The Development of a 2D Transmission-Line-Modeling Alternating-Direction-Implicit Method

Yu-Min Lee

Charlie Chung-Ping Chen

Department of Electrical and Computer Engineering
University of Wisconsin-Madison

Sponsored by Avant! Corp. and Intel Corp.

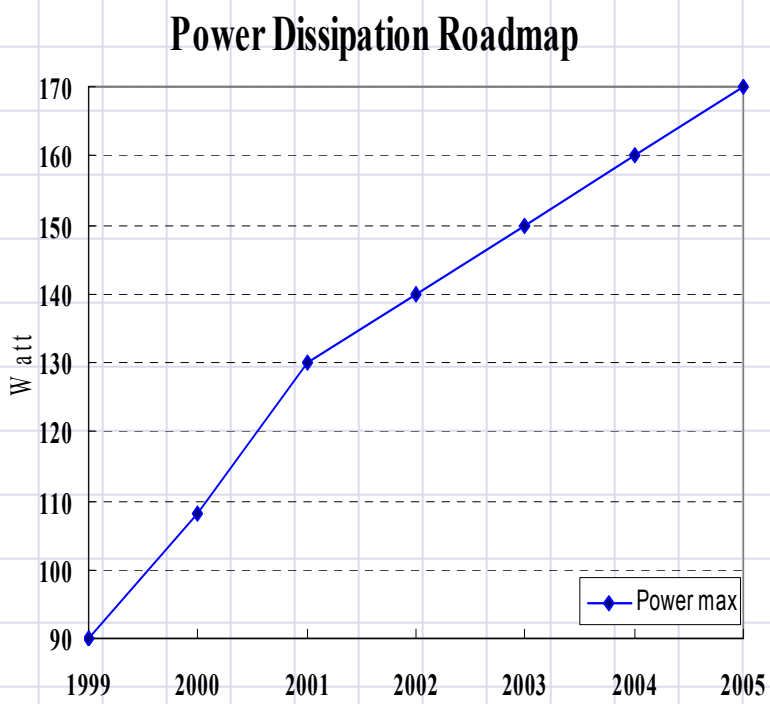
Outline

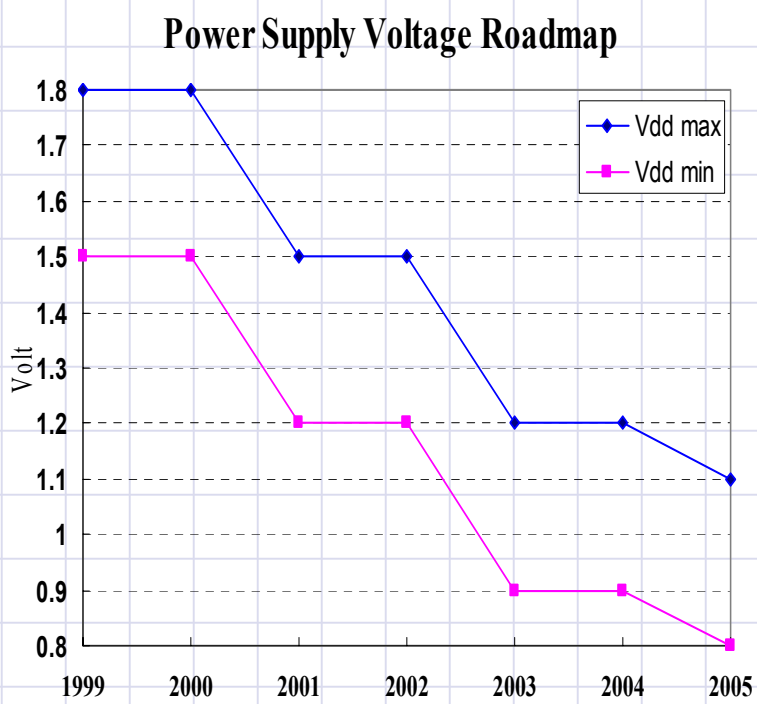
- Motivation
- Power Grid Modeling and Transmission-Line-Equations
- Finite Difference Method
- Transmission-Line-Modeling

  Alternating-Direction-Implicit Method
- Experimental Results
- Conclusion

Motivation: Trend of VLSI Technology

- Increasing power dissipation
- Decreasing supply voltage





* Data from International Technology Roadmap for Semiconductors (ITRS)

Motivation: Why Power Grids Analysis?

- Power fluctuation sources increase significantly
  - IR-drop: $\Delta V = I \times R = (P\uparrow / Vdd\downarrow) \times R$
  - L di/dt: $\Delta V = L\, di/dt$

    roughly proportional to $L(I \times f) = L\,(P\uparrow \times f\uparrow\uparrow / Vdd\downarrow)$

    - I : consuming current
    - f : clock frequency
    - P : power dissipation
    - Vdd : supply voltage
  - Other noises such as resonance and electromigration also affect power grid reliability
- Power delivery quality becomes a critical issue

