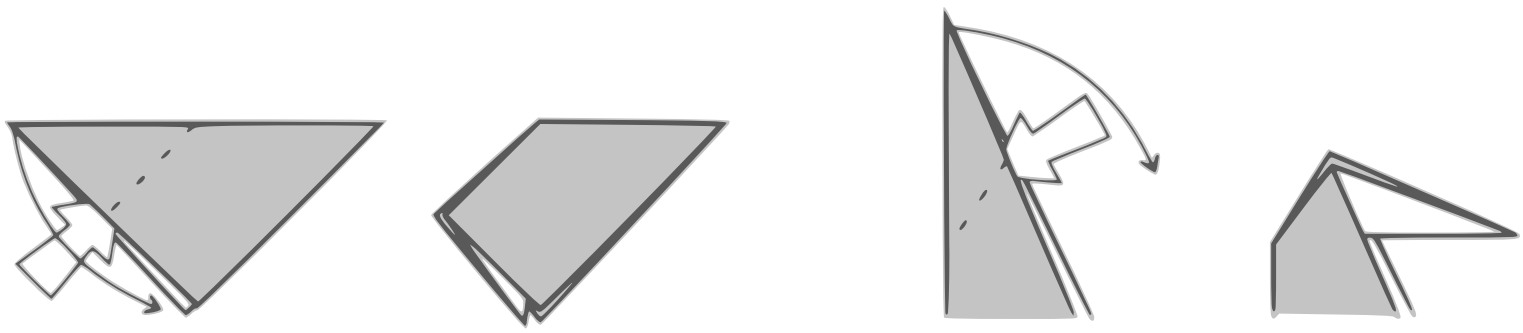
Integrating Circuit Simulation with EIT FEM Models

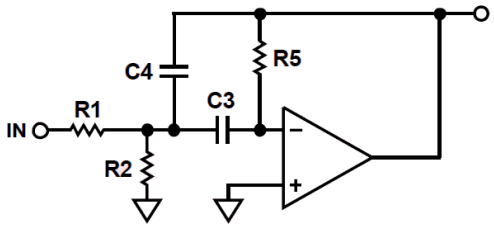
Alistair Boyle and Andy Adler

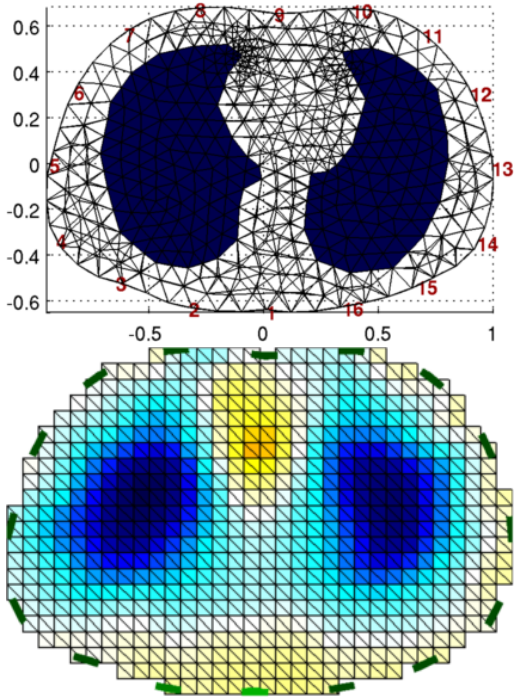
University of Ottawa Carleton University
Ottawa, Canada

EIT2018, June 11–13, 2018



Circuits and FEM





Step 1: forward solution

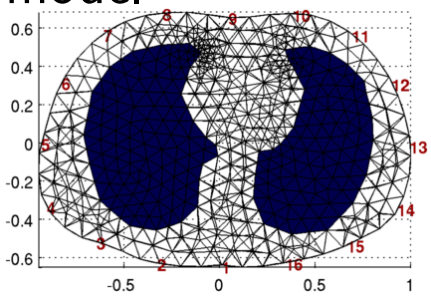
improve hardware designs with accurate simulations

