Graph Based Next Generation Energy Management System

Zhiwei Wang

GEIRI North America

August 18, 2018

GEIRI North America

STATE GRID
GEIRI NORTH AMERICA
全球能源互联网美国研究院

GEIRI North America (GEIRINA)

- Founded in Dec. 2013 in Santa Clara, California, USA (www.geirina.net)
- Conducts cross-disciplinary R&D for power system modernization
- R&D subsidiary and overseas platform of State Grid Corporation of China
- 30+ Researchers and Engineers (50+ in summer) (still recruiting)

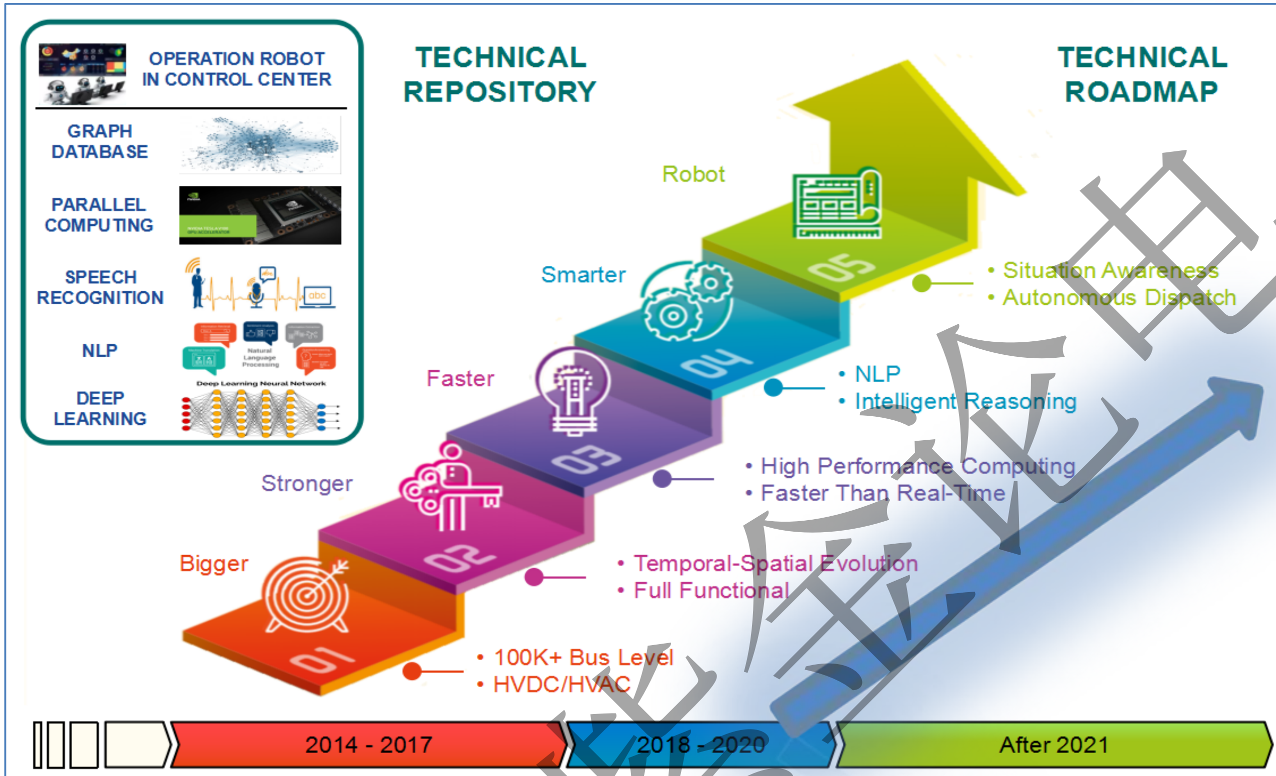
Research Groups & Areas

- Advanced Computing in Power System
- Power System Artificial Intelligence
- Integrated and Distributed Energy Services
- Intelligent Sensing and Chip for Energy System