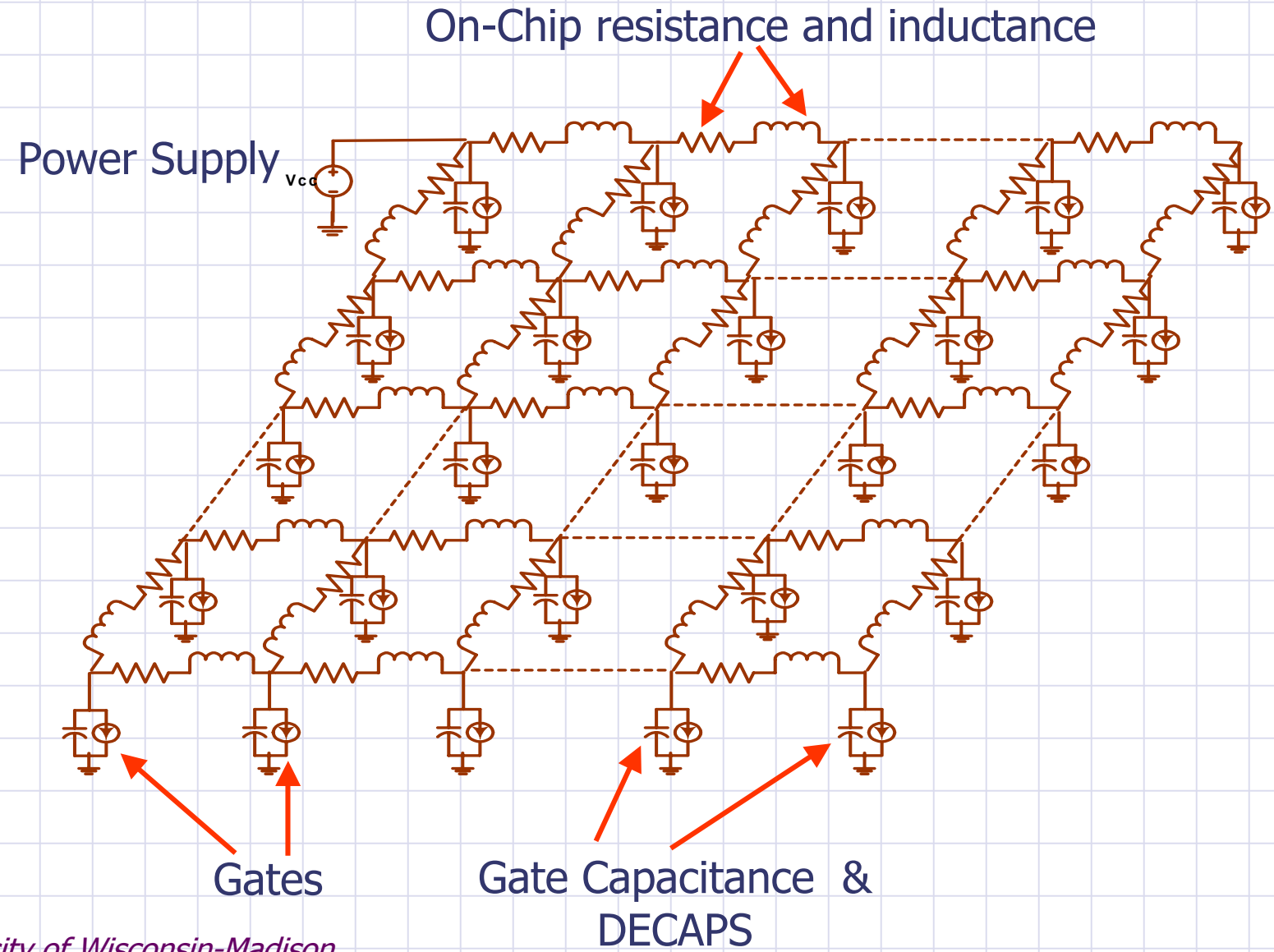
Motivation: Power Grids Analysis Challenges

- More than 40-million transistors on a chip
- Sparse direct method takes super linear time to solve a matrix.
  - Introduce a large amount of fill-ins
  - Slow and huge memory requirement
  - SPICE takes 6 hours to finish DC analysis for an 80,000-node circuit
  - How about more than millions?

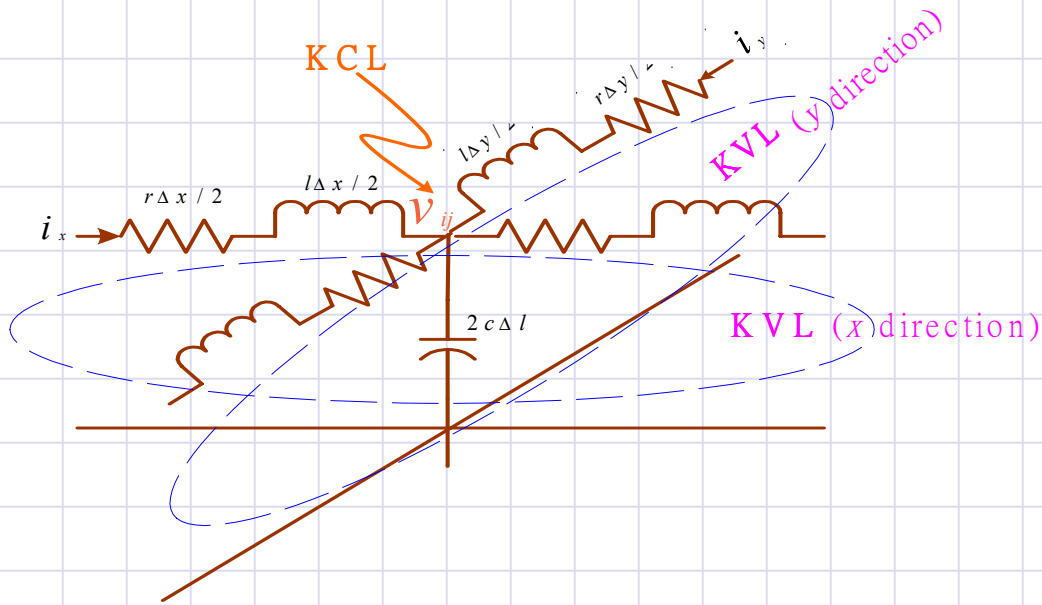
Literature Overview

- Hierarchical analysis of power distribution network (Sachin-DAC2000)
  - Decoupled linear and nonlinear simulation (only for RC circuit)
  - Divide and Conquer simulation for large scale simulation
  - Sparsfication to reduce memory usage
- Efficient large-scale RLC power grid analysis based on preconditioined Krylov-subspace iterative methods (Chen-DAC2001)
  - Incomplete cholesky decomposition and iterative method to reduce fillins and number of iterations -> over 100X speed up
- Multigrid-like technique for power grid analysis (Sani R. Nassif-ICCAD2001)
  - Reduce the grids to a coarser structure, and the solution is mapped back to the original grid