Step later: inverse solution

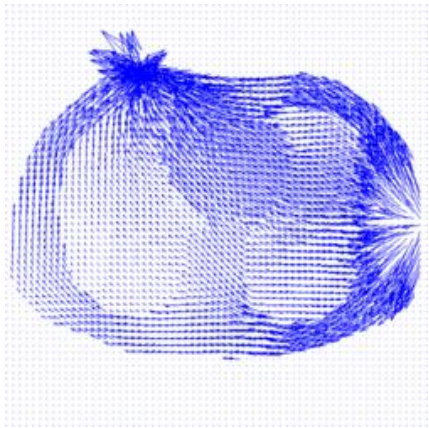
handle hardware imperfections and aging

EXAMPLE: EIT

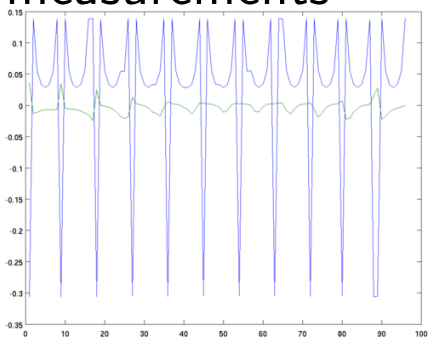
model



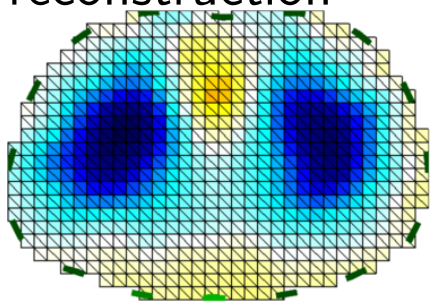
simulations



measurements



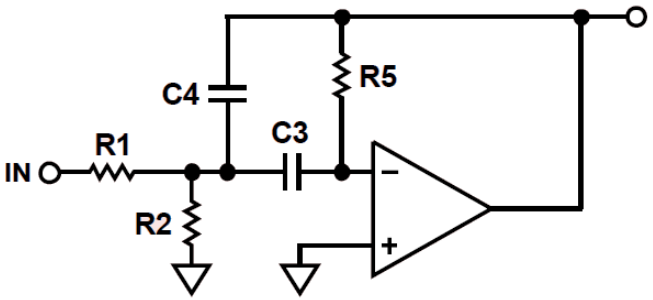
reconstruction



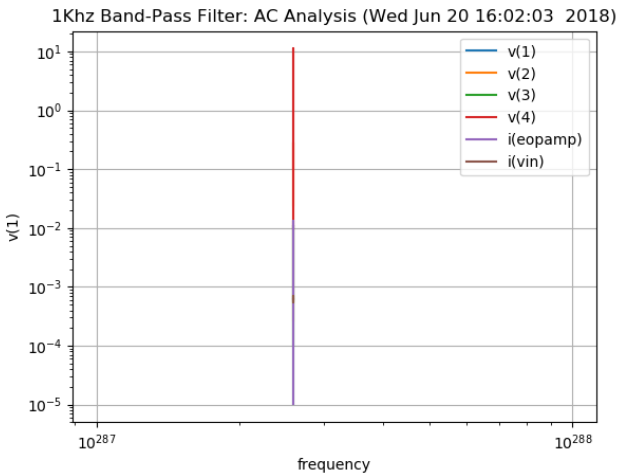
EXAMPLE: CIRCUIT SIMULATION (SPICE)

Non-linear circuit analysis (opamps, I/O models) and signal integrity (transmission lines)

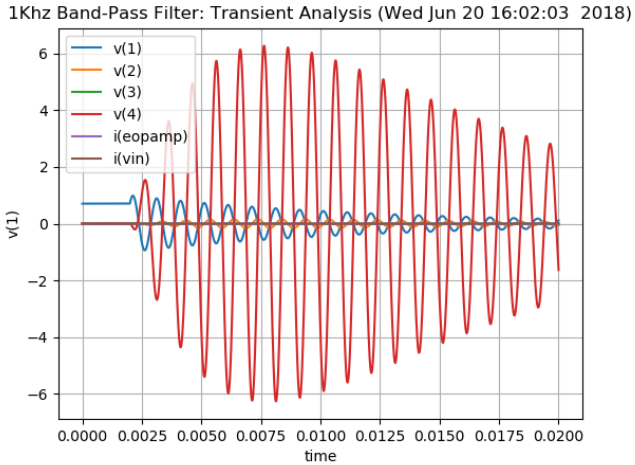
```
 1kHz Band-Pass Filter
Vin 1 0 DC 0 AC 1
R1 1 2 1.59k
R2 2 0 41
C3 2 3 100nF
C4 2 4 100nF
R5 4 3 1 64k
Eopamp 5 0 0 3 1000
```



