Compact Behavioural Modelling of Electromagnetic Effects in On-Chip Interconnect

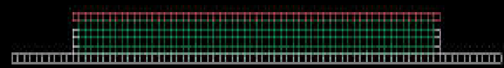
Invited presentation at MACSI-NET 2003, May 2-3, Zürich

Peter B.L. Meijer

Philips Research Laboratories
Peter.B.L.Meijer@philips.com

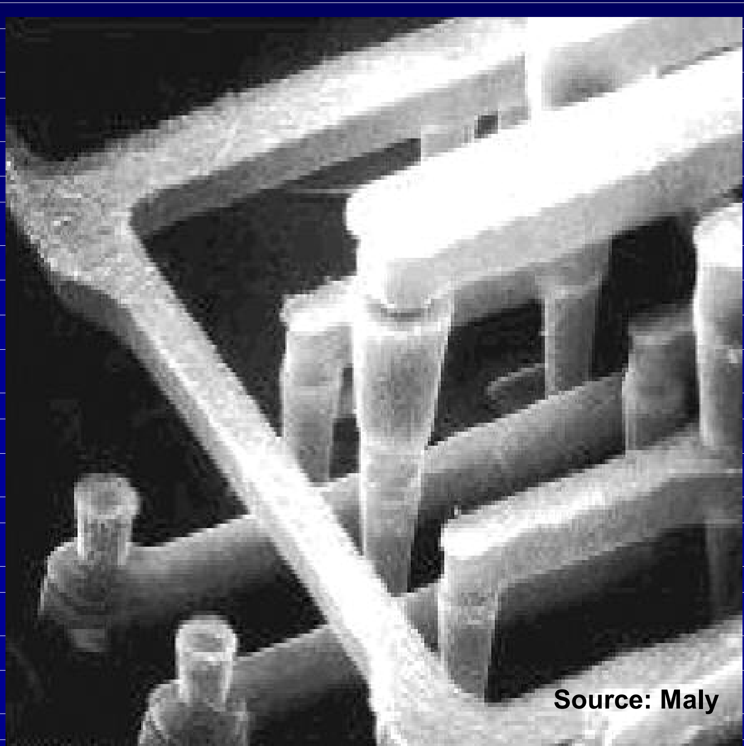
Contents

- Introduction
- Modelling "flow"
- From Maxwell's Equations to circuit simulation
  - Example 1: one wire, load modelling
  - Example 2: two wires with cross-talk



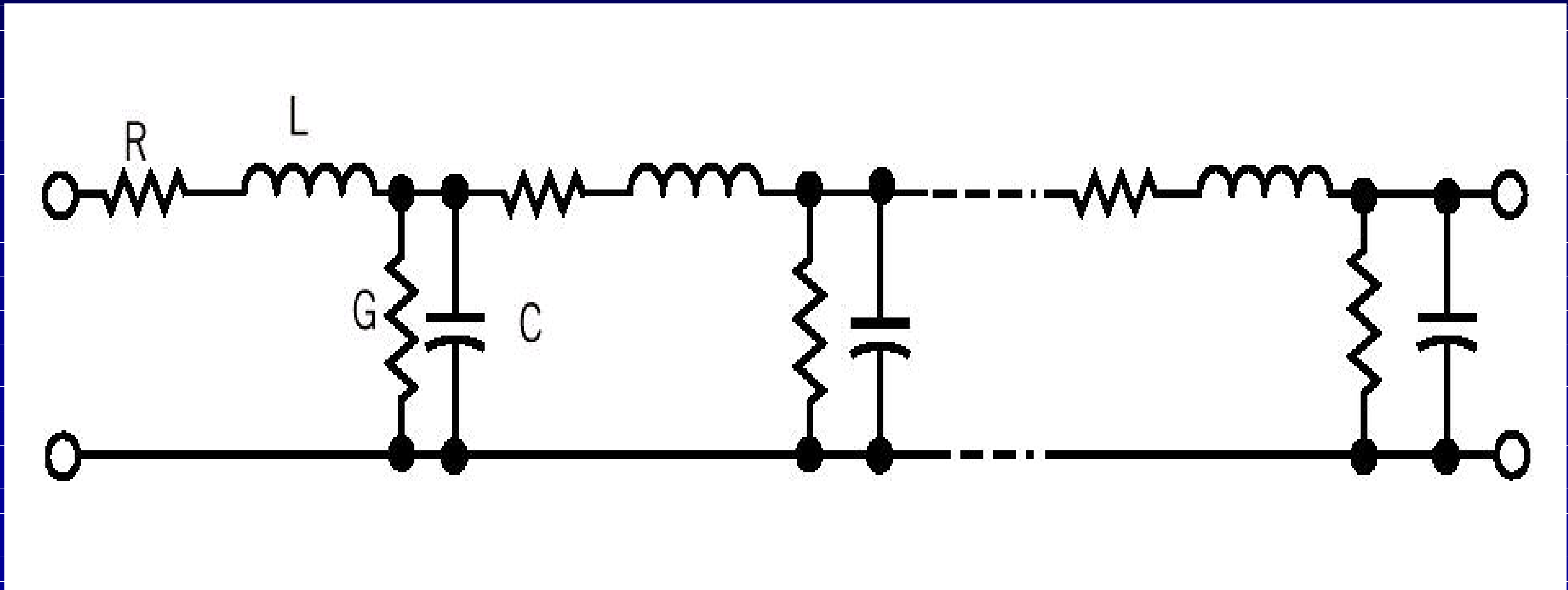
- Conclusions

Why?



- Find limiting factors in high-speed digital circuits
- Reference for validation of future design rules
- Reference for validation of alternative models

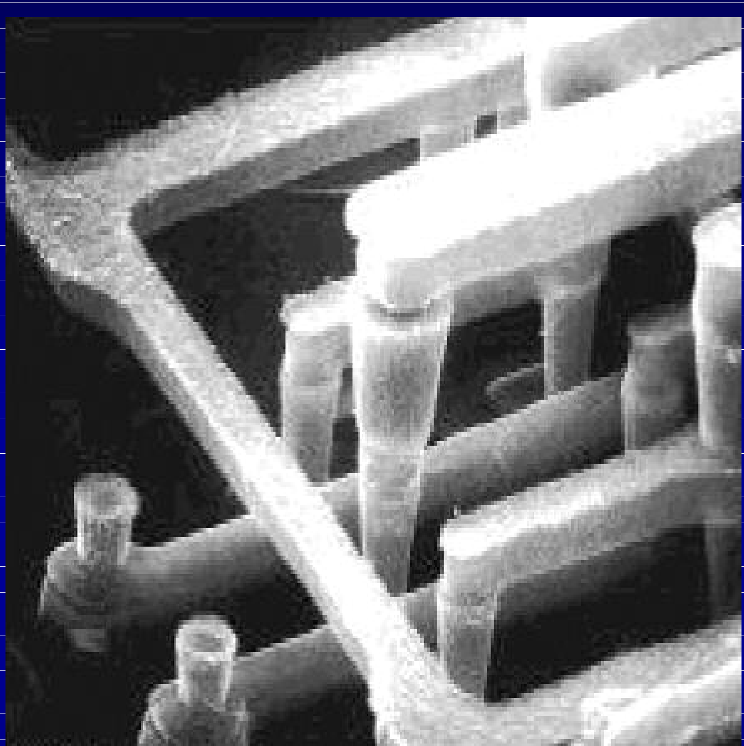
Other approaches



RLC lumped models
Transmission line models
ROM techniques
...

} Modelling assumptions?

Modelling flow?



Maxwell Equations

↓

``FDTD'' gives V(t), I(t)

↓

Fit a linear dynamic model

↓

Our generalized formalism

↓

Post-optimize & generate syntax for simulation model

↓

Circuit Simulation

FDTD method

FDTD = Finite Difference Time Domain: a method for solving the Maxwell Equations

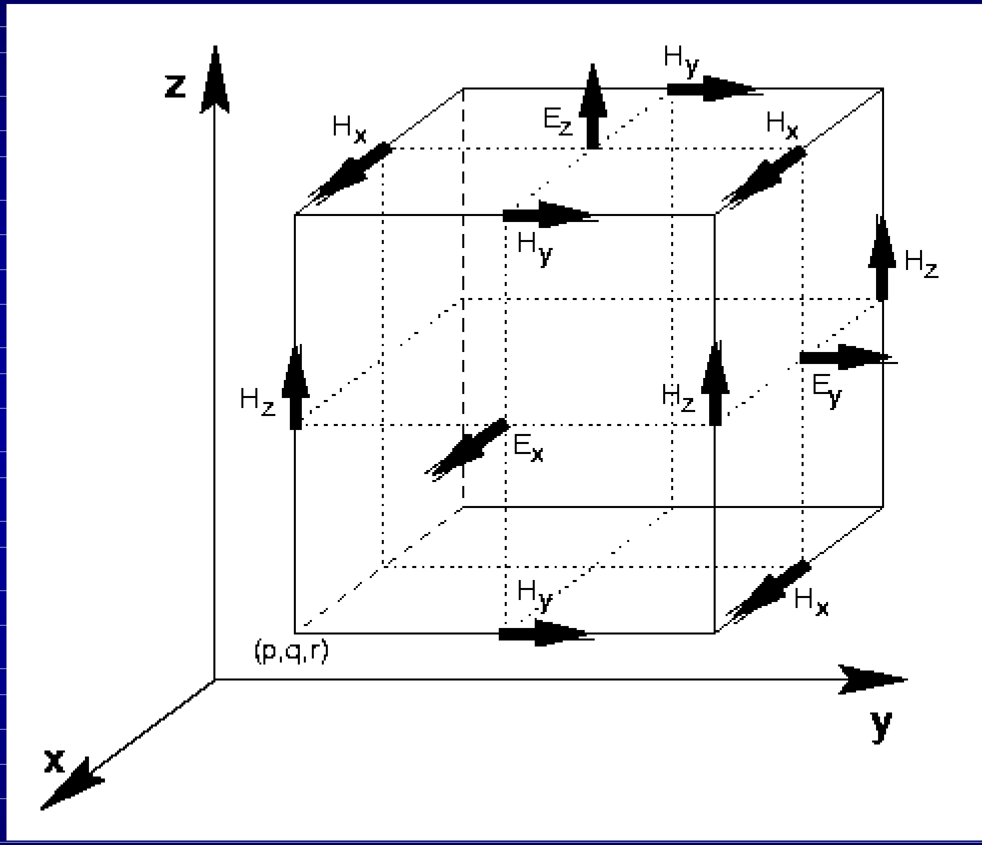
- **Discretizes and solves Maxwell's Equations in both space and time**