Content

Next Generation EMS Roadmap

STATE GRID
GEIRI NORTH AMERICA
全球能源互联网美国研究院



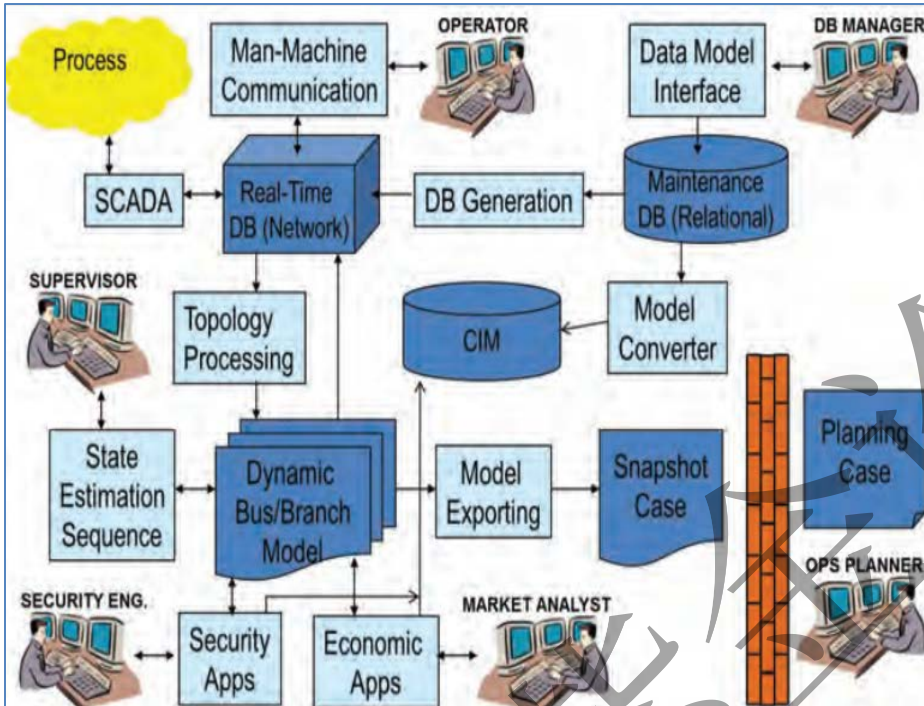
- ❑ 05-The final goal is Robot EMS
- ❑ 02-The industry is at the stage to make EMS full functional
- ❑ 03-The critical path to meet the gap is faster than real-time EMS

Next Generation EMS Goals:

- ✓ Provides real-time, proactive, intelligent, and predictable operation system in control center.
- ✓ Employs graph database, parallel computation, natural language processing, deep learning, and situation awareness and autonomous dispatch to drive anlaytical EMS to intelligent/robot EMS.

Faster-than-Real-Time EMS

Why We Need it



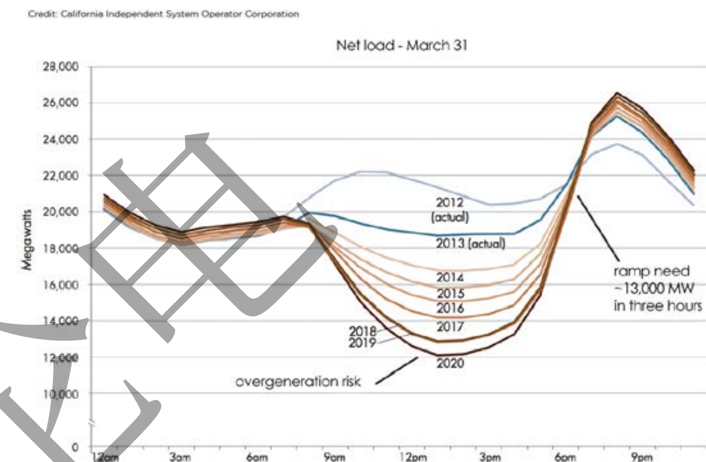
EMS at Present ◀◀ **Gap** ▶▶ **EMS in Future**

- ❑ EMS in future supported by AI decision requires sophisticated model, intensive calculation, and **fast computing**
- ❑ **High performance computing** is the key to make EMS intelligence
- ❑ **Accelerating** application computing is critical for next generation EMS