frequency domain



time domain



and detailed non-linear semiconductor and IBIS device models

EIT + SPICE

1. EIT → spice
2. EIT ← spice

EIT + SPICE

1 EIT → spice

EIT $\rightarrow$ SPICE

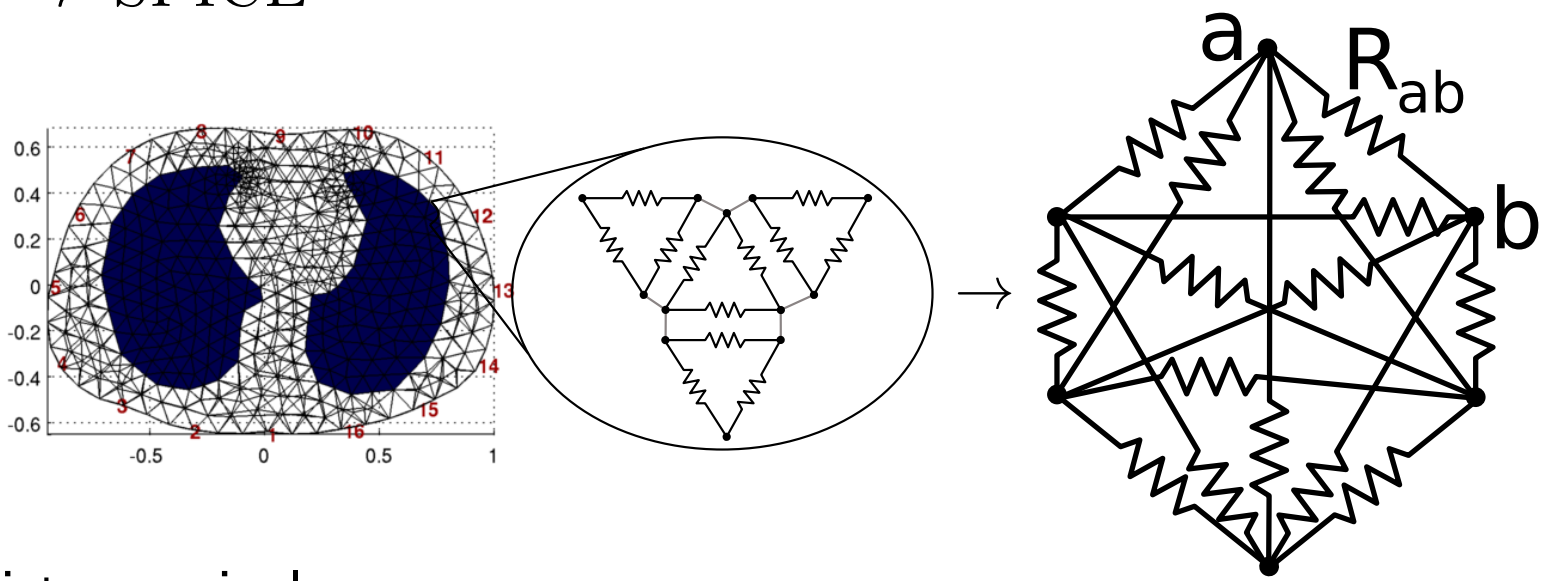
blockwise matrix inverse: a model reduction

$$\begin{bmatrix} \mathbf{V}_A \\ \mathbf{V}_D \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\mathsf{T} & \mathbf{D} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbf{X}_D \end{bmatrix} \tag{1}$$

$$\mathbf{D}' = \mathbf{D} - \mathbf{B}^\mathsf{T}\mathbf{A}^{-1}\mathbf{B} \tag{2}$$

$$\mathbf{V}_D = \mathbf{D}'^{-1}\mathbf{X}_D \tag{3}$$

EIT → SPICE



resistor equivalence:

$$\mathbf{D}' = \mathbf{D} - \mathbf{B}^{\mathsf{T}}\mathbf{A}^{-1}\mathbf{B}$$

- $\mathbf{D}'$ is an N-port full mesh resistor network
- $\mathbf{D}'$ off-diagonal values are $1/R_{ab}$ between electrode a and b
- delta-wye conversion for 2D models, but its not worth the trouble
- complex-valued extension: R to RLC network

General FEM → spice methods used in EMI and antenna analysis

EXAMPLE: EIT → SPICE (FORWARD SOLUTION)

