NEEDS Simulation-ready Compact Models

Tianshi Wang
A. Gokcen Mahmutoglu
Archit Gupta
Jaijeet Roychowdhury

EECS Department, University of California, Berkeley
Compact Models for Simulation

memristor
(RRAM, CBRAM, PCM...)

Applications:
- computation
- communication
- energy systems
Compact Models for Simulation

```
1 int isON = 0;
2 if (abs(V(...)) > V_snap)
3     isON = 1;
4 if (isON) {
5     ...
6 } else {
7     ...
8 }
```

```
1 real i;
2 if (V(br) < -1)
3     i = -1;
4 if (V(br) > +1)
5     i = +1;
6 I(br) <+ i;
7 end
```

```
$bound_step(tstep);
c_time = $abstime;
dt = c_time - p_time;
x = x_last + dt * exp(...);
```

```
$rdist_normal(rand_seed, 0);
```

```
@(initial_step) begin
    x = x_init;
end
```

"memory state"
"hidden state"

not an analog compact model

only for TRAN
none works for DC, AC, PSS

Boolean variable
"hybrid model"

T. Wang, UC Berkeley

Slide 3
Good Compact Models

• “simulation-ready”
  – run in all analyses (DC/AC/TRAN/sensitivity/shooting/HB/…)
  – run in all simulators consistently

• a simple (trivial) example

```
I(p, n) <+ V(p, n)/R;
I(p, n) <+ ddt(C * V(p, n));
...
```

- differential equation format

\[ \frac{d}{dt} q(v_{pn}) + f(v_{pn}) = \frac{d}{dt}(C \cdot V) \]

“charges” and “currents”, continuous and smooth
- no \$abstime, idt(), @initial_step, @cross, $bound_step, $rdist_normal etc.
- well-posed

Mahmutoglu/Wang/Gupta/Roychowdhury (2017) "Well-Posed Device Models for Electrical Circuit Simulation"
Good Compact Models

  – a solution exists
  – the solution is unique
  – the solution's behavior changes continuously with the initial conditions.
  
  *original definition applies to problems/analyses, not models*

• finite and unique outputs
  – 1/(x-a), log(x), sqrt(x), …
  – random number generation for noise and variability?

• continuous and smooth
  – \( C^\infty \): higher-order derivatives for PSS, distortion, homotopy

• input range
  – *should a model evaluate at 1000V?*

• higher-level requirements
  – well-understood physics, well-formulated (in DAE), well-tested
  – well-written in Verilog-A
Good Verilog-A Practices

- DO use branches.
- DO declare and initialize all variables and DO NOT use memory states.
- DO NOT use event control statements.
- DO NOT use analysis dependent functions.
- DO use ddt, but only in allowed ways.
- DO NOT use idt.
- DO NOT use time-varying functions.
- DO NOT use random number generators.
- DO take great care when using implicit equations.
- DO NOT allow any nodes in your model without having at least one branch with a well-defined contribution attached to it.
- DO NOT use bias-dependent switch branch and node collapse conditions.
- DO use parameter ranges.

Geoffrey Coram, “How to (and how not to) write a compact model in Verilog-A”
Case Study: Devices with Hysteresis

**Example:**

\[
\begin{align*}
\text{ipn} &= f(vpn) \\
\text{ipn} &= f_1(vpn, s) \\
\frac{d}{dt}s &= f_2(vpn, s)
\end{align*}
\]

\[
\begin{align*}
f_1(vpn, s) &= \frac{vpn}{R} \cdot (1 + \tanh(s)) \\
f_2(vpn, s) &= vpn - s^3 + s
\end{align*}
\]

internal state variable “memory”

hysteresis ≠ discontinuity or if-else

hysteresis ≠ $\text{abstime}$, ≠ hybrid models
Case Study: Devices with Hysteresis

**Template:**

\[
\text{ipn} = f_1 (\text{vpn}, \ s)
\]

\[
\frac{d}{dt} s = f_2 (\text{vpn}, \ s)
\]

**MAPP:**

\[
\text{ipn} = \frac{d}{dt} q_e (\text{vpn}, \ s) + f_e (\text{vpn}, \ s)
\]

\[
0 = \frac{d}{dt} q_i (\text{vpn}, \ s) + f_i (\text{vpn}, \ s)
\]

![Diagram of circuit and graphs showing hysteresis curves](image)
Case Study: Devices with Hysteresis

all DC sols from homotopy analysis (like a curve tracer)