Faster-than-Real-Time EMS

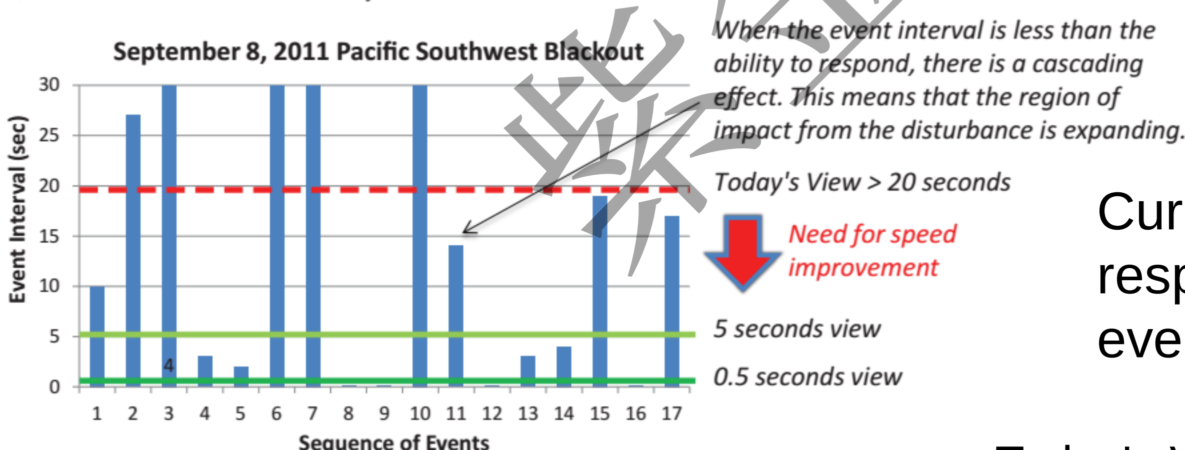
Why We Need it

Power Electronics Dominated
Large and Complex System



Fast Change of Operation in Minutes
Due to the Renewable Intermittence

Credit: Pacific Northwest National Laboratory

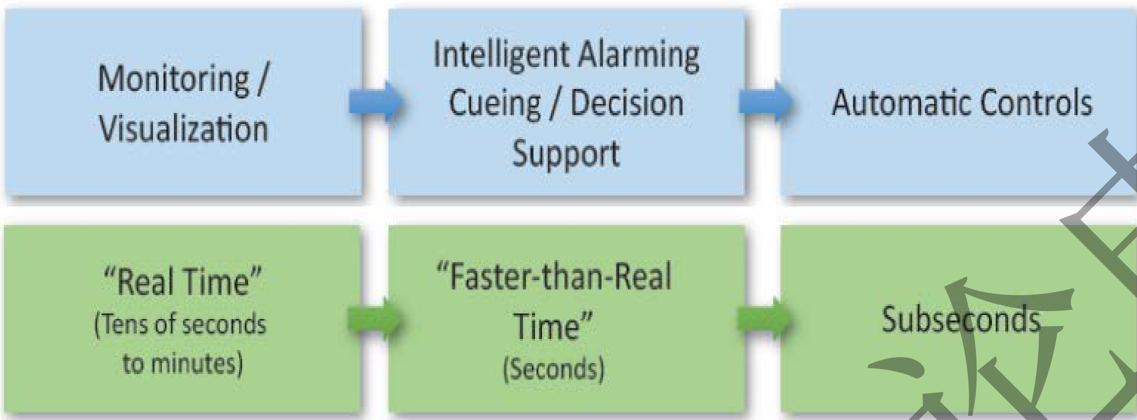


Sequence of Cascading Events in the
2011 Southwest Blackout in the US

Current EMS cycle delays responses to the cascding events driving the blackout

Today's View > 20 seconds
Needed View < 0.5 seconds

Faster-than-Real-Time EMS
Why We Need it



Pathway to Speed Improvements in Analytical Decision Making

Subseconds analytical processing time enabling the critical functions:

- ❑ Security Constrained Automatic Generation Controls
- ❑ On-line SPS Arming Decision
- ❑ Heavy Loaded Power Flow Accurate Solution

- ❑ The analytical processing time needs to be reduced, from tens of seconds to subseconds, to move from monitoring and visualization to automatic controls.
- ❑ The need for fast and predictive analytics is amplified by physical and cyber attack on critical infrastructure.
- ❑ US DOE requires to develop State Estimation at 0.5 seconds speed for medium size system. DOE funded $220 Millions to Faster Real-time Analytical Tools.