SPICE input files:

file: stim.s

```
Test circuit for EIT
.inc eit.s
rgnd e1 0 1e-16
isrc e1 e2 dc 1A
Xeit e1 e2 e3 e4 e5 e6 e7 e8 eit
.control
print V(e1,e2) V(e2,e3) V(e3,e4)
+ V(e4,e5) V(e5,e6) V(e6,e7)
+ V(e7,e8) v(e8,e1)
.endc
.end
```

file: eit.s

```
* full-mesh R network of 8-electrode EIT
FEM
.subckt eit e1 e2 e3 e4 e5 e6 e7 e8
re01 e1 e2 3.9298245614035099
re02 e1 e3 11.200000000000005
re03 e1 e4 21.000000000000018
            [...]
re25 e5 e8 21.000000000000018
re26 e6 e7 1.5849056603773584
re27 e6 e8 11.200000000000008
re28 e7 e8 3.9298245614035099
.ends
```

EIDORS output:

```
-1.6239
0.4517
0.2500
0.0483
0.1239
0.0483
0.2500
0.4517
```

=

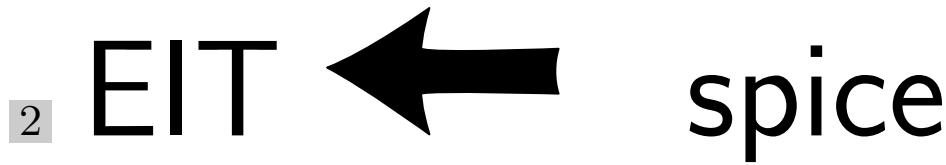
SPICE output:

Circuit: test circuit for eit

[...]

v(e1,e2) = -1.62395e+00
v(e2,e3) = 4.516807e-01
v(e3,e4) = 2.500000e-01
v(e4,e5) = 4.831933e-02
v(e5,e6) = 1.239496e-01
v(e6,e7) = 4.831933e-02
v(e7,e8) = 2.500000e-01
v(e8,e1) = 4.516807e-01

EIT + SPICE



SPICE → EIT

Input: SPICE netlist

Modified Nodal Analysis (MNA) circuit analysis ideal elements:

2 node elements

- "Rn" resistor
- "Cn" capacitor
- "Ln" inductor
- "Vn" voltage source
- "In" current source

4 node elements

- "En" voltage controlled voltage source (VCVS)
- "Fn" current controlled current source (CCCS)
- "Gn" voltage controlled current source (VCCS)
- "Hn" current controlled voltage source (CCVS)

ind: $\begin{bmatrix} -1/sL & 1/sL \\ 1/sL & -1/sL \end{bmatrix} \begin{bmatrix} V_+ \\ V_- \end{bmatrix}$

VCVS: $\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & 0 & 0 \\ G & -1 & 1 & -G & 0 \end{bmatrix} \begin{bmatrix} V_{i+} \\ V_{o+} \\ V_{o-} \\ V_{i-} \\ I_o \end{bmatrix}$

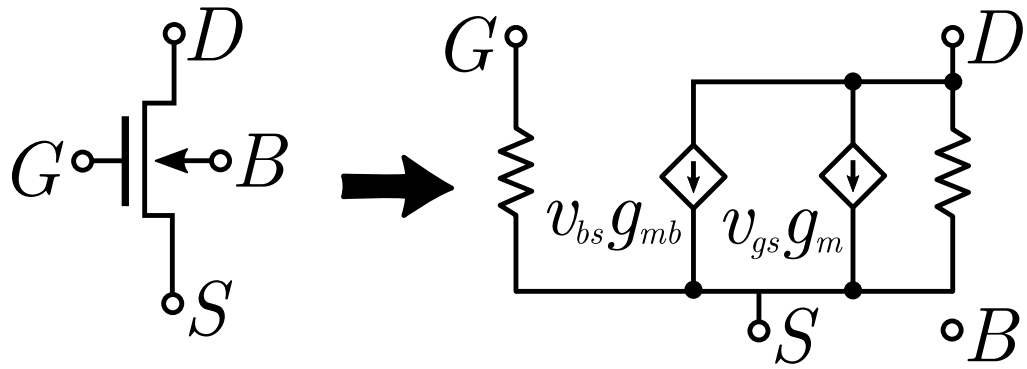
$s = \mathbf{j}\omega$

Grounded nodes are a special case

We use the FEM connectivity matrix to "stitch" nodes in FEM (electrodes) to circuit nodes