$$\nabla \cdot \vec{D} = \rho_f$$

$$\nabla \cdot \vec{B} = 0$$

$$\nabla \times \vec{E} = \frac{\partial \vec{B}}{\partial t}$$

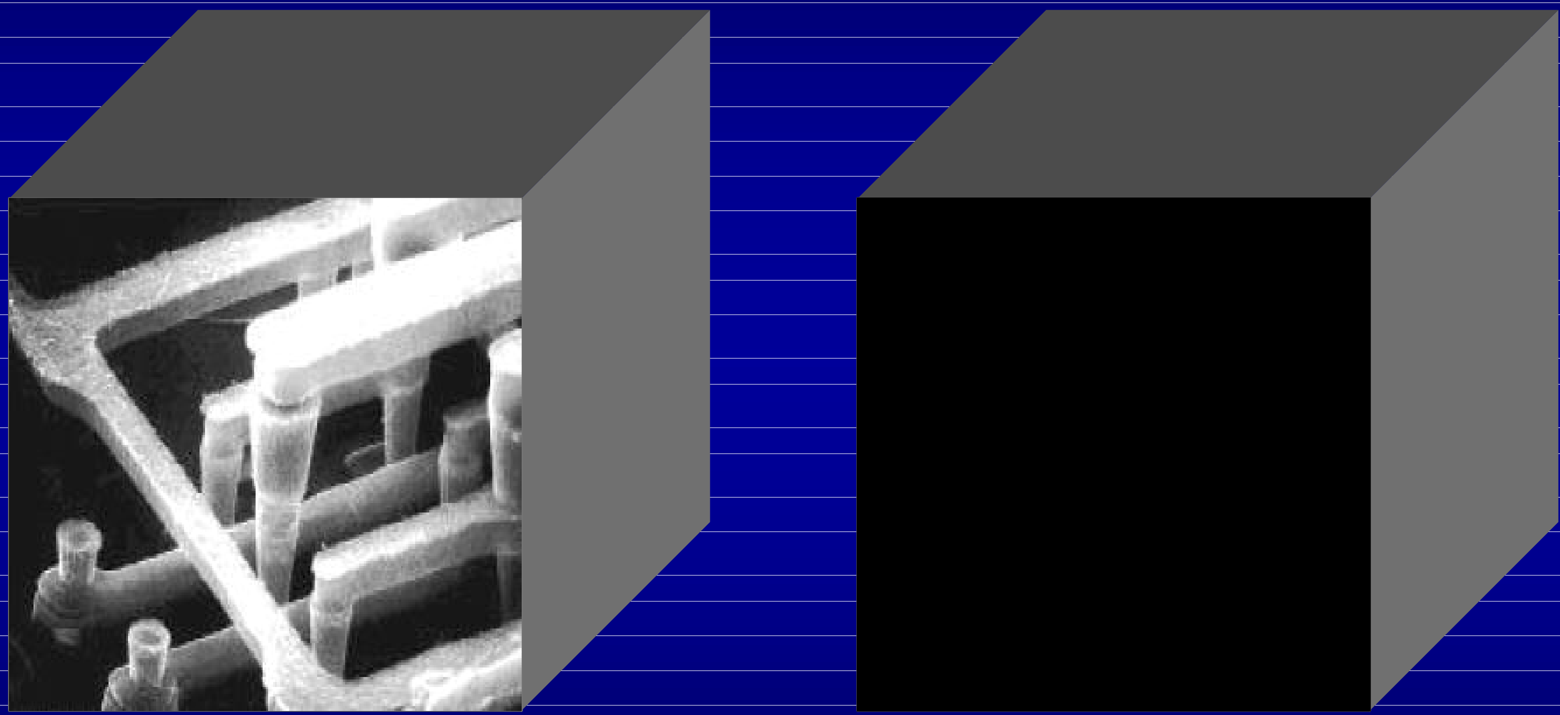
$$\nabla \times \vec{H} = \frac{\partial \vec{D}}{\partial t} + \vec{J}_f$$



- **CPU intensive!**

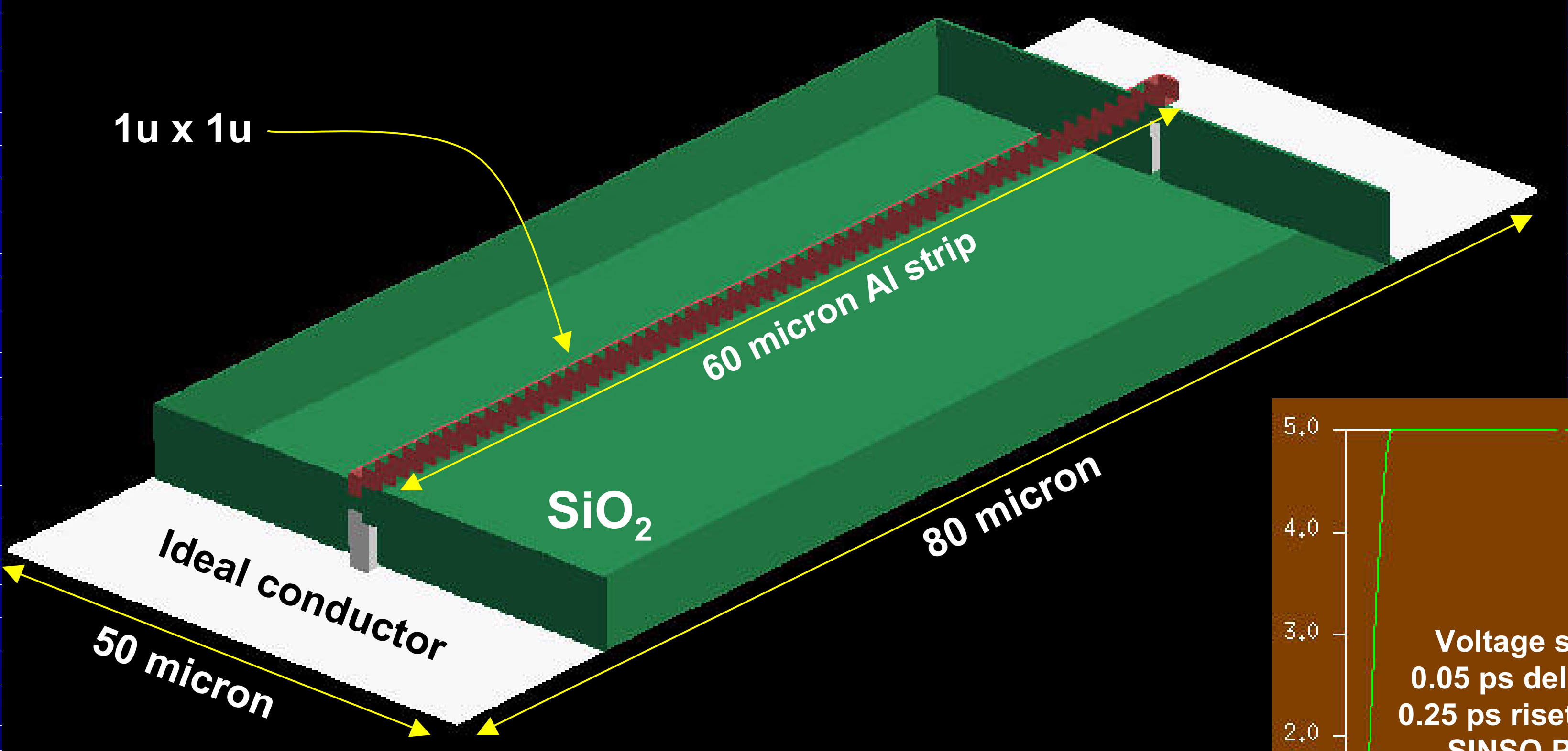
Also yields voltages and currents

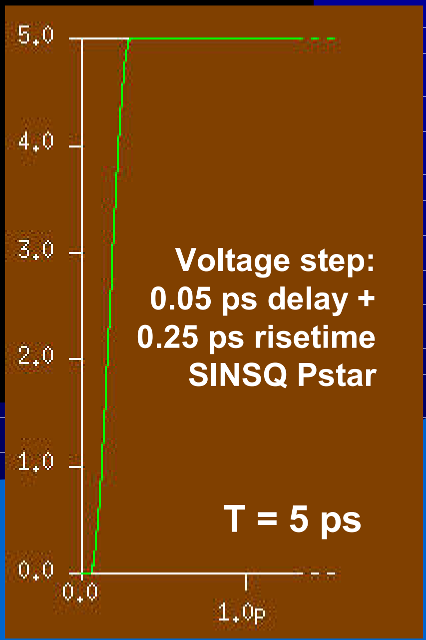
FDTD: CPU time = f(mesh)