Power Grid Modeling



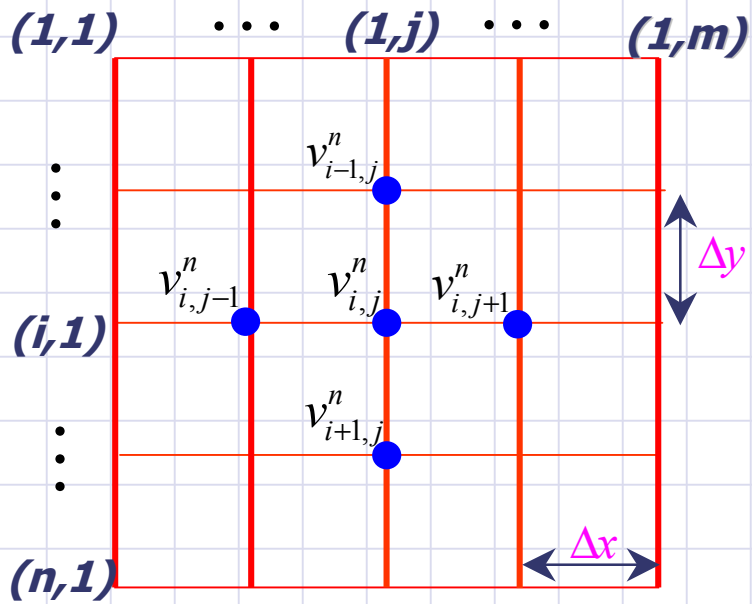
MNA and Transmission-Line-Equations



- For each cell, write down **KCL** at center node, **KVL** at x- and y-directions. Then, taking $\Delta x - 0, \Delta y - 0, \Delta l - 0$

$$\left.\begin{aligned}
\frac{\partial v}{\partial t} &= \frac{1}{2c}\left(-\frac{\partial i_x}{\partial x} - \frac{\partial i_y}{\partial y}\right) \\
\frac{\partial i_x}{\partial t} &= \frac{1}{l}\left(-\frac{\partial v}{\partial x} - r\, i_x\right) \\
\frac{\partial i_y}{\partial t} &= \frac{1}{l}\left(-\frac{\partial v}{\partial y} - r\, i_y\right)
\end{aligned}\right\} \text{Transmission Line Equations}$$

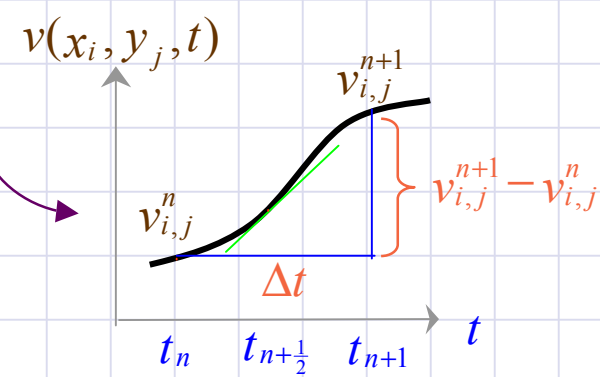
Simple Explicit Finite Difference Method



- Central-Finite-Difference Approximation for first derivatives

$$\frac{\partial v(x_i, y_j, t)}{\partial t_{n+\frac{1}{2}}} \approx \frac{v^{n+1}_{i,j} - v^n_{i,j}}{\Delta t}$$

$$\frac{\partial i_x(x, y_j, t_{n+\frac{1}{2}})}{\partial x_i} \approx \frac{i_x\big|^{n+\frac{1}{2}}_{i+\frac{1}{2},j} - i_x\big|^{n+\frac{1}{2}}_{i-\frac{1}{2},j}}{\Delta x}$$



Simple Explicit Finite Difference Method (*Cont.*)

$$\frac{\partial v}{\partial t} = \frac{1}{2c}\left(-\frac{\partial i_x}{\partial x} - \frac{\partial i_y}{\partial y}\right) \quad \Rightarrow \quad \frac{v_{i,j}^{n+1} - v_{i,j}^{n}}{\Delta t} = \frac{1}{2c}\left(-\frac{i_x\big|_{i+\frac{1}{2},j}^{n+\frac{1}{2}} - i_x\big|_{i-\frac{1}{2},j}^{n+\frac{1}{2}}}{\Delta x} - \frac{i_y\big|_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} - i_y\big|_{i,j-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta y}\right)$$

$$\frac{\partial i_x}{\partial t} = \frac{1}{l}\left(-\frac{\partial v}{\partial x} - r\, i_x\right) \quad \Rightarrow \quad \frac{i_x\big|_{i+\frac{1}{2},j}^{n+\frac{1}{2}} - i_x\big|_{i+\frac{1}{2},j}^{n-\frac{1}{2}}}{\Delta t} = \frac{1}{l}\left(-\frac{v_{i+1,j}^{n} - v_{i,j}^{n}}{\Delta x}\right)$$

$$\frac{\partial i_y}{\partial t} = \frac{1}{l}\left(-\frac{\partial v}{\partial y} - r\, i_y\right) \quad \Rightarrow \quad \frac{i_y\big|_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} - i_y\big|_{i,j+\frac{1}{2}}^{n-\frac{1}{2}}}{\Delta t} = \frac{1}{l}\left(-\frac{v_{i,j+1}^{n} - v_{i,j}^{n}}{\Delta y}\right)$$

- No matrix inversion, only substitutions, thus the runtime is linear
- Time steps are limited by stability constraint $\Delta t \le \sqrt{lc} \Big/ \sqrt{1/(\Delta x)^2 + 1/(\Delta y)^2}$

  For example, for the VLSI technology with feature size as 0.1 μm and the dielectric permittivity as 4, the stability constraint is close to 0.47 fs.