Faster-than-Real-Time EMS
What We Achieved

Test System： A Real Provincial System （2650 Bus）			
Commerical EMS		Faster-than-Real-Time EMS （GEIRINA EMS Prototype）	
SCADA Sampling Rate	**5** s	SCADA Sampling Rate	**5** s
EMS Execution Cycle	**300 s**	EMS Execution Cycle	**5 s**
SE Execution Time	**~4490 ms**	SE Execution Time	**~200 ms**
PF Execution Time	**~3820 ms**	PF Execution Time	**~70 ms**
CA Execution Time	**~18000 ms**	CA Execution Time	**~1000 ms**

Source: Commerical EMS

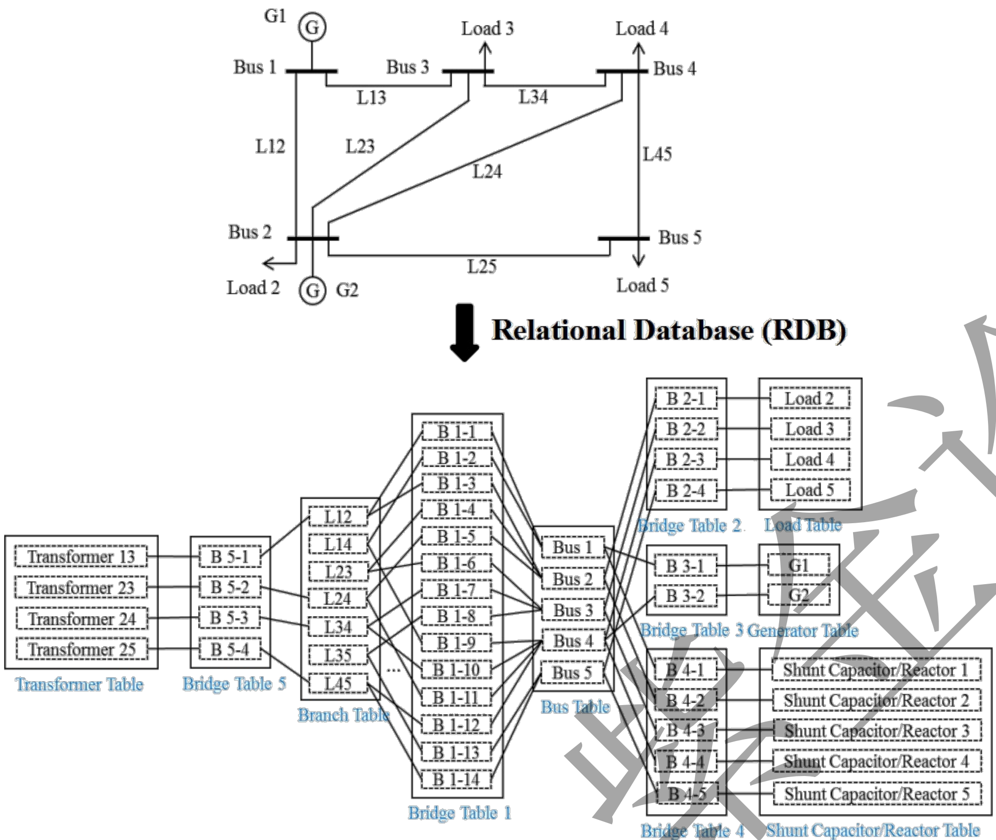
Source: GEIRINA EMS Prototype

Faster Than Real-time EMS

What We Achieved - Demo

Please note the running-time on the prototype is wall clock time incluing function calling time, execution time, and communication time.