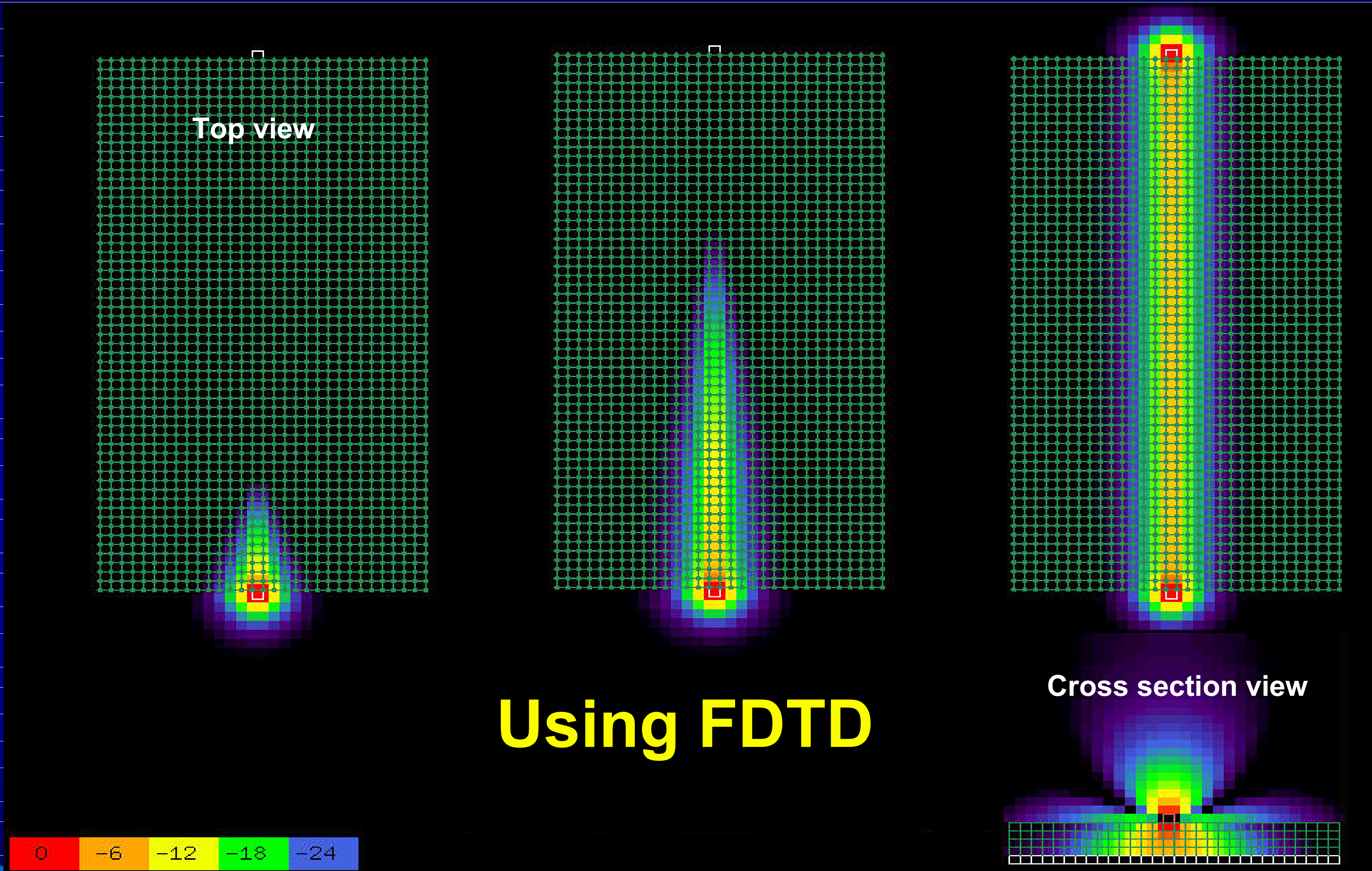
Using FDTD

Mesh: 250,000 cells @ 2500 time points



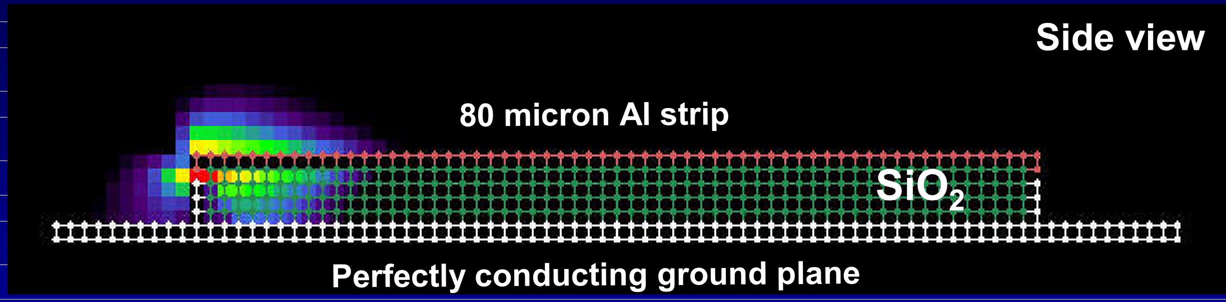


t = 0.2 picosecond | t = 0.4 picosecond | t = 4.6 picosecond

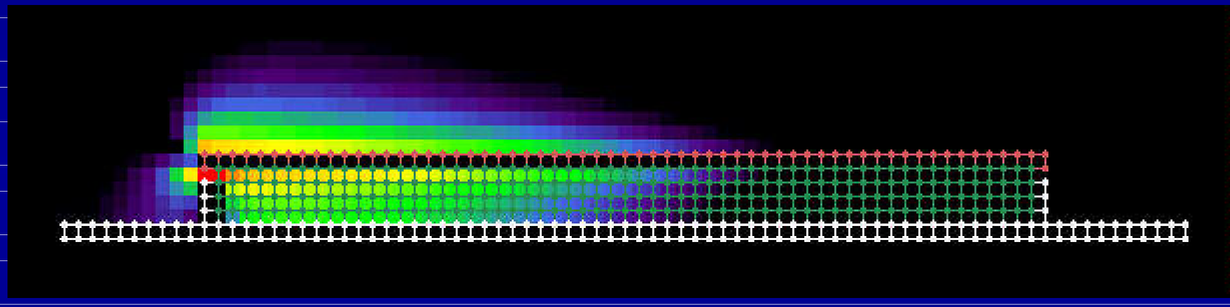


Using FDTD

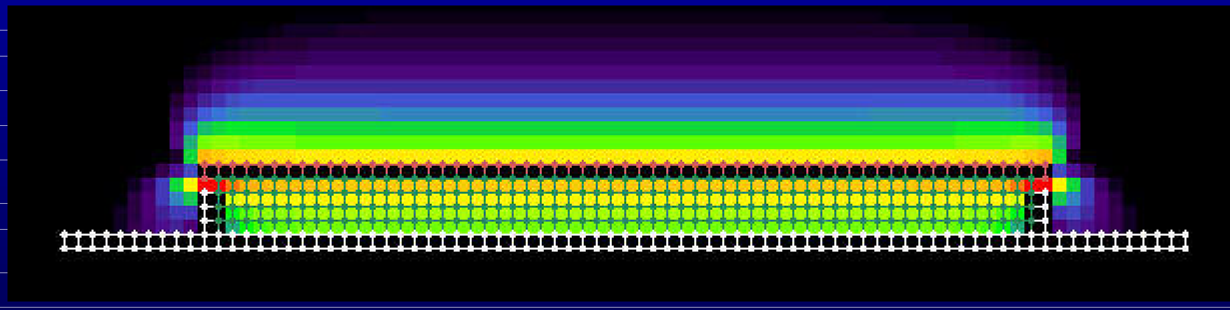
t = 0.2 picosecond



t = 0.4 picosecond

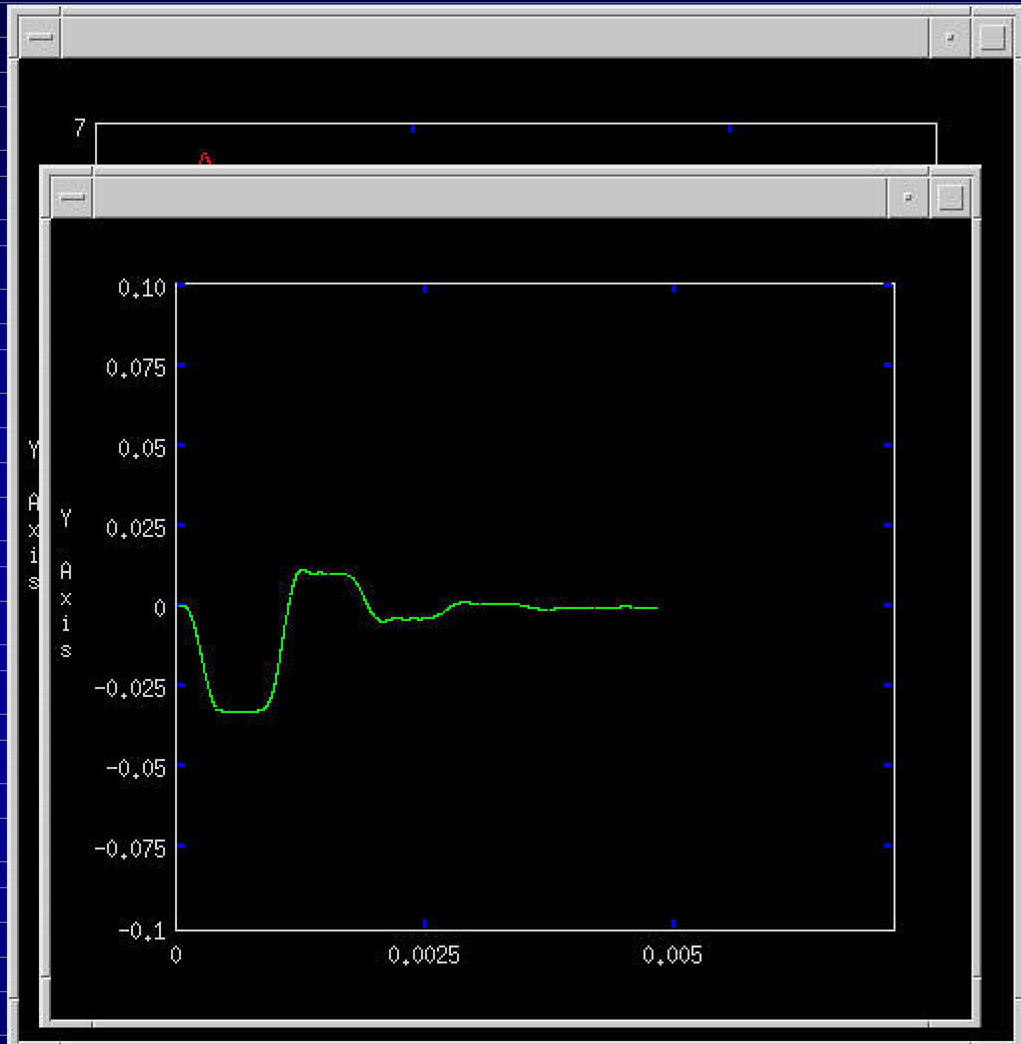


t = 4.6 picosecond



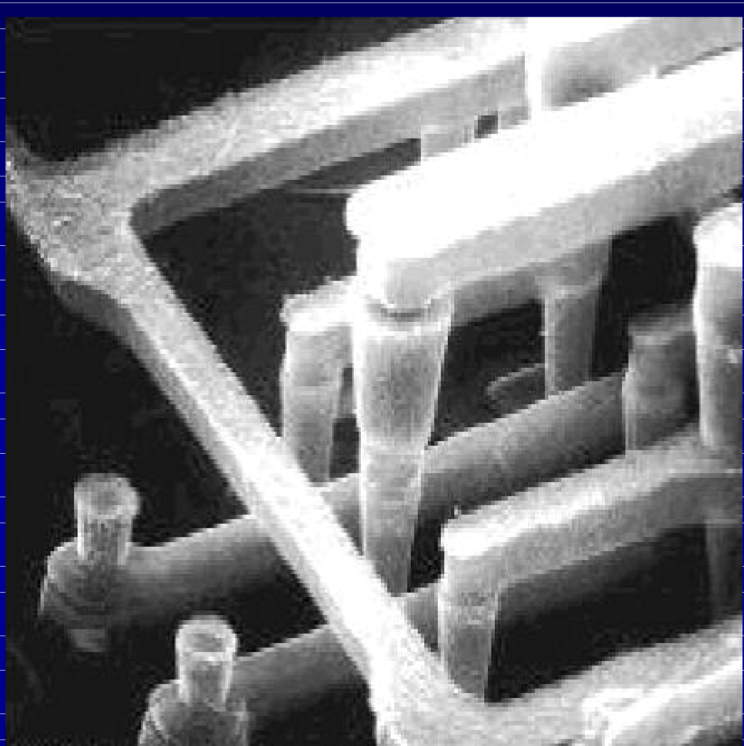
Using FDTD

- **500 GHz reflections**
  - Reflections give overshoot and affect wave shape at source side
  - Analyze beyond measurement options
  - Results sanity check: ~0.4 ps for 60 micron: 1.5 x 10^8 m/s ~= c / √3.9, (3.9 is rel. perm. SiO_2)



Current $I_s(t)$ at source side

Voltages $V_s(t)$ and $V_t(t)$



Maxwell Equations

FDTD gives V(t), I(t)

Circuit Simulation

Linear State Space Modelling

- Assume linear state space model:

  Matrix equations

  $$x'(t) = A\,x(t) + B\,u(t)$$
  $$y(t) = C\,x(t) + D\,u(t)$$

- Determine parameter matrices A, B, C, D for given input vectors u(t) and output vectors y(t) such that a best fit is obtained

- MOESP/4SID class of subspace identification algorithms

Linear State Space Modelling (MOESP/4SID)

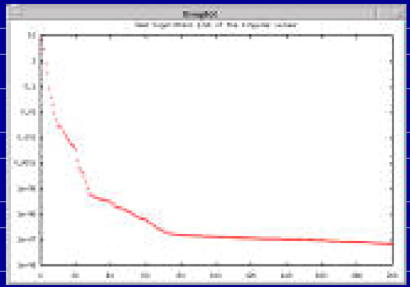
Behavioural modelling based on time domain data (waveforms)

