1N} \\ Z_{21} & \ddots & & Z_{2N} \\ \vdots & & & \vdots \\ Z_{N1} & \dots & & Z_{NN} \end{pmatrix}}_{\mathbf{Z}} \begin{pmatrix} I_1 \\ I_2 \\ \vdots \\ I_N \end{pmatrix}$$

Classical Network Theory

WDF uses wave variable substitution and scattering.

$$A = V + RI \qquad V = \frac{A+B}{2}$$

$$B = V - RI \qquad I = \frac{A-B}{2R}$$

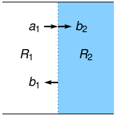
- Variable substitution from V and I to incident and reflected waves, A and B, and a port impedance R
- An N-port gives an $N \times N$ scattering matrix
- Allows use of scattering concept of waves
- Matching port impedances eliminates wave reflections
- Adaptation (Sarti and De Poli 1999) refers to matching the discretized DC impedance of the element (i.e. $T/2C$ for the capacitor)

Wave Digital Elements

Basic one port elements are derived from reflection between wave impedances.

- Work with voltage wave variables b and a. Substitute into Kirchhoff circuit equations and solve for b as a function of as.
- Wave reflectance between two impedances is well known

$$\rho = \frac{b}{a} = \frac{R_2 - R_1}{R_2 + R_1}$$



- Define a port impedance R_p
- Input wave comes from port and reflects off the element's impedances.
  - Resistor $Z_R = R$, $\rho_R(s) = \frac{1 - R_p/R}{1 + R_p/R}$
  - Capacitor $Z_C = \frac{1}{sC}$, $\rho_C(s) = \frac{1 - R_p Cs}{1 + R_p Cs}$
  - Inductor $Z_L = sL$, $\rho_L(s) = \frac{s - R_p/L}{s + R_p L}$

Wave Digital Capacitor from Bilinear Transform

Eliminate instantaneous dependence between input and output.

- Plug in bilinear transform

$$\frac{b_n}{a_n} = \frac{1 - R_p C \frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}}}{1 + R_p C \frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}}}$$

$$(1 + R_p C \tfrac{2}{T}) b[n] + (1 - R_p C \tfrac{2}{T}) b[n-1] = (1 - R_p C \tfrac{2}{T}) a[n] + (1 + R_p C \tfrac{2}{T}) a[n-1]$$

- Choose R_p to eliminate depedence of $b[n]$ on $a[n]$, e.g., $R_p = \frac{T}{2C}$, resulting in:

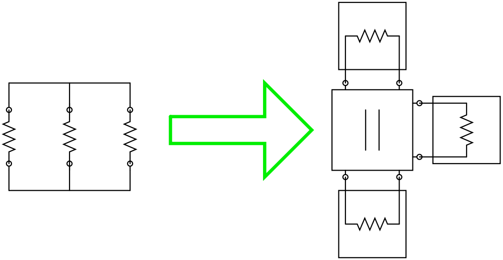
$$b[n] = a[n-1]$$

- Note that chosen R_p exactly the instantaneous resistance of the capacitor when discretized by the bilinear transform

Adaptors

Adaptors perform the signal processing calculations.

- Treat connection of N circuit elements as an N-port
- Derive scattering junction from Kirchhoff's circuit laws and port impedances determined by the attached element
- Parallel and series connections can be simplified to linear complexity
  - Dependent port - one coefficient can be implied,
  - Reflection free port - match impedance of one port to eliminate reflection



Connection Strategy

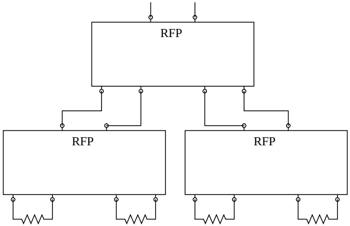
Avoid delay-free loops by connecting adaptors as a tree.

- De Sanctis, et al. (2003), Binary Connection Tree - implement WDF with three-port adaptors
- Karjalainen (2003), BlockCompiler - describe WDF in text, produces efficient C code

Scheduling to compute scattering :

- Directed tree with RFP of each node connected to the parent

1. Label each node (a, b, c, ...)
2. Label downward going signals *d* by node and port number
3. Label upward going signals *u* by node
4. Start from leaves, calculate all *u* going up the tree
5. Then start from root, calculate all *d* going down the tree



Nonlinearity

Formulation of nonlinear WDF has received much attention in the literature.