$\Longrightarrow$ Need 2.1 millions time steps to simulate an 1ns period

We Propose the Transmission-Line-Modeling method based on Alternating-Direction-Implicit (ADI)

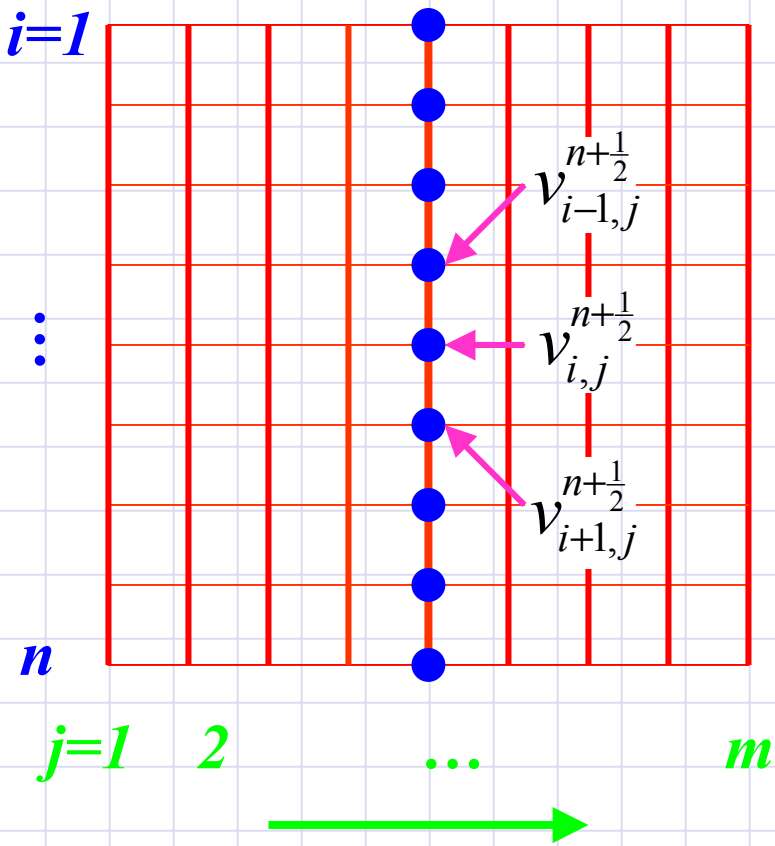
No ADI method was applied in TLM analysis

- Achievements
  - We prove that our TLM-ADI method is unconditionally stable, no limits on time step for stability
  - No large scale matrix inversion, only tridiagonal matrix solving. Therefore, the runtime is linear
  - Solve higher dimension problem by successive lower dimension methods
  - Alternating X and Y directions for implicit and explicit solving