DeWilde, Verhaegen, Ciggaar, Meijer, Schilders

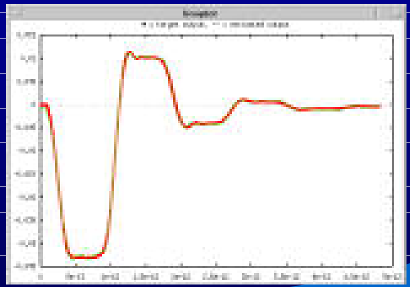
Steps:

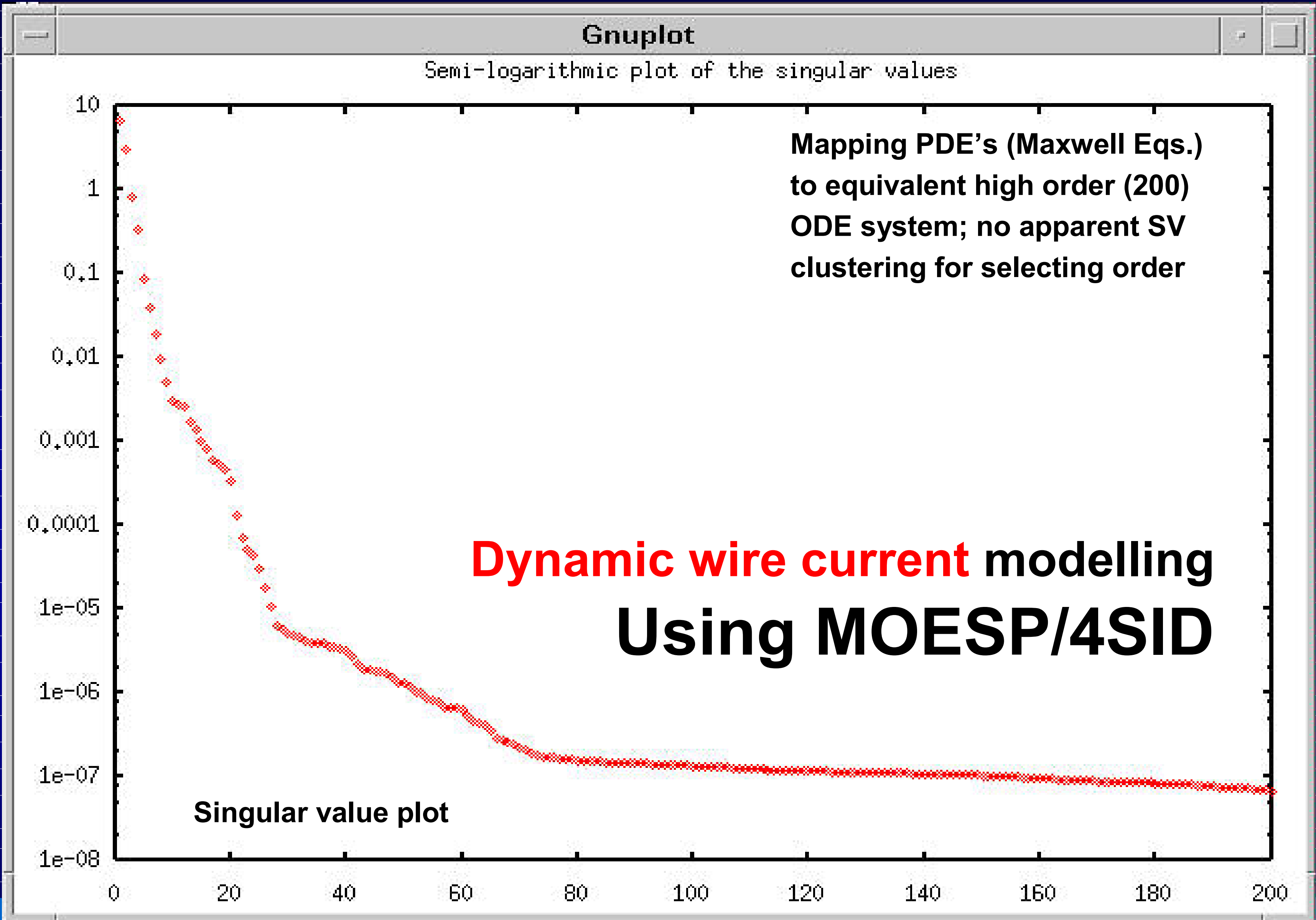
- Initial order sufficiently high, e.g., 200

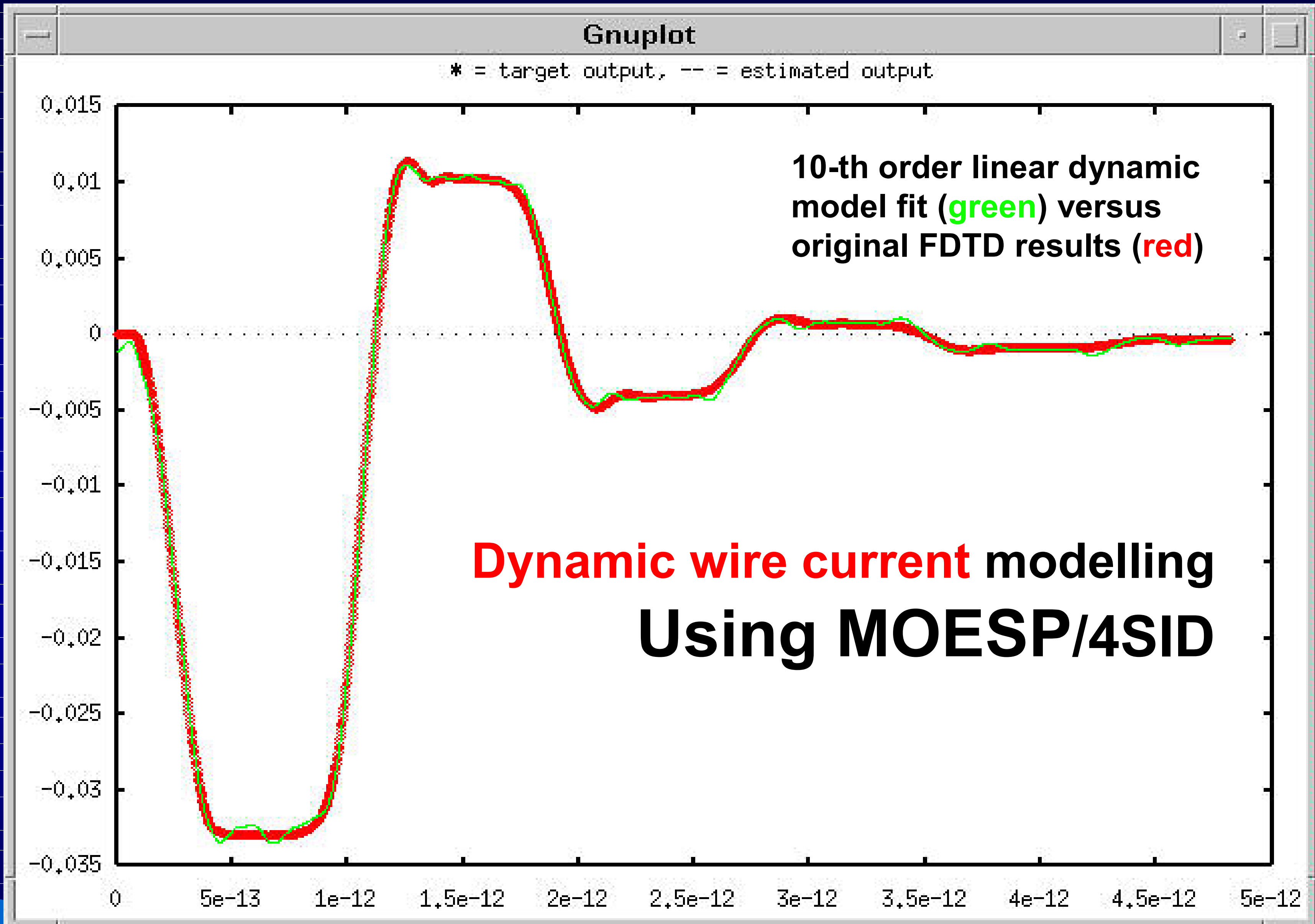
  SVD plot to estimate # time constants

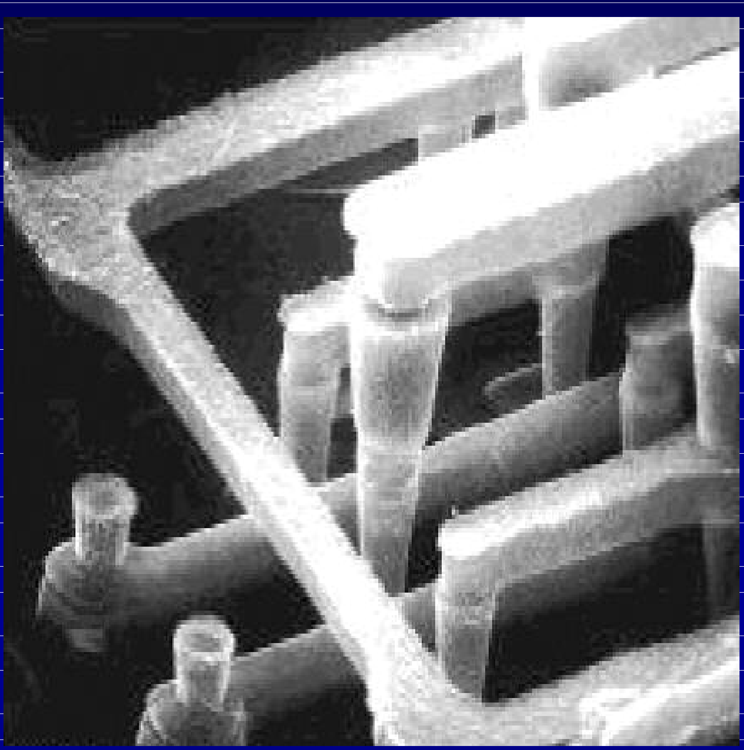


- Reduce order to desired value, e.g., 10




Gnuplot
Semi-logarithmic plot of the singular values
10
1
0.1
0.01
0.001
0.0001
1e-05
1e-06
1e-07
1e-08
0
20
40
60
80
100
120
140
160
180
200
Mapping PDE's (Maxwell Eqs.) to equivalent high order (200) ODE system; no apparent SV clustering for selecting order
Dynamic wire current modelling
Using MOESP/4SID
Singular value plot


Gnuplot
* = target output, -- = estimated output
10-th order linear dynamic model fit (green) versus original FDTD results (red)
Dynamic wire current modelling
Using MOESP/4SID
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
-0.02
-0.025
-0.03
-0.035
0
5e-13
1e-12
1.5e-12
2e-12
2.5e-12
3e-12
3.5e-12
4e-12
4.5e-12
5e-12



Maxwell Equations

FDTD gives V(t), I(t)

Linear state space modelling to get linear dynamic model

Circuit Simulation

Generalized Modelling Formalism

- **Map** linear state space model + parameters to our neural network modelling formalism

Constructive and mathematically exact!