- Meerkötter and Scholz (1989), *Digital Simulation of Nonlinear Circuits by Wave Digital Filter Principles.*
- Sarti and De Poli (1999), *Toward Nonlinear Wave Digital Filters.*
- Felderhoff (1996), *A New Wave Description for Nonlinear Elements.*
- Petrausch and Rabenstein (2004), *Wave Digital Filters with Multiple Nonlinearities.*
- Either conceive as nonlinear resistor or dependent source
- Introduces a delay free loop, which must be solved as a system of equations in wave variables

Nonlinear Conductance

Substitute wave variables into nonlinear expression and solve for reflected wave.

- Current is a nonlinear function of voltage, $i = i(v)$
- In wave variables

$$a = f(v) = v + R_p i(v)$$
$$b = g(v) = v - R_p i(v)$$

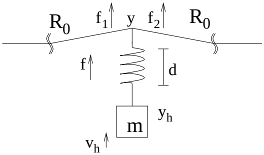
- Substituting wave variables into Kirchhoff variable definition of nonlinear resistance and solving for $b(a)$ if f is invertible

$$b = b(a) = g(f^{-1}(a))$$

- Port resistance R_p can be chosen arbitrarily
- Instantaneous dependence exists regardless of R_p

Nonlinear Piano Hammer

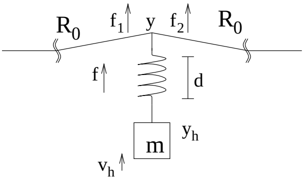
Derive a computational model based upon physical arguments.



- Pianist strikes key, hammer launches with initial velocity
  - Model hammer as a mass in flight that collides with string
  - Assume a digital waveguide string with velocity waves
  - Felt compression is modeled as a nonlinear spring (Hooke's law)
- Switched model between two configurations
  - When hammer contacts string, mass and nonlinear spring act as loaded junction
  - When hammer leaves string, junction no longer exists - the left and right delay lines fuse

Hammer-loaded waveguide junction equations

Interpretation: This is a nonlinear dynamical system.



$$\dot{d} = v_h + \frac{1}{2R_0} f - v_w \qquad (1a)$$

$$\dot{v}_h = \ddot{y}_h = \frac{1}{m_h} f \qquad (1b)$$

$$f = -kd^{\gamma} \qquad (1c)$$

- Only valid when $d >= 0$
- Incoming waves from string: $v_w = (v_1^+ + v_2^-)$
- f is negative from (1c), contact causes downward force on hammer
- (1a) and (1b) are dynamical equations for this system
- (1c) is nonlinear relation constraining f and d

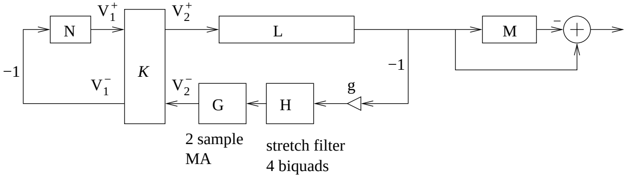
STK Implementation

Hammer with nonlinear felt is a nonlinear scattering junction in the DWG string.

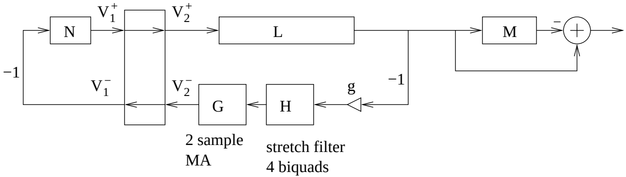
http://ccrma.stanford.edu/software/stk/

- Borrow comb filter and loop filter from StiffKarp in STK
- Split loop delay line at strike position

In contact ($d >= 0$) use K-method



No contact ($d < 0$), pass through signals through K-block



- Commute the velocity wave reflection -1 through V_1^- and V_2^-

$$v_1^- = -v + v_1^+$$
$$v_2^+ = v + v_2^-$$
$$v_w = v_1^+ - v_2^-$$

Nonlinear Piano Hammer (Borin et al. 2000)

Choosing the right state variable is important in the K-method formulation.

- State is felt compression d and hammer velocity v_h
  $\mathbf{x} = \begin{bmatrix} d & v_h \end{bmatrix}^T$
    - Input is sum of incoming waves from two waveguides
      $\mathbf{u} = v_w$
    - Output of nonlinearity is $\mathbf{v} = f$
    - K-method is state-space like formulation with an additional term for the nonlinear contribution
    - (Note: Variable naming is changed from the paper to be consistent with a state-space convention)

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} + \mathbf{Cv}$$

$$\begin{bmatrix} \dot{d} \\ \dot{v}_h \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d \\ v_h \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \end{bmatrix} \mathbf{u} + \begin{bmatrix} \frac{1}{2Z_0} \\ \frac{1}{m_h} \end{bmatrix} \mathbf{v}$$