Illustration of ADI

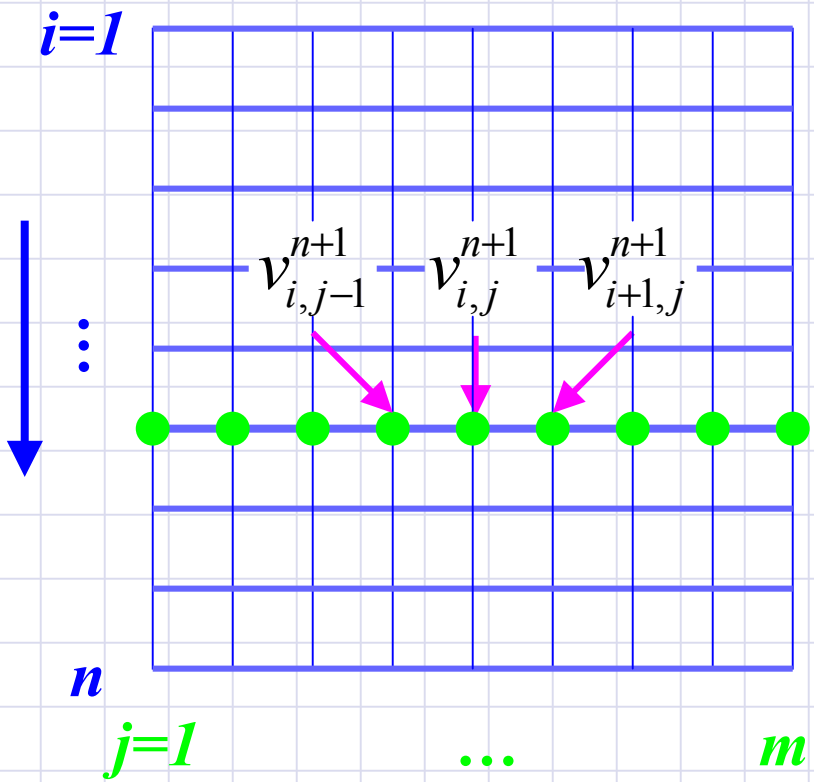
Step I

Y-direction implicit



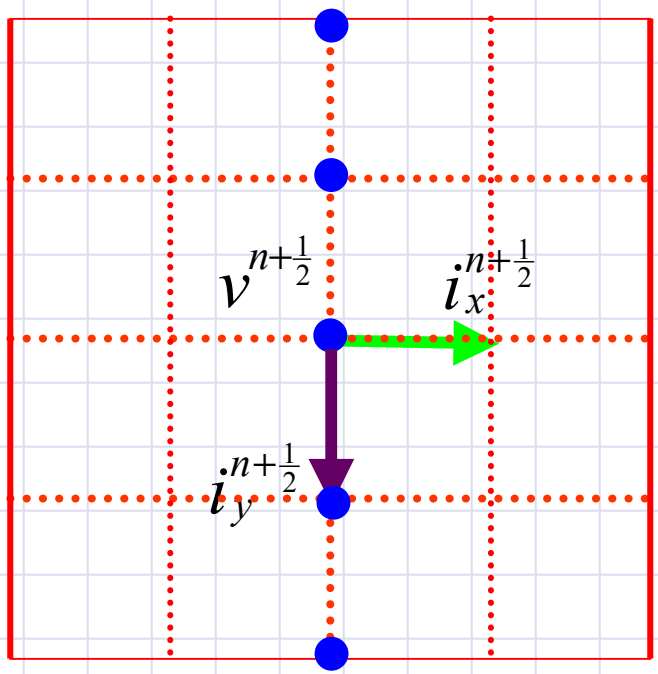
Step II

X-direction implicit



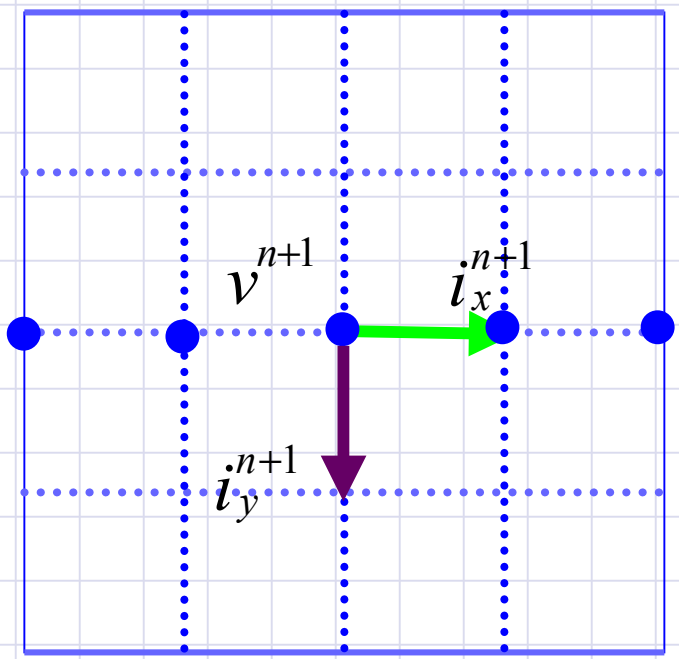
Alternating-Direction-Implicit (ADI) Method *(Cont.)*

- **Step I: Form n-th to n+1/2-th time step**
  1. Explicitly update $i_x^{n+\frac{1}{2}}$
  2. Implicitly update $v^{n+\frac{1}{2}}$
  3. Explicitly update $i_y^{n+\frac{1}{2}}$ by using updated node voltage $v^{n+\frac{1}{2}}$



Alternating-Direction-Implicit (ADI) Method *(Cont.)*

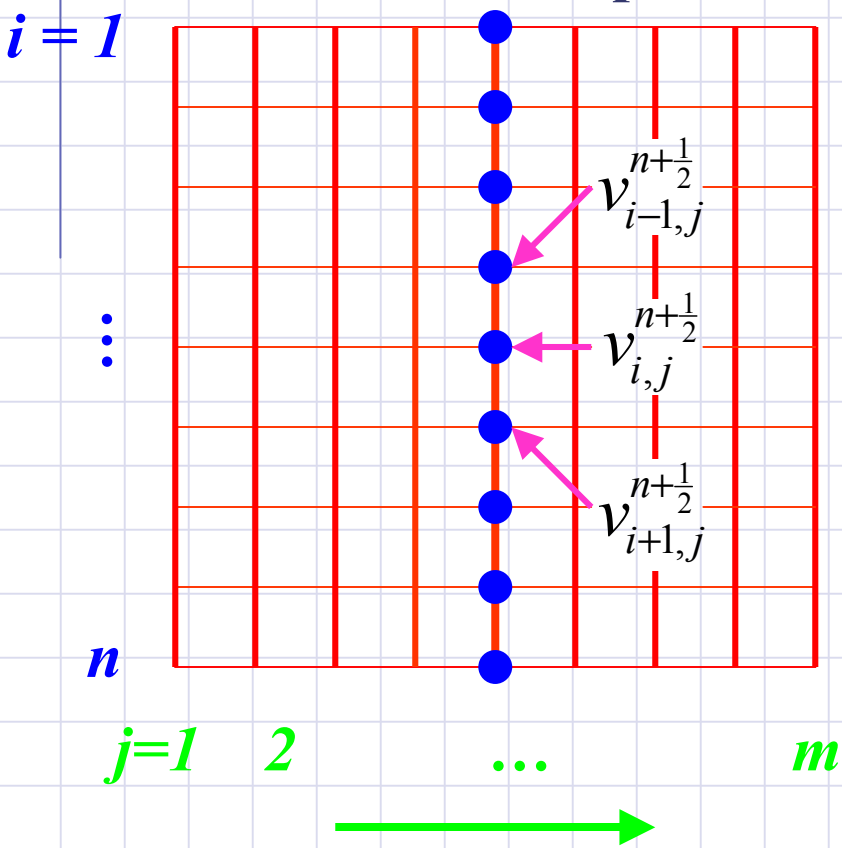
- **Step II: From n+1/2-th to n+1-th time step**
  1. Explicitly update i_y^{n+1}
  2. Implicitly update v^{n+1}
  3. Explicitly update i_x^{n+1} by using update node voltage v^{n+1}