- **Post-optimize** to deal with numerical artefacts of MOESP/4SID (instability & no implicit DC)
- Automatically **generate** lumped linear circuit **models for Pstar**, Spectre, VHDL-AMS, …

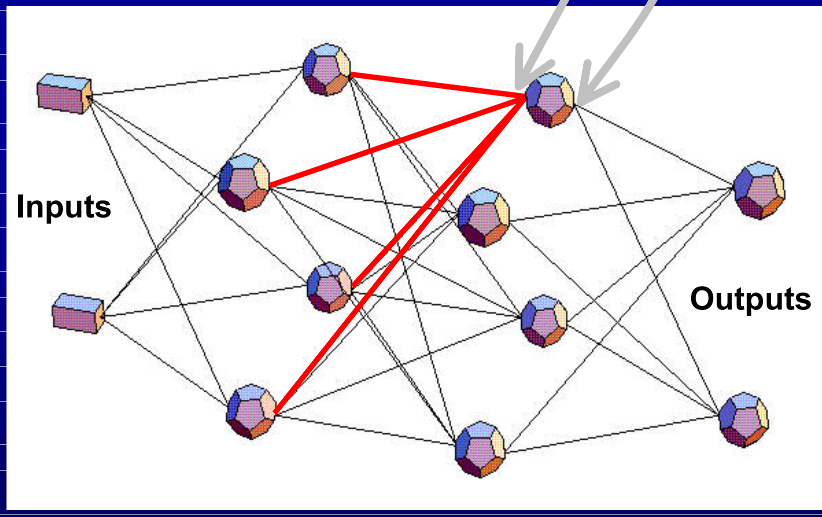
Basic multilayer perceptron theory + extensions (Meijer 1996)

$$s_{ik} \triangleq w_{ik} \cdot y_{k-1} - \theta_{ik} + v_{ik} \cdot \frac{\mathrm{d}y_{k-1}}{\mathrm{d}t} = \sum_{j=1}^{N_{k-1}} w_{ijk}\, y_{j,k-1} - \theta_{ik} + \sum_{j=1}^{N_{k-1}} v_{ijk} \frac{\mathrm{d}y_{j,k-1}}{\mathrm{d}t}$$

Weighted sum s_{ik}

$$\tau_{2,ik} \frac{\mathrm{d}^2 y_{ik}}{\mathrm{d}t^2} + \tau_{1,ik} \frac{\mathrm{d}y_{ik}}{\mathrm{d}t} + y_{ik} = \mathcal{F}^{(ik)}(s_{ik}, \delta_{ik})$$

Differential equation for neuron output y_{ik}



Feedforward neural network

Learning (=Optimization)

- Define cost function, e.g., Σ (model - data)^2
- Discretize and apply optimization algorithm*, involving combinations of

  - DC, TR and AC small signal analysis
  - DC, TR and AC sensitivity (for gradients)

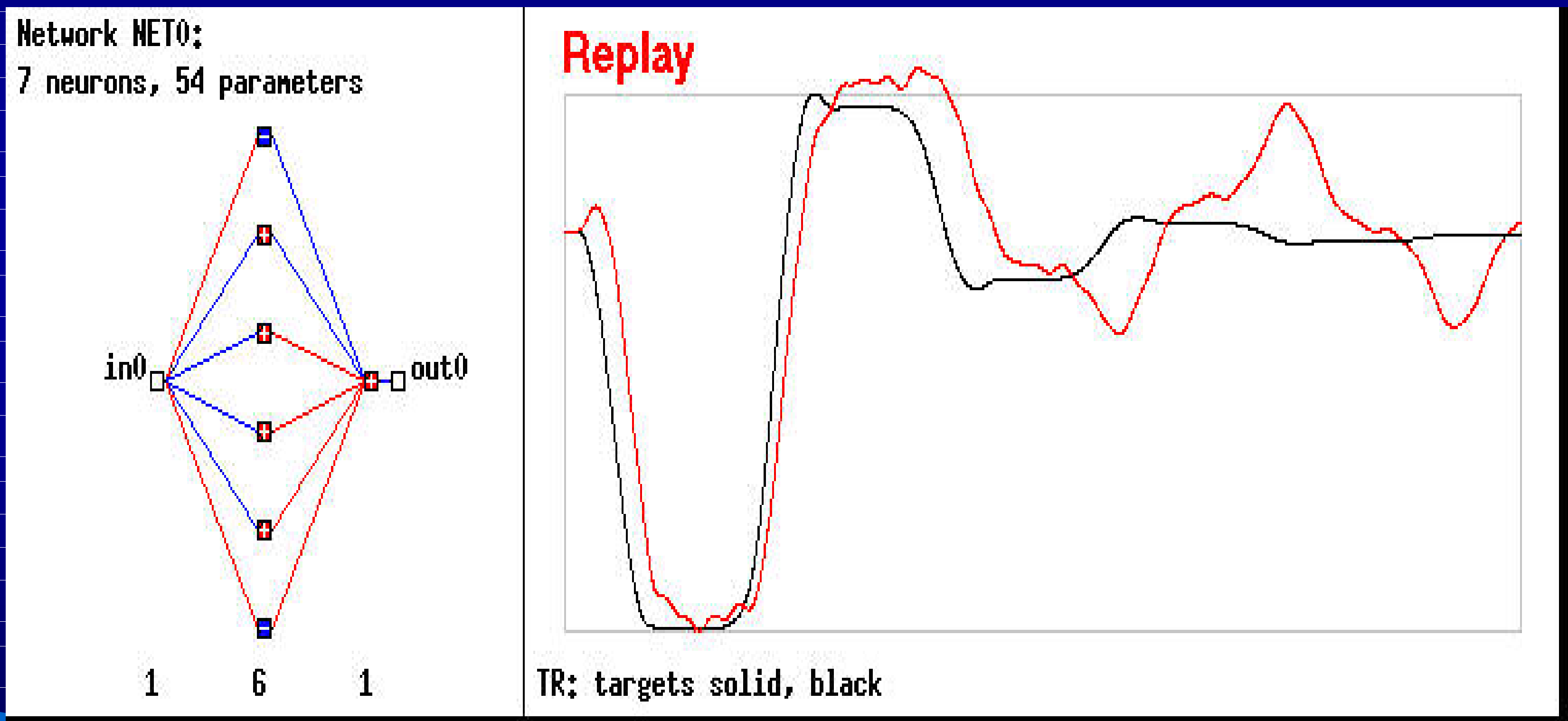
  *Conjugate gradient, BFGS, …

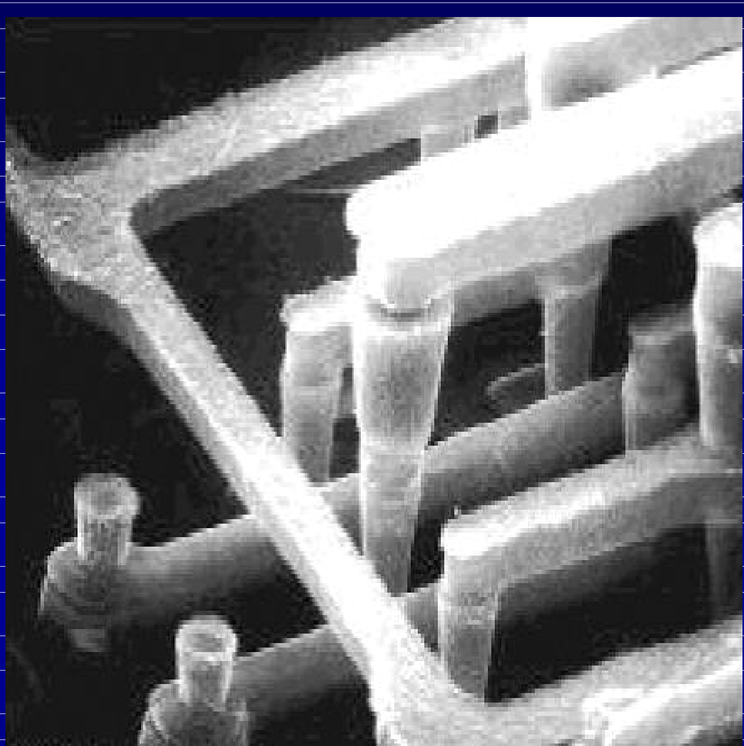
- Risks: slow convergence, local minima

Applying Generalized Formalism

Post-optimize →

Fix MOESP/4SID artefacts:
ensure stable model &
fit with DC initial state





Maxwell Equations

↓

FDTD gives V(t), I(t)

↓

Linear state space modelling to get linear dynamic model

↓

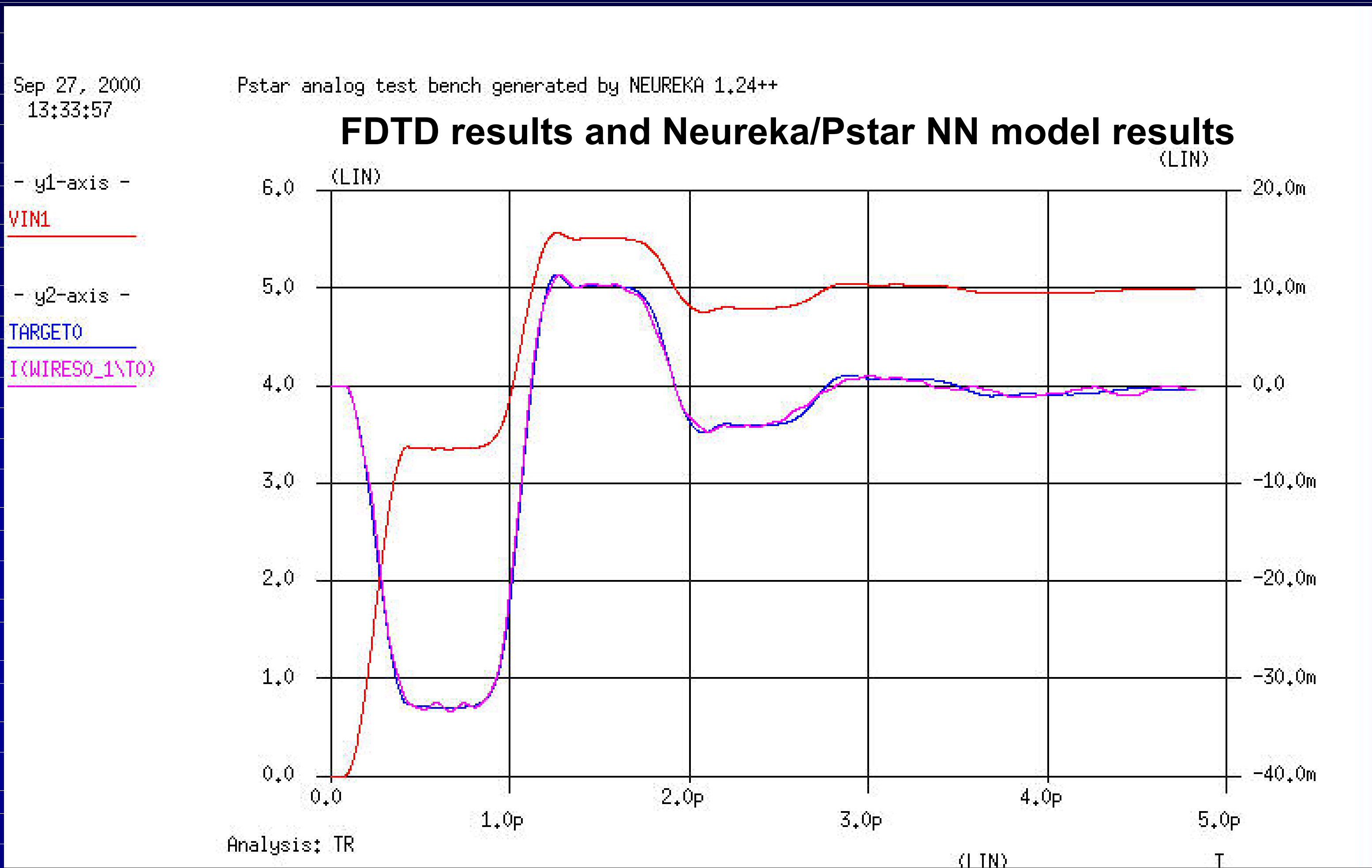
Our generalized formalism

↓

Post-optimize & generate syntax for simulation model (lumped linear circuit model)