RDB for Power System Modeling



Physical System

- ❑ Nodes are connected by edges
- ❑ Connectivity is naturally a graph

Relational Database

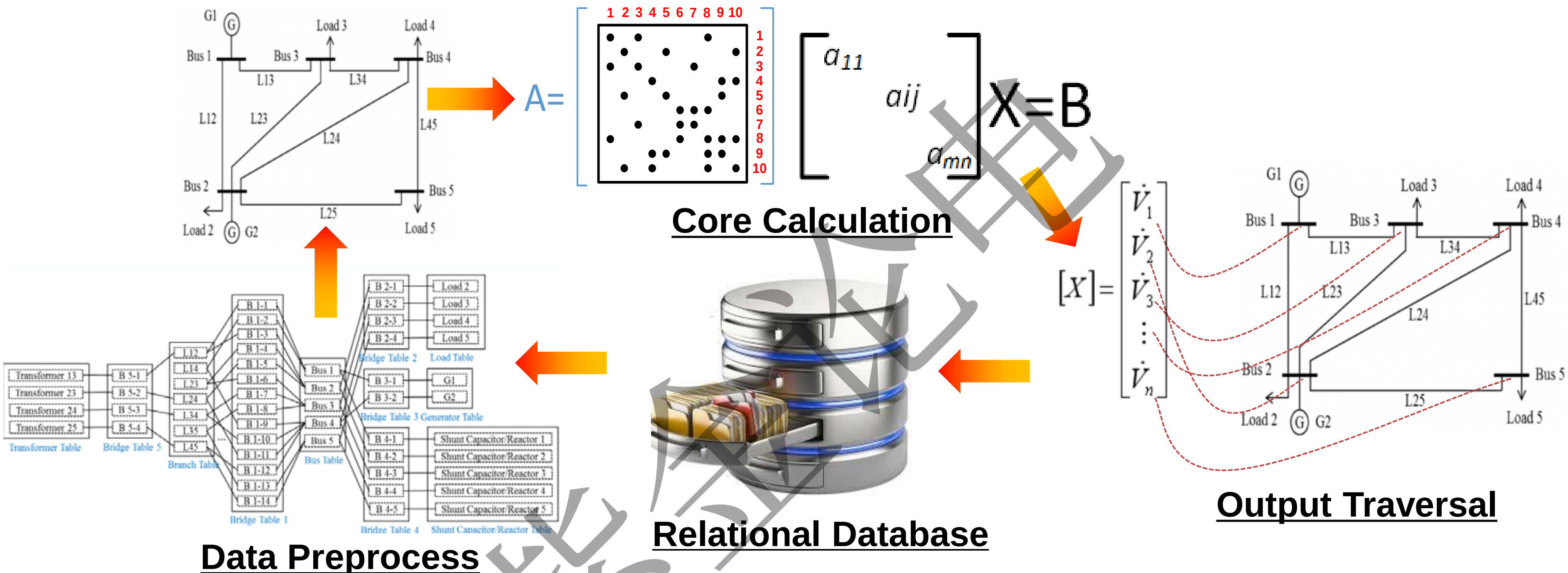
- ❑ Use table structure
- ❑ Not support unstructured data
- ❑ Attribute relations modelled by separated tables
- ❑ Use commonly shared key values to represent data relationships

Issues of Relational Database for Power System Modeling

- ❑ Join intensive queries for the whole database invite large computation time
- ❑ Maintain small portion of system requires multiple table update
- ❑ Time consuming to support recursive queries and parallel queries

RDB based Power System Computing

STATE GRID
GEIRI NORTH AMERICA
全球能源互联网美国研究院

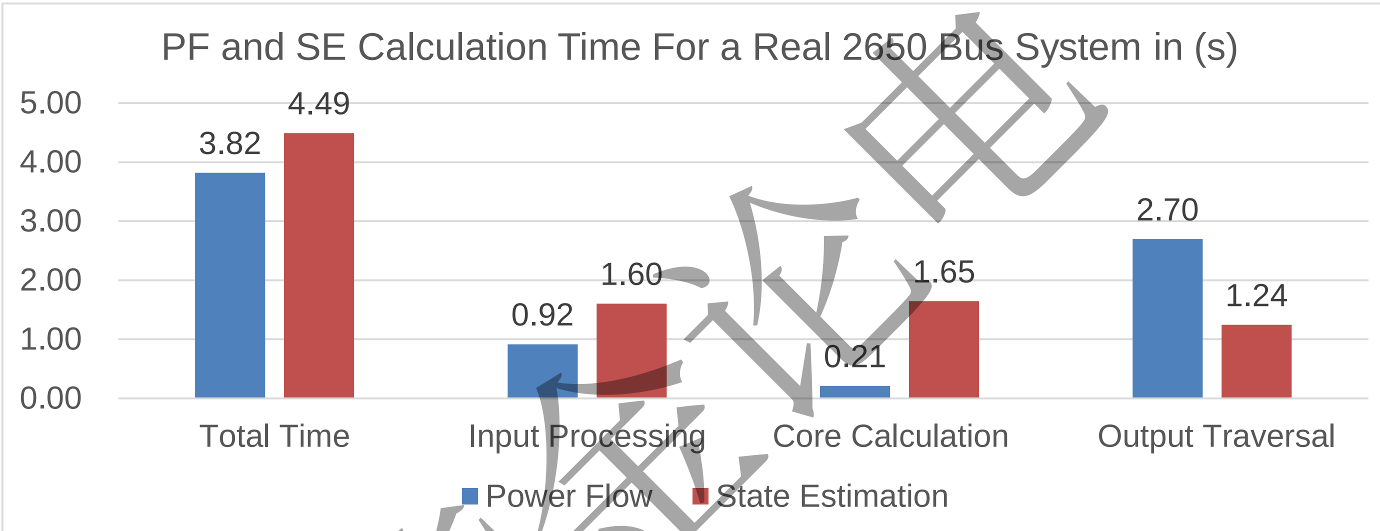


Issues of Relational Database for Power System Computing

- ❑ Need loop through branch table and bus table to create connectivity
- ❑ Complicated to support linear equation parallel computing
- ❑ Map solved variables to bus voltages and branch flows inviting time consuming output traversal