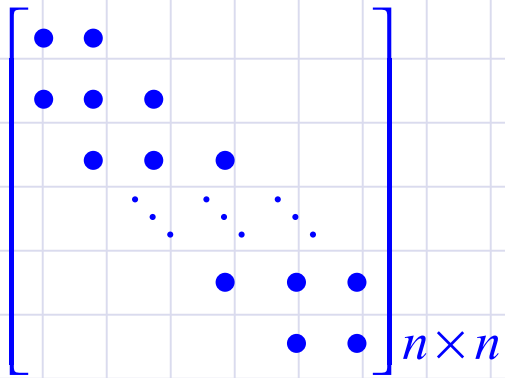
Analysis of ADI Method: Linear Run Time

Step I

Y-direction implicit



Tridiagonal Matrix



2xmxn = 2mn =2N

2 steps m matrices tridaigonal matrix

⟹ Time complexity: O(N)

Analysis of ADI Method: Unconditionally Stability

We can analytically prove the TLM-ADI method is **unconditionally stable**, by using Von Neumann analysis in the LC circuit ($r=0$) as follows

1. $\mathbf{F}^n = [i_x^n \quad i_y^n \quad v^n]^T$ represents the spatial Fourier-transformation of currents and voltage at time n
2. From Step I and II of the TLM-ADI method

$$\Longrightarrow \quad \mathbf{F}^{n+1} = \mathbf{M}\,\mathbf{F}^n$$

   where $\mathbf{M}$ is the gain matrix from time-step n to time-step n+1
3. We analytically solve all the eigenvalues of M and find out that the amplitude of all eigenvalues of M are all 1's

Hence, TLM-ADI is unconditionally stable

Experimental Results: Run time Comparison

All simulation results are performed on an alpha workstation with Dual SLOTB 667 MHZ Alpha 21264 processors

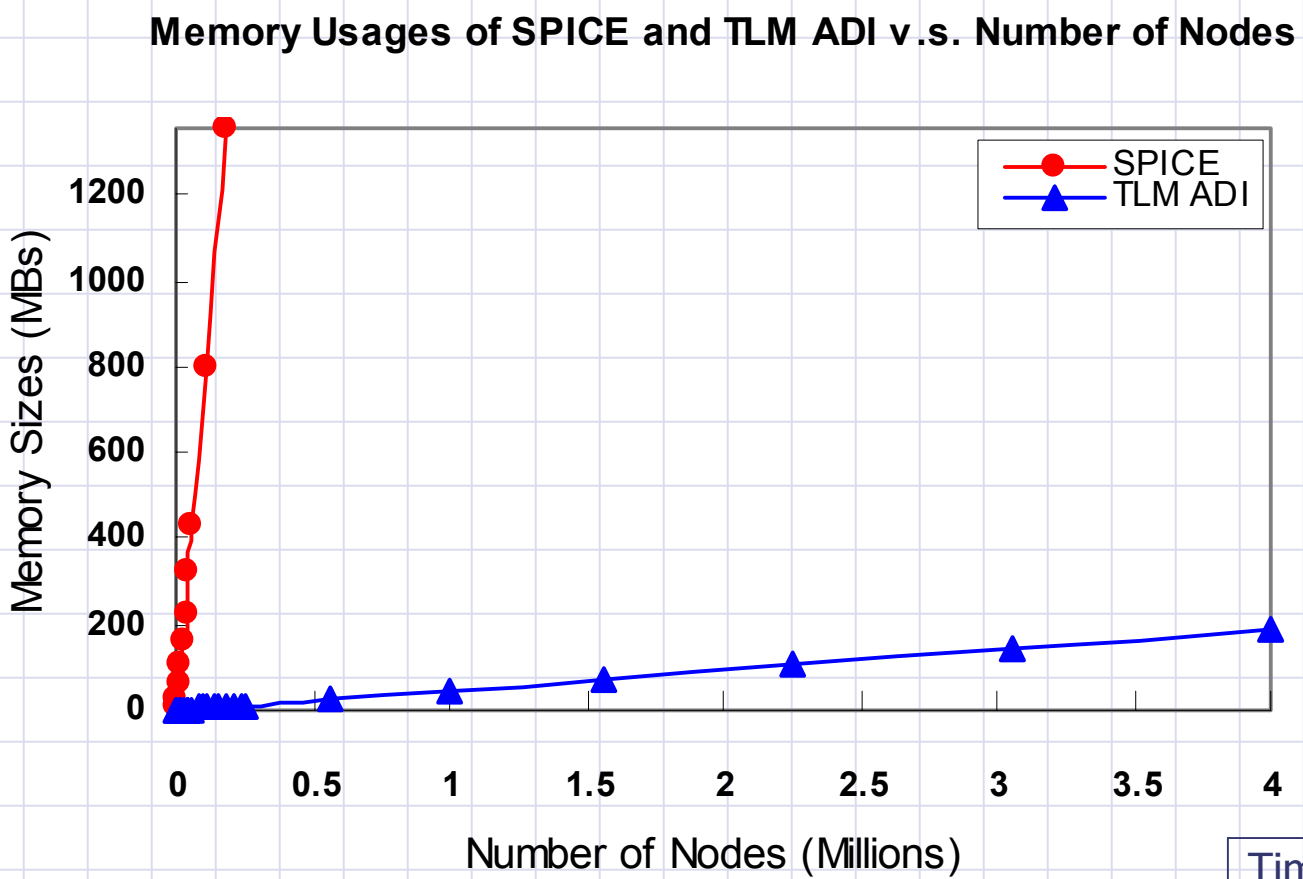
# of Nodes	Run Time (Seconds)		Speed up
	SPICE	ADI	
1,600	35.26	0.31	113
10,000	533.02	2.26	236
25,600	3,433.69	7.94	432
48,400	10,702.30	18.54	577
78,400	25,182.35	34.64	727
115,600	51,393.15	58.39	880
136,900	71,921.13	70.59	1019
184,900	122,730.40	103.97	1180
1,000,000	-	776.14	-
4,000,000	-	3,237.87	-