↓

Circuit Simulation



Circuit simulation results vs FDTD

Verify model generalization

- **Linear model:** modelling for one signal with all (relevant) frequencies should suffice - **in theory!**

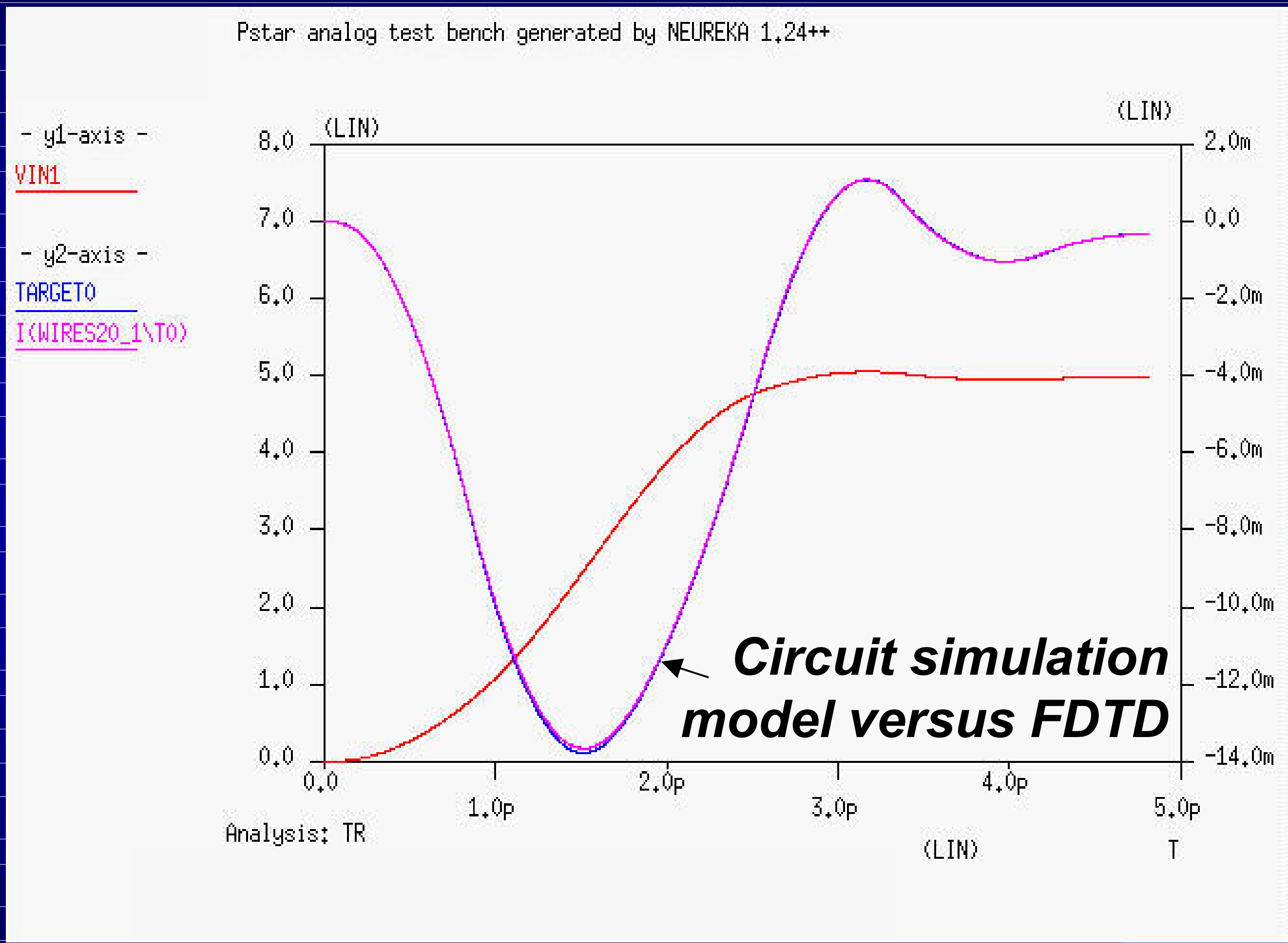
- **Verify:** define a **different stimulus** and check if the FDTD simulation still matches results for the ***unchanged*** circuit model simulation

  If so, that will confirm that the circuit model indeed applies to **all stimuli** - and not just the one(s) used during modelling

Verify model generalization [1]

- *New* stimulus STEP with slope /= 10, applied to
  - FDTD simulation
  - *Unchanged* circuit simulation model

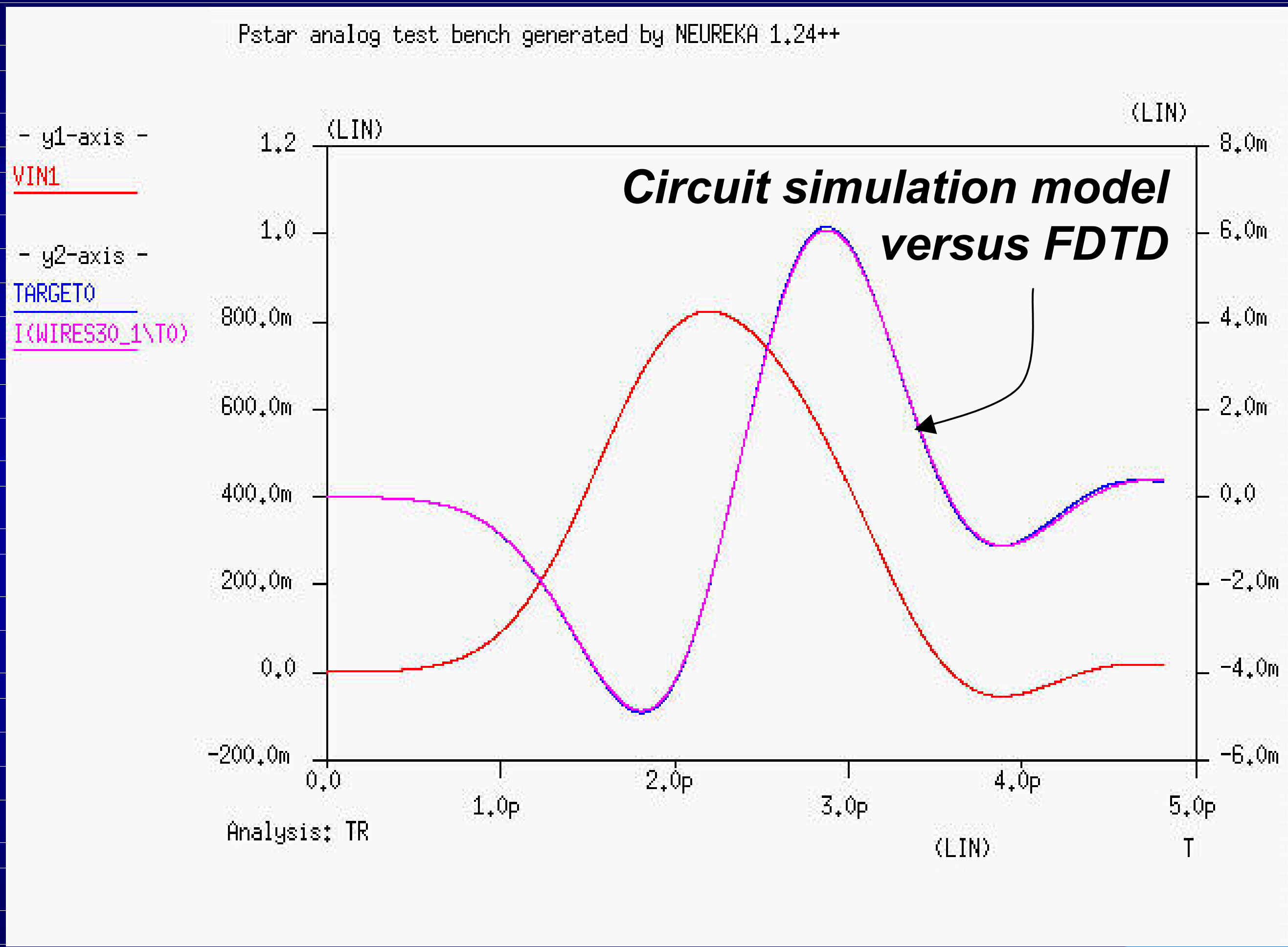
Excellent fit!



Verify model generalization [2]

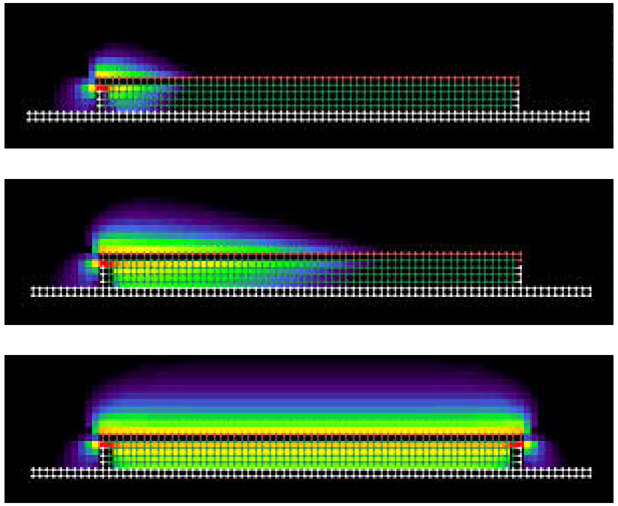
- ***New*** **stimulus GAUSSIAN applied to**
  - **FDTD simulation**
  - ***Unchanged*** **circuit simulation model**

Excellent fit!