Nonlinear Piano Hammer

Computational operations are multiply-accumulates and table lookup.

$$\boxed{p_n = d_{n-1} + T v_{h,n-1} - \frac{T}{2}(u_n + u_{n-1}) + (\frac{T}{4Z_0} + \frac{T^2}{4m_h}) f_{n-1}}$$

Find the force $\boxed{f_n = g(p_n)}$ given by the implicit description:

$$k(\mathbf{K} f_n + p_n)^\gamma + f_n = 0, \qquad \mathbf{K} = \frac{T}{4Z_0} + \frac{T^2}{4m_h},$$

State update

$$\boxed{\begin{bmatrix} d_n \\ v_{h,n} \end{bmatrix} = \begin{bmatrix} p_n + (\frac{T}{4Z_0} + \frac{T^2}{4m_h}) f_n \\ v_{h,n-1} + \frac{T}{2m_h}(f_{n-1} + f_n) \end{bmatrix}}$$

Junction velocity and coupling to waveguides

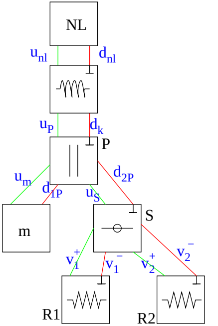
$$v = v_h - \dot{d} = v_h - (v_h + \frac{1}{2Z_0} - v_w) = v_w - \frac{1}{2Z_0} f, \qquad v_w = v_1^+ - v_2^-$$

$$\boxed{\mathbf{y_n} = \begin{bmatrix} v_1^- \\ v_2^+ \end{bmatrix} = \begin{bmatrix} -v + v_1^+ \\ v + v_2^- \end{bmatrix} = \begin{bmatrix} v_2^- + \frac{1}{2Z_0} f_n \\ v_1^+ - \frac{1}{2Z_0} f_n \end{bmatrix}}$$

WDF Nonlinear Piano Hammer (Pedersini, et al. 1998)

Lumped mechanical system is modeled as an equivalent circuit.

- Draw mass/spring/waveguide system in terms of equivalent circuits
- Waveguides look like resistors to the lumped hammer. Waves enter lumped junction directly.
- WDF result in tree like structures with adaptors/scattering junctions at the nodes, and elements at the leaves.
- The root of the tree allowed to have instantaneous reflections
- Nonlinearity gives instantaneous reflection, WDF handles only 1 nonlinearity naturally.



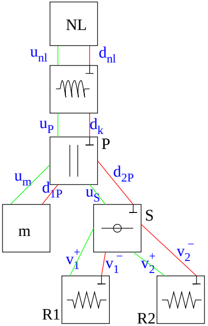
To left and right waveguides

WDF Computational Operations

Like standard filters, WDFs are computed by explicit multiply accumulates.

v_1^+, v_2^- from waveguides

$$u_S = -(v_1^+ + v_2^-)$$
$$u_m = d_{1P}[n-1]$$
$$u_P = \gamma_{1P} u_m + \gamma_{2P} u_S$$
$$u_{nl} = u_P + (u_P[n-1] - d_{nl}[n-1])$$
$$d_{nl}: \text{ solve } \left\{ \frac{u_{nl}+d_{nl}}{2} = k \left(\frac{u_{nl}-d_{nl}}{2R_{nl}} \right)^{\gamma} \right\}$$
$$d_k = d_{nl} + (u_P[n-1] - d_{nl}[n-1])$$
$$d_{1P} = u_P + d_k - u_m$$
$$d_{2P} = u_P + d_k - u_S$$
$$v_1^- = v_1^+ - \gamma_{1S}(d_{2P} - u_S)$$
$$v_2^+ = v_2^- - \gamma_{2S}(d_{2P} - u_S)$$



To left and right waveguides

Robustness/Stability

in the presence of coefficient quantization.

- WDF is stable and insensitive to variations in coefficients.
  - Direct form with second order section biquads are also robust, but transfer function representation abstracts relationship between component and filter state.
  - WDF provides direct one-to-one mapping from physical component to filter state variables.
  - Component-wise discretization facilitates verification that each component is passive.
- K-Method is more difficult to analyze
  - Find eigenvalues of state transistion matrix
  - Include contribution from Jacobian of nonlinear part.
  - More difficult to ensure that quantized coefficients will also result in a stable system.
  - Nonlinearity is no longer a function of a physical variable.

Parallelizability of K-Method

Explicit matrix operations are parallelizable.