Time-step: 1ps

Time period: 500ps

TLM-ADI is over 1000 times faster than SPICE

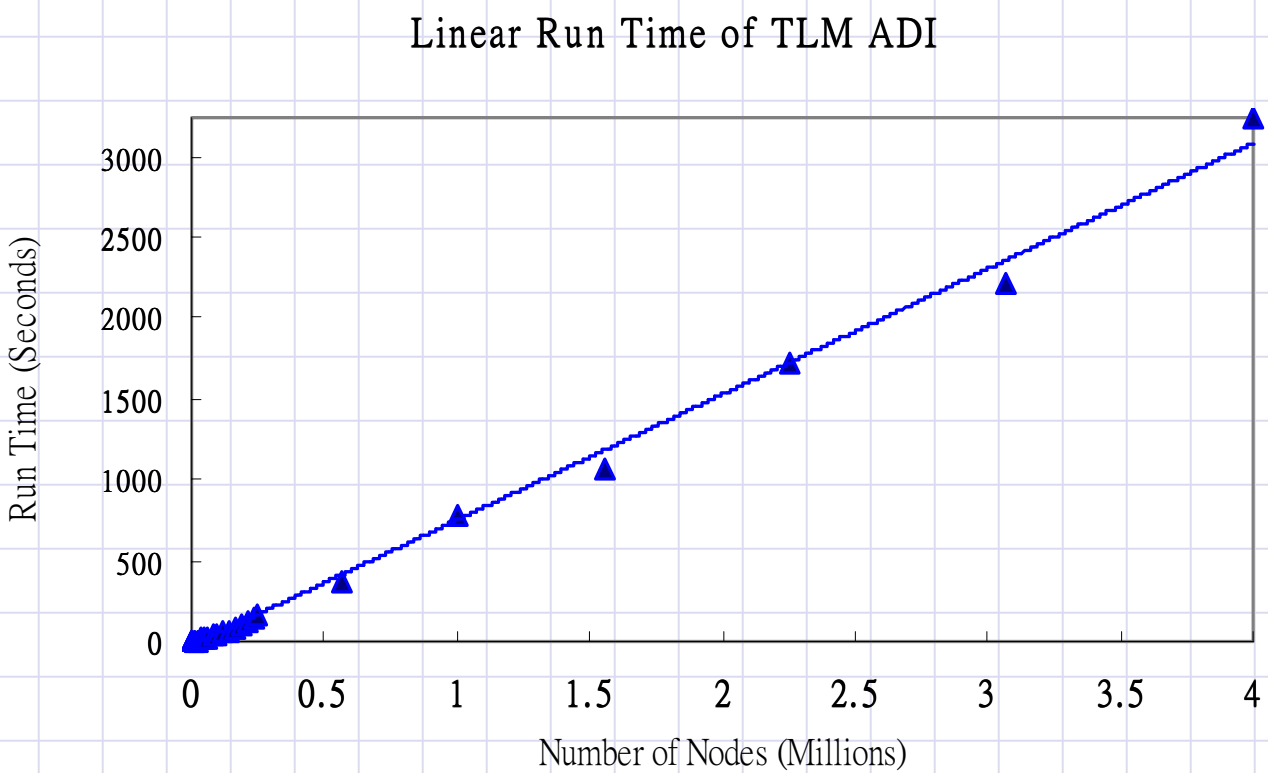
Experimental Results



Time-step: 1ps

Time period: 500ps

Experimental Results: Linear Run Time

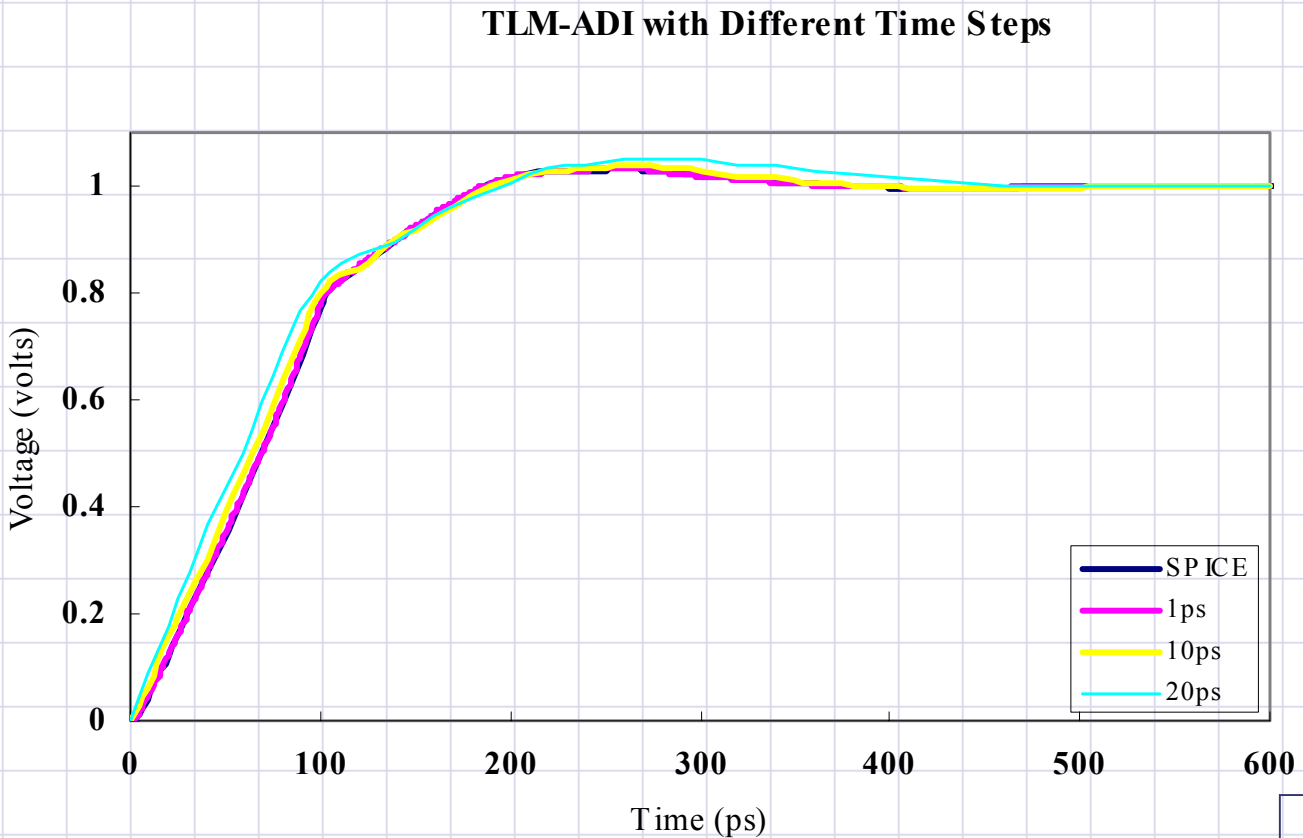


Time-step: 1ps

Time period: 500ps

TLM-ADI need only 6.5 seconds to finish an iteration of 4 million-node power grid circuits, and the **run time is linear**

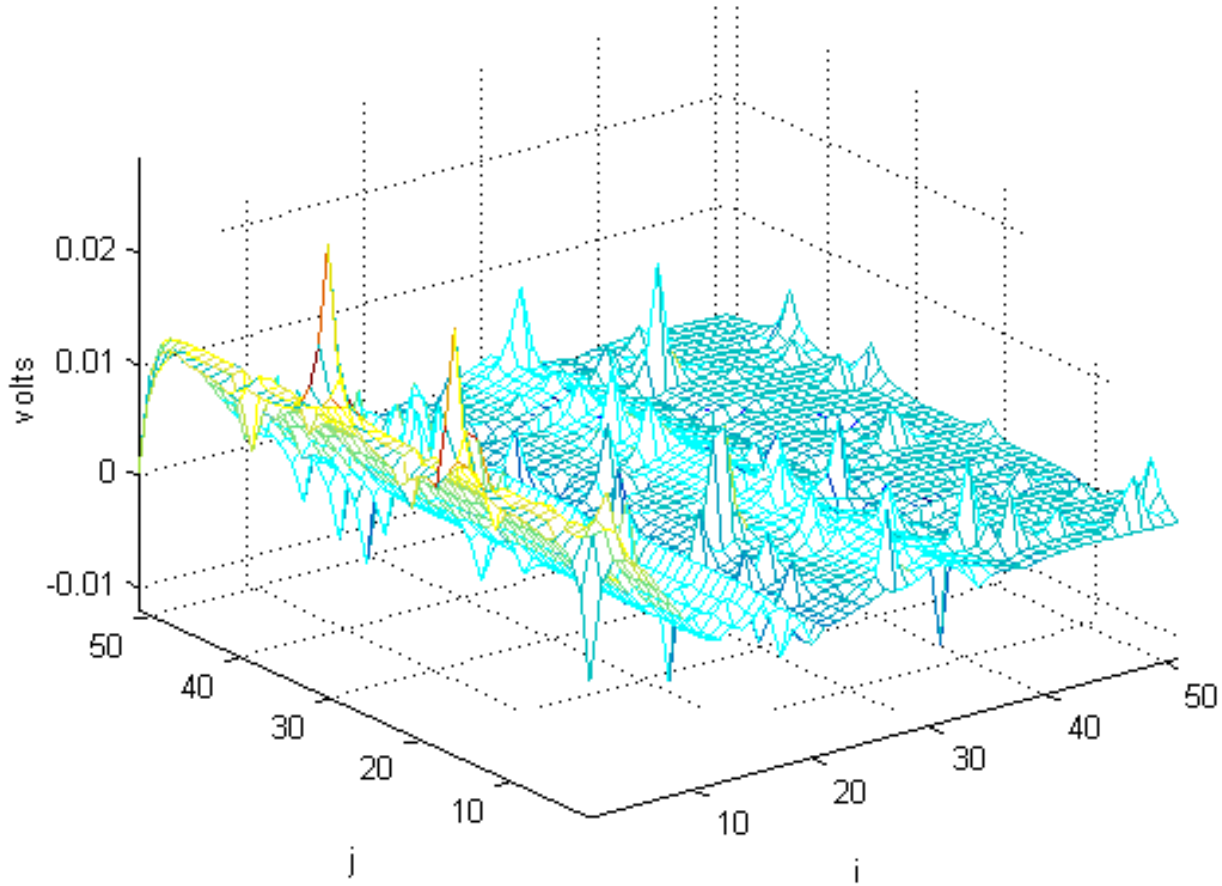
Experimental Results: Unconditionally Stable



# of nodes: 100

The Courant stability constraint is *1.9442* ps

A snapshot of the transient response



Conclusion

- An efficient TLM-ADI algorithm for transient power grid simulation is developed.
- TLM-ADI method is unconditionally stable, and its run time is linear.
- The numerical simulation shows that TLM-ADI method not only speeds up 1000 times over the SPICE but also cuts down the memory requirement

Thank you for your attention!