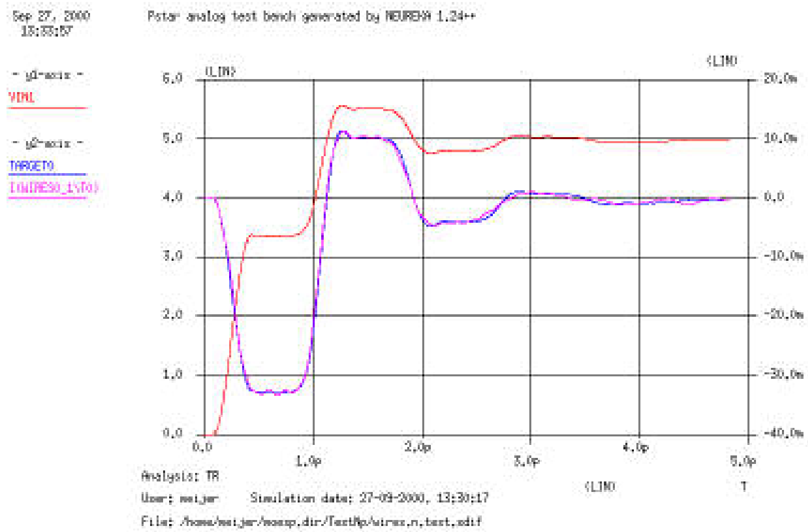
Single Wire Modelling

FDTD simulation of Maxwell Equations in space and time



Complex wire load modelling

Lumped linear dynamic circuit simulation model



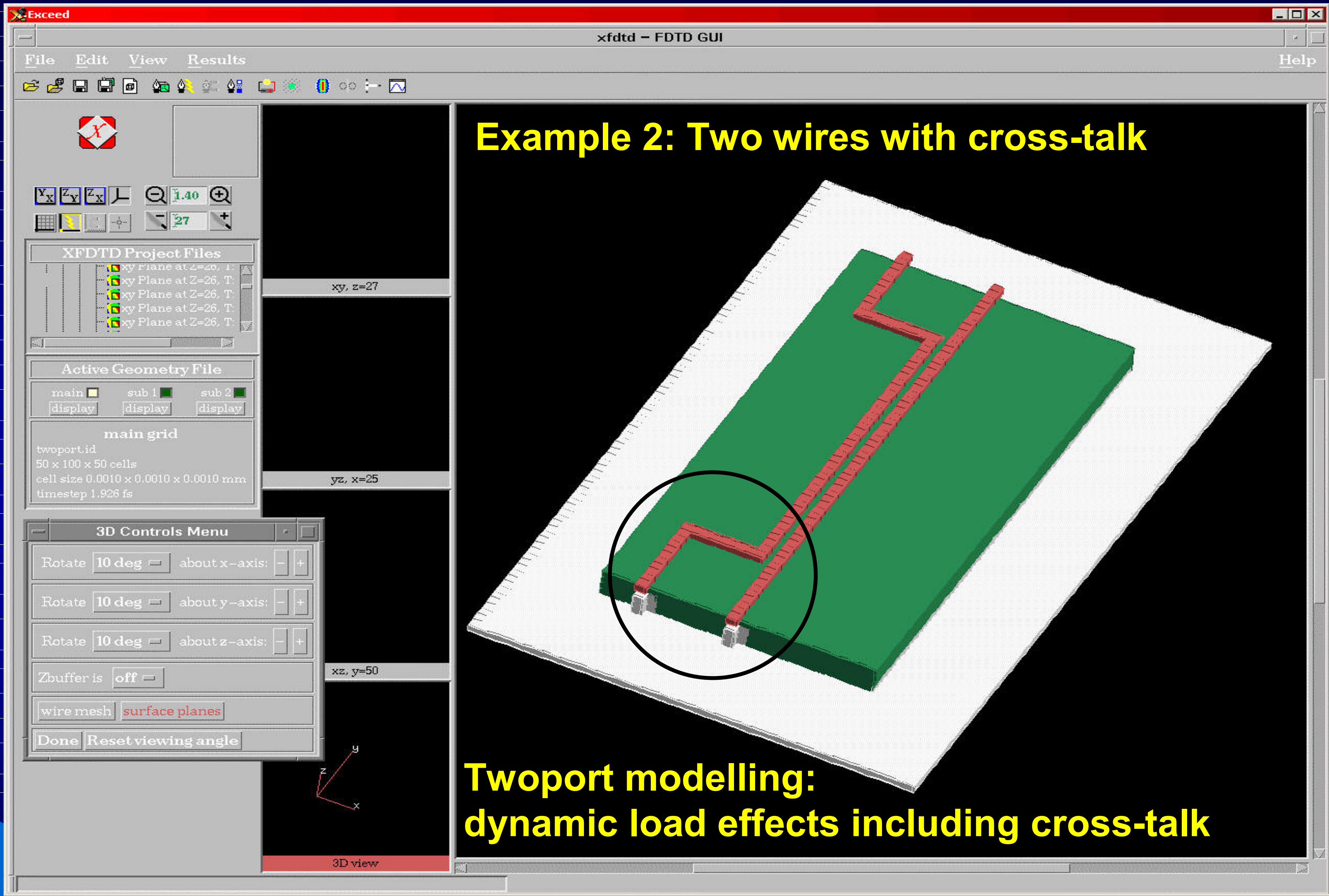
Model Eqs. $10^5 - 10^6$

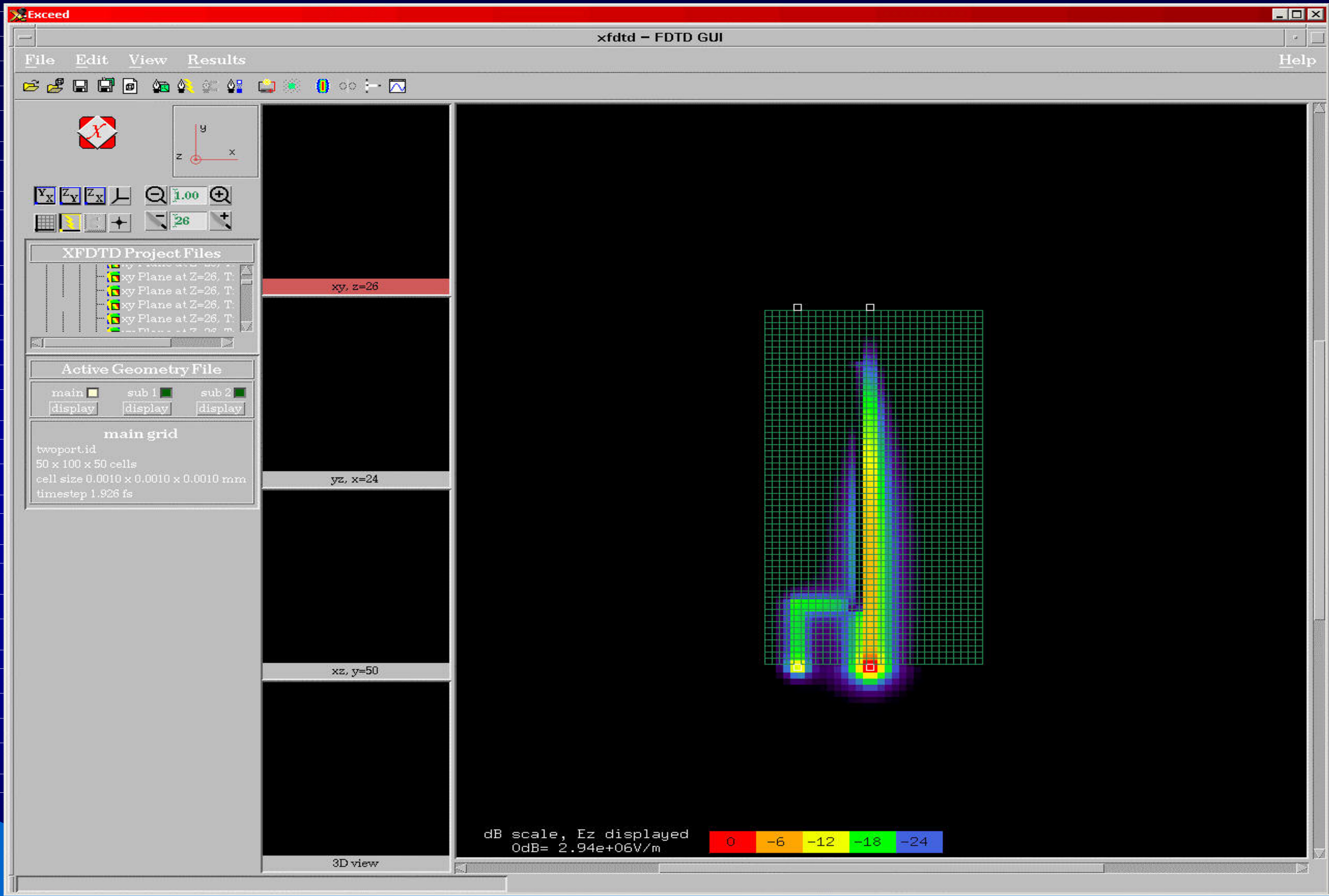
Model Eqs. $10^1 - 10^2$

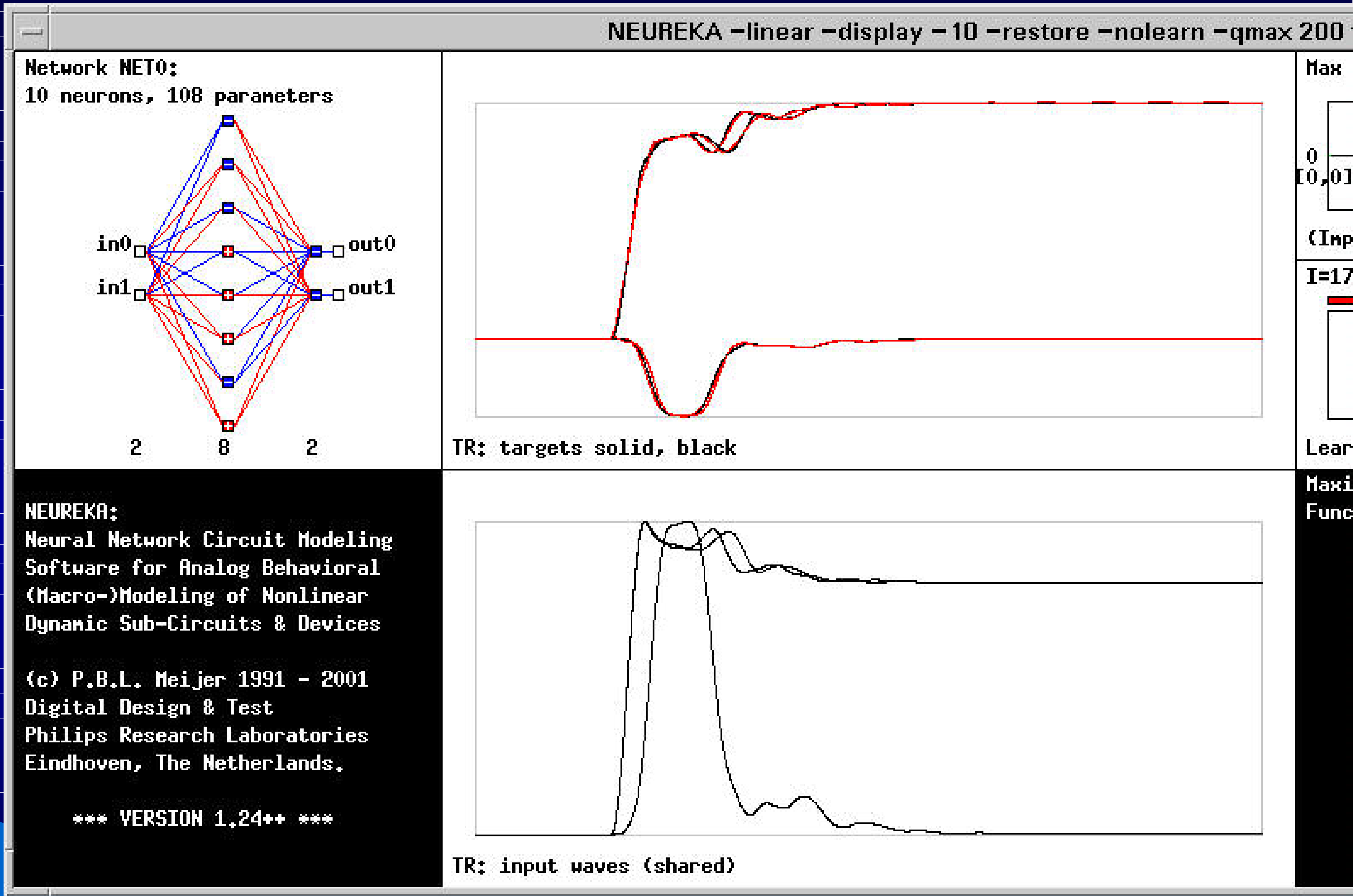
Orders of magnitude gain in simulation speed while preserving detailed (parasitic) effects