---

e_1

---

\[ i_1 (mA) \]

---

\[ e_1 (V) \]

---

\[ s \]

---

2

---

1.5

---

1

---

0

---

-0.5

---

-1

---

-1.5

---

\[ e_1 (V) \]

---

\[ i_1 (mA) \]

---

1.5

---

1

---

0

---

-0.5

---

-1

---

-1.5

---

\[ e_1 (V) \]

---

\[ i_1 (mA) \]

---

1.5

---

1

---

0

---

-0.5

---

-1

---

-1.5

---

\[ s \]

---

2

---

1.5

---

1

---

0

---

-0.5

---

-1

---

-1.5

---

\[ e_1 (V) \]
Internal Unknowns in Verilog-A

\[ \frac{d}{dt}(\tau \cdot s) = \frac{vpn}{R} \cdot (1 + \tanh(s)) \]

\[ dpn = \frac{vpn}{R} \cdot (1 + \tanh(s)) \]

```
1 `include "disciplines.vams"
2 module hys(p, n);
3    inout p, n;
4    electrical p, n, ns;
5    branch (ns, n) ns_br1;
6    branch (ns, n) ns_br2;
7    parameter real R = 1e3 from (0:inf);
8    parameter real k = 1 from (0:inf);
9    parameter real tau = 1e-5 from (0:inf);
10   real s;
11
12   analog begin
13     s = V(ns, n);
14     I(p, n) <+ V(p, n)/R * (1+tanh(k*s));
15     I(ns_br1) <+ V(p, n) - pow(s, 3) + s;
16     I(ns_br2) <+ ddt(-tau*s);
17   end
18 endmodule
```
ESD Snapback Model


\[ I_{on} = G_{on} \cdot (V - V_H) \]

\[ I_{off} = I_S \cdot (1 - e^{-V/\phi_T}) \cdot \sqrt{1 + \frac{\max(V, 0)}{V_D}} \]

\[ I = s \cdot I_{on} + I_{off} \]

internal state: indicator of impact ionization

\[ \frac{d}{dt} s = f(V, s) \]

T. Wang, UC Berkeley
ESD Snapback Model

\[ \frac{d}{dt}s = f(V, s) \]

many possible functions

\[
V_{\text{star}} = 2 \times (V(p, n) - 0.5 \times V_T1 - 0.5 \times V_{IH})/(V_T1 - V_{IH});
\]

\[
s_{\text{star}} = 2 \times (s - 0.5);
\]

\[
I(ns, n) \leftarrow \tanh(K \times (V_{\text{star}} + s_{\text{star}})) - s_{\text{star}};
\]

... shift transition points
... shift range to (0, 1)
ESD Snapback Model

```
`include "disciplines.vams"

module ESDclamp(p, n);
  inout p, n;
  electrical p, n, ns;

  parameter real Gon = 0.1 from (0:inf);
...

  analog begin
    s = V(ns, n);
    Ion = smoothclip(Gon*(V(p, n)-VH), smoothing)
        - smoothclip(-Gon*VH, smoothing);
    Ioff = Is * (1 - limexp(-V(p, n)/VT))
        * sqrt(1 + max(V(p, n), 0)/VD);
    I(p, n) <+ Ioff + pow(s, Alpha) * Ion;
    I(p, n) <+ ddt(C * V(p, n));

    Vstar = 2*(V(p, n)-0.5*VT1-0.5*VIH)/(VT1-VIH);
    sstar = 2*(s-0.5);
    I(ns, n) <+ tanh(K*(Vstar + sstar)) - sstar;
    I(ns, n) <+ ddt(-tau*s);

  end
endmodule
```
ESD Snapback Model

forward/backward DC sweeps

homotopy (all DC sols)

transient voltage sweeps

“impact ionization doesn't happen instantaneously (although all compact models assume that it does)” — C.C. McAndrew

Human Body Mode (HBM) test

+2000V

1Meg

1.5k

100p

V

ESD clamp
ESD Snapback Model

![Graphs showing ESD snapback model for MM and CDM.
Simple TLP equivalent circuit diagram with R1=50.
Measurements and simulation curves for V-I characteristics.
]
RRAM Model

Template:

\[ i_{pn} = f_1(v_{pn}, s) \]
\[ \frac{d}{dt}s = f_2(v_{pn}, s) \]