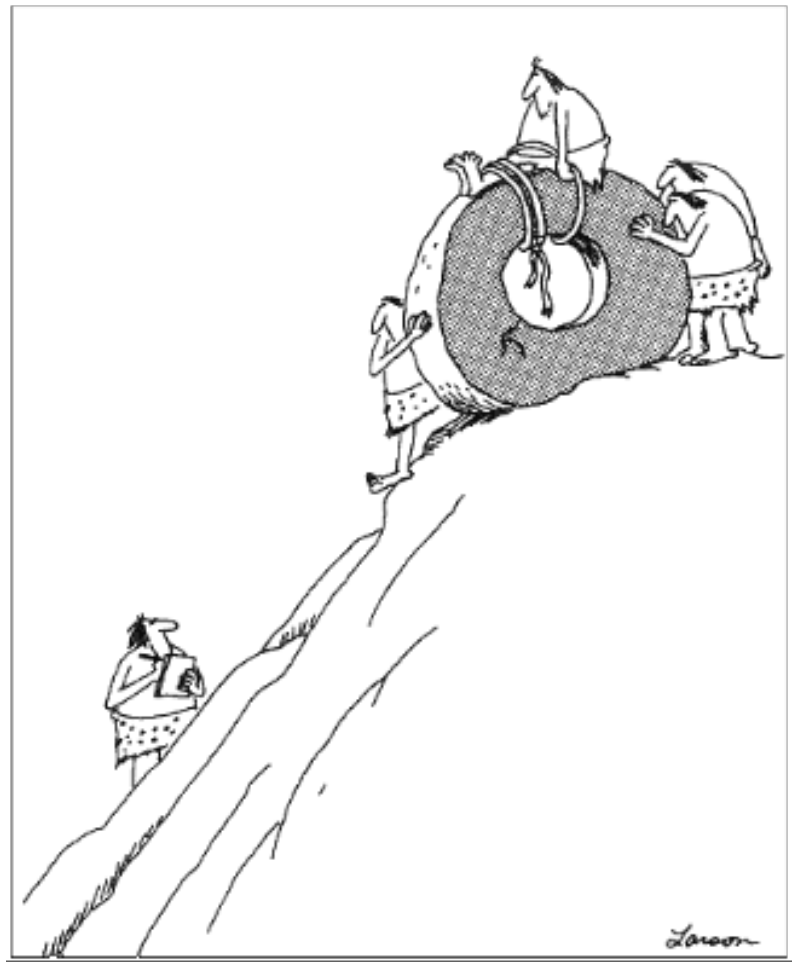
Step later: inverse problem

Nonlinear components (opamp, diode, transmission line, transistor) modelled by linearizing to ideal components



SPICE implementations provide algorithms to find operating points for small signal nonlinear models, including those that exhibit hystersis (ex. Schmitt triggers)

Challenges



Early experiments in transportation [1]

[1] ©Gary Larson, The Farside

Challenges

On the failure of the Cadence Spectre SPICE simulator to penetrate the market, despite being faster, more accurate, more robust:

> *[... we] completed the first version in two weeks [...] One can find serious faults in any* SPICE *simulator you try. Ironically, the faults are one of the things that make it hard to replace.* [2]

- HSpice *Synopsys*
- PSpice *Cadence*
- Spectre *Cadence*
- LTspice IV *Linear Technologies*
- TINA *Texas Instruments*
- 5Spice *Cadence*
- SuperSpice *AnaSoft*
- NI Multisim *National Instruments*
- TopSpice *Penzar*
- ngSpice *GPL*
- EveryCircuit *MuseMaze*
- Qucs *GPL*

[2] K. Kundert (2011) *The Life After Spice*, IEEE Solid-State Circuits Magazine

Possible directions

1. EIT → spice
   - support grounding current
   - complex-valued conductivities (RLC networks)
   - time/frequency-dependent models

2. EIT ← spice
   - SPICE library integration[a] (ngspice)
   - solve inverse problem for circuit parameters

[a] not with matlab

Available now at a code repository near you

1. EIT → spice
   - for real-valued conductivities
   - at a single-frequency
   - generates "standard" spice netlist (sub-circuit)
   - `eidors/solvers/forward/eit_spice.m`

2. EIT ← spice
   - parses "standard" spice netlist
   - for linear circuits
   - forward model[a]
   - `eidors/solvers/forward/spice_eit.m`

[a] MNA "stitching" not yet implemented