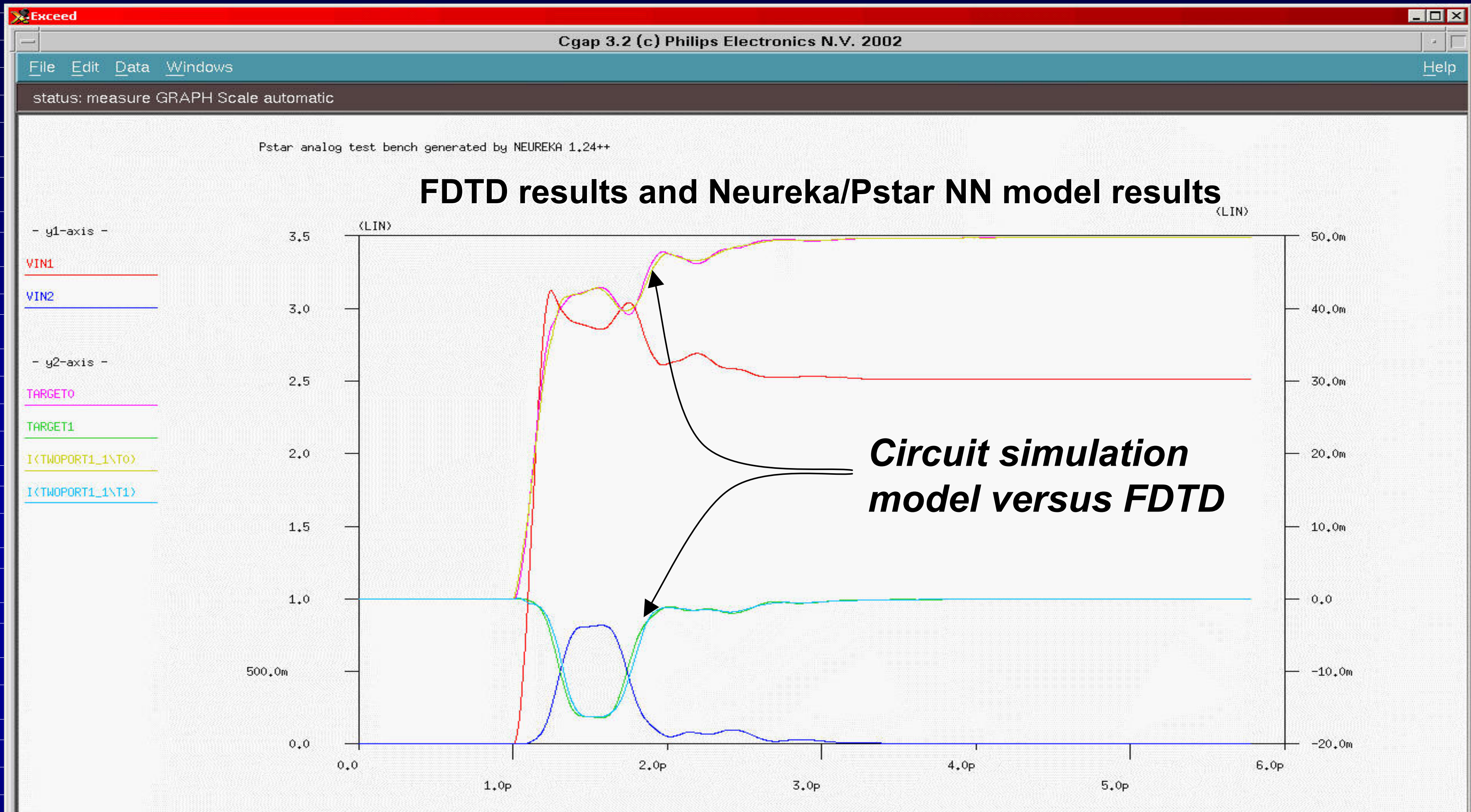
Let's make things better.

PHILIPS


Exceed
xfdtd – FDTD GUI
File
Edit
View
Results
Help
1.40
27
XFDTD Project Files
xy Plane at Z=26, T:
xy Plane at Z=26, T:
xy Plane at Z=26, T:
xy Plane at Z=26, T:
Active Geometry File
main
sub 1
sub 2
display
display
display
main grid
twoport.id
50 x 100 x 50 cells
cell size 0.0010 x 0.0010 x 0.0010 mm
timestep 1.926 fs
3D Controls Menu
Rotate 10 deg about x–axis:
Rotate 10 deg about y–axis:
Rotate 10 deg about z–axis:
Zbuffer is off
wire mesh
surface planes
Done
Reset viewing angle
xy, z=27
yz, x=25
xz, y=50
3D view
Example 2: Two wires with cross-talk
Twoport modelling:
dynamic load effects including cross-talk


Exceed
xfdtd – FDTD GUI
File Edit View Results
Help
1.00
26
XFDTD Project Files
xy Plane at Z=26, T:
xy Plane at Z=26, T:
xy Plane at Z=26, T:
xy Plane at Z=26, T:
Active Geometry File
main sub 1 sub 2
display display display
main grid
twoport.id
50 x 100 x 50 cells
cell size 0.0010 x 0.0010 x 0.0010 mm
timestep 1.926 fs
xy, z=26
yz, x=24
xz, y=50
3D view
dB scale, Ez displayed
0dB= 2.94e+06V/m
0 -6 -12 -18 -24


NEUREKA -linear -display -10 -restore -nolearn -qmax 200
Network NET0:
10 neurons, 108 parameters
in0
in1
out0
out1
2
8
2
TR: targets solid, black
Max
0
[0,0]
(Imp
I=17
Lear
NEUREKA:
Neural Network Circuit Modeling
Software for Analog Behavioral
(Macro-)Modeling of Nonlinear
Dynamic Sub-Circuits & Devices
(c) P.B.L. Meijer 1991 - 2001
Digital Design & Test
Philips Research Laboratories
Eindhoven, The Netherlands.
*** VERSION 1.24++ ***
TR: input waves (shared)
Maxi
Func



Circuit simulation results vs FDTD

PostScript file twoport.n.test.eps generated
File '/home/meijer/moesp.dir/TestNp/gaussian.n.test.sdif' is read.
PostScript file gaussian.n.test.eps generated

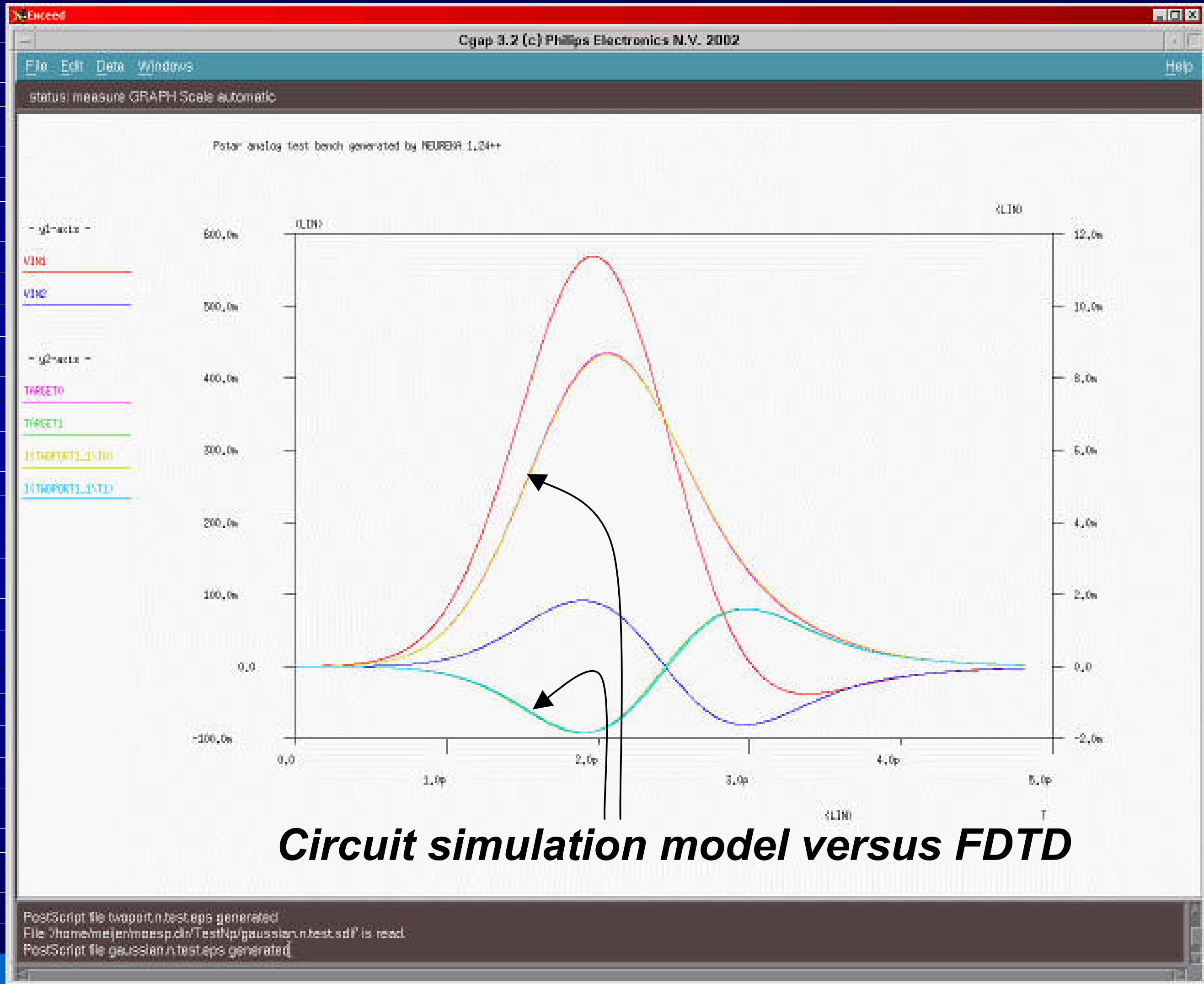
Let's make things better.

PHILIPS

Verify twoport model generalization

- *New* stimulus GAUSSIAN applied to
  - FDTD simulation
  - *Unchanged* circuit simulation model

Excellent fit!



Conclusions

- New numerical interconnect analysis options going well beyond 100 GHz
- Many/All RF4D parasitics can be included: capacitive and inductive effects, skin effect
- Reference for validation of new design rules
- Reference for validation of alternative models