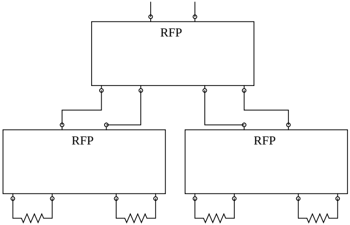
$$p_n = DH(\alpha I + A)x_{n-1} + (DHB + E)u_n + DHBu_{n-1} + DHCv_{n-1}$$
$$v_n = g(p_n)$$
$$x_n = H(\alpha I + A)x_{n-1} + HB(u_n + u_{n-1}) + HC(v_n + v_{n-1})$$

- Matrix multiplies are characterized by Multiply Accumulates (MACs).
- Explicit process can be characterized by dataflow.

Parallelizability of WDF

WDF connection tree indicates data dependency.

- Data flows from leaves to root and then from root to leaves because of reflection free ports always point to the parent.
- Nonlinearity at root is made explicit.
- Computations are independent between different branches.
- Each level of tree is parallel.

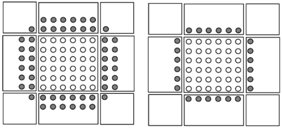


2-D Membrane/Plate PDE

FPGA Implementation for Real-Time Synthesis (Motuk et al. 2007)

Plate PDE: $\frac{\partial^2 u}{\partial t^2} = -\kappa^2\nabla^4 u + c^2\nabla^2 u - 2\sigma\frac{\partial u}{\partial t} + b_1\frac{\partial}{\partial t}\nabla^2 u + f(x,y,t)$

FDS: $u_{i,j}^{n+1} = \eta\sum_{|k|+|l|\le 2}\beta_{|k|,|l|}u^n_{i+k,j+l} + \eta\sum_{|k|+|l|\le 1}\gamma_{|k|,|l|}u^{n-1}_{i+k,j+l} + \Delta t^2 f^n_{i,j}$



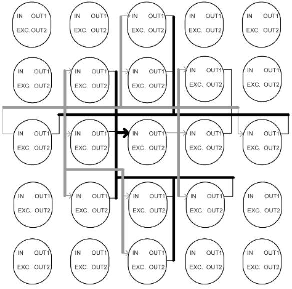
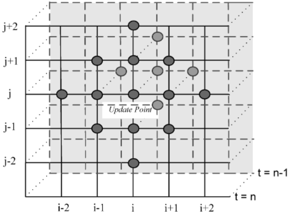


Fig. 5. Dataflow representation of a general FD algorithm for a domain of size 5 × 5.



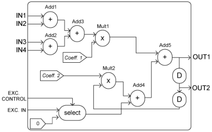


Fig. 6. Dataflow representation corresponding to a node in Fig. 5.

PERFORMANCE RESULTS OF THE IMPLEMENTATIONS

PE network	$N_{comp.}$	$N_{comm.}$	$N_{over.}$	N_{total}	f_{update} (kHz)	t_{1s} (s)	Pent. (s)	C6415 (s)
I (25 by 25)	2800	144	36	2980	60.4	0.73	32	5.97
II (10 by 100)	3000	600	67	3667	49.1	0.89		

E. Motuk, et al. "Design Methodology for Real-Time FPGA-Based Sound Synthesis," IEEE Trans. Sig. Proc., 55(12), pp. 5833–5845, Dec. 2007.

Summary

- Applications of nonlinear differential equations are abundant in musical acoustics.
- Even nonlinear differential equations can be made explicit if solution exists.
- Future work should consider hardware aspects and parallelism.
- Likewise, applications inform the design of hardware.
- Powerful hardware and physically accurate models can enable new and expressive electronic instruments.

Classical Network Theory

Element-wise discretization for digital computation.

- For example, use Bilinear transform $s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$
- Capacitor: $Z(s) = \frac{1}{sC}$

$$Z(z^{-1}) = \frac{T}{2C}\frac{1+z^{-1}}{1-z^{-1}} = \frac{V(z^{-1})}{I(z^{-1})}$$

$$v[n] = \frac{T}{2C}(i[n] + i[n-1]) + v[n-1]$$

- v[n] depends instantaneously on i[n] with $R_0 = \frac{T}{2C}$
- This causes problems when trying to make a signal processing algorithm
- Can also solve for solution using a matrix inverse (what SPICE does).

WDF Adaptors
Dependent ports

- Adaptors have property of low coefficient sensitivity, e.g., coefficients can be rounded or quantized with little change in passband frequency response. (Fettweis 74)
- Adaptors are also lossless and passive:
  total (pseudo-)energy in = total (pseudo-)energy out.
- Making a port dependent takes advantage of property that coefficients sum to two.
- Use this fact when quantizing coefficients to ensure that adaptor implementation remains (pseudo-)passive.