RRAM:

\[ f_1(v_{pn}, \text{Gap}) = I_0 \cdot e^{-\text{Gap}/g_0} \cdot \sinh(v_{pn}/V_0) \]
\[ f_2(v_{pn}, \text{Gap}) = -v_0 \cdot \exp\left(-\frac{E_a}{V_T}\right) \cdot \sinh\left(\frac{v_{pn} \cdot \gamma \cdot a_0}{t_{ox} \cdot V_T}\right) \]


minGap \leq \text{Gap} \leq \text{maxGap}

if \text{gap} < \text{minGap}
\[ \text{gap} = \text{minGap}; \]

hybrid model
RRAM Model

Template:

\[
\text{ipn} = f_1(vpn, s) \quad f_1(vpn, \text{Gap}) = I_0 \cdot e^{-\text{Gap}/g_0} \cdot \sinh(vpn/V_0)
\]

\[
\frac{d}{dt}s = f_2(vpn, s) \quad f_2(vpn, \text{Gap}) = -v_0 \cdot \exp\left(-\frac{E_a}{V_T}\right) \cdot \sinh\left(\frac{vpn \cdot \gamma \cdot a_0}{t_{ox} \cdot V_T}\right)
\]

\[
\times F_{\text{window}}(\text{Gap})
\]

Biolek, Jogelkar, Prodromakis, UMich, TEAM/VTEAM, Yakopcic, etc.

---

T. Wang, UC Berkeley
RRAM Model

Analogy: MEMS switch
Zener diode voltage regulator

clipping functions
Memristor Models

\[ \frac{d}{dt} s = f_2 (v_{pn}, s) \]

Available \( f_2 \):

1. linear ion drift
   \[ f_2 = \mu_i \cdot R_{on} \cdot f_1 (v_{pn}, s) \]

2. nonlinear ion drift
   \[ f_2 = a \cdot v_{pn}^m \]

3. Simmons tunnelling barrier
   \[ f_2 = \begin{cases} c_{off} \cdot \sinh \left( \frac{s}{i_{off}} \right) \cdot \exp \left( - \exp \left( \frac{s}{w_c} - \frac{s}{b} \right) - \frac{s}{w_c} \right), & \text{if } i \geq 0 \\ c_{on} \cdot \sinh \left( \frac{s}{i_{on}} \right) \cdot \exp \left( - \exp \left( \frac{s}{w_c} + \frac{s}{b} \right) - \frac{s}{w_c} \right), & \text{otherwise} \end{cases} \]

4. TEAM model

5. Yakopcic model

6. Stanford/ASU
   \[ f_2 = -v_0 \cdot \exp \left( -\frac{E_a}{V_T} \right) \cdot \sinh \left( \frac{v_{pn} \cdot \gamma \cdot a_0}{t_{ox} \cdot V_T} \right) \]

\[ ipn = f_1 (v_{pn}, s) \]

Available \( f_1 \):

1. \[ f_1 = \left( R_{on} \cdot s + R_{off} \cdot (1 - s) \right)^{-1} \cdot v_{pn} \]

2. \[ f_1 = \frac{1}{R_{on}} \cdot e^{-\lambda \cdot (1 - s)} \cdot v_{pn} \]

3. \[ f_1 = s^n \cdot \beta \cdot \sinh (\alpha \cdot v_{pn}) + \chi \cdot (\exp (\gamma \cdot s) - 1) \]

4. \[ f_1 = \begin{cases} A_1 \cdot s \cdot \sinh (B \cdot v_{pn}), & \text{if } v_{pn} \geq 0 \\ A_2 \cdot s \cdot \sinh (B \cdot v_{pn}), & \text{otherwise} \end{cases} \]

5. \[ f_1 = I_0 \cdot e^{-\text{Gap}/g^0} \cdot \sinh \left( \frac{v_{pn}}{V_0} \right) \]
   \[ \text{Gap} = s \cdot \min \text{Gap} + (1 - s) \cdot \max \text{Gap} \]

- set up boundary
- fix \( f_2 \) flat regions
- smooth, safe funcs, scaling, etc.
Memristor Models

A collection of models:
- all smooth, all well posed
- not just RRAM, but general memristive devices
- not just bipolar, but unipolar
- not just DC, AC, TRAN, but homotopy, PSS, ...

PSS using HB
Good Compact Models

**Well-posedness:**
- finite and unique outputs
- continuous and smooth
- input range
- physics, DAE, tests, Verilog-A …

**Good Verilog-A practices**

- Geoffrey Coram, “How to (and how not to) write a compact model in Verilog-A”

**Case study with hysteretic devices**

- ESD snapback
- RRAM/memristors