RDB Time Decomposition

A Real Case of 2650 bus system

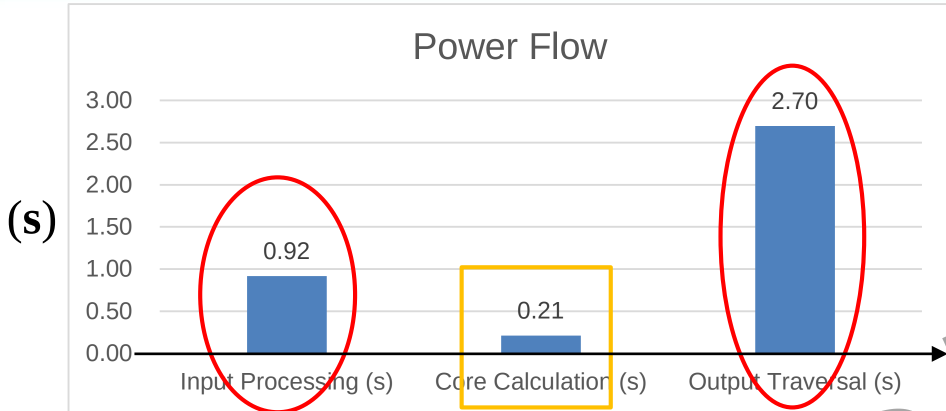


Data source: Beijing Kedong Electric Power Control System Co., LTD.

Observations:

- ❑ PF Data Input Processing and Output Traversal cost 94.5% of the total time
- ❑ SE Data Input Processing and Output Traversal cost 64.3% of the total time
- ❑ Need a new platform to integrate data management, calculation, and visualization

Graph Potentials
- Parallel Computing - PF

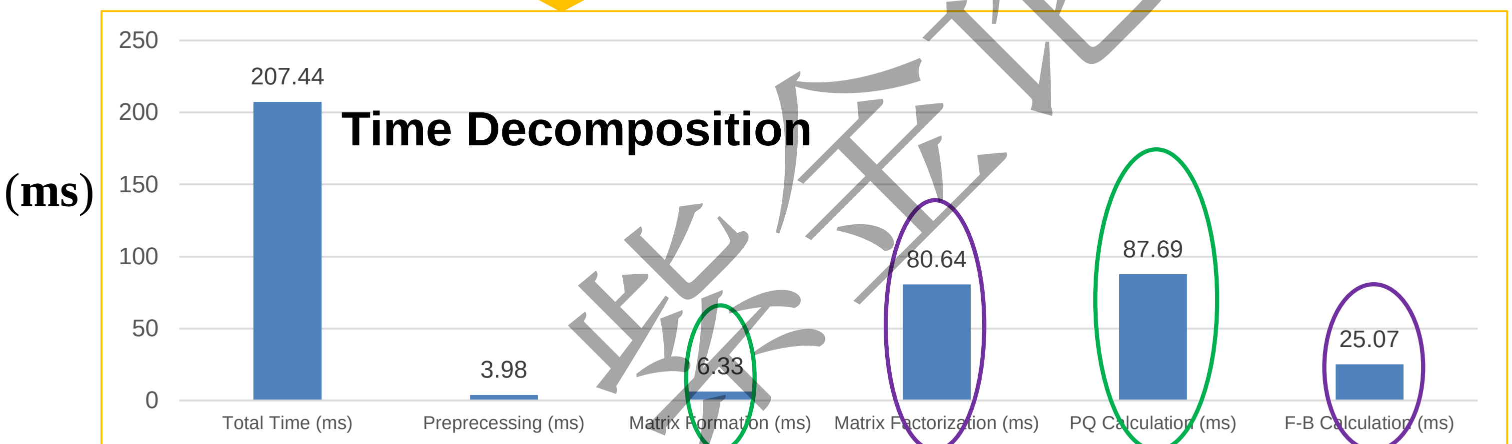


Using graph database model to save the time of

- ❑ **Input Processing**
- ❑ **Ouptut Traversal**

Further improve Core Calculation efficiency by

- ❑ **Graph based Nodal Parallel Computing**
- ❑ **Parallel LU Solver Hierarchical Parallel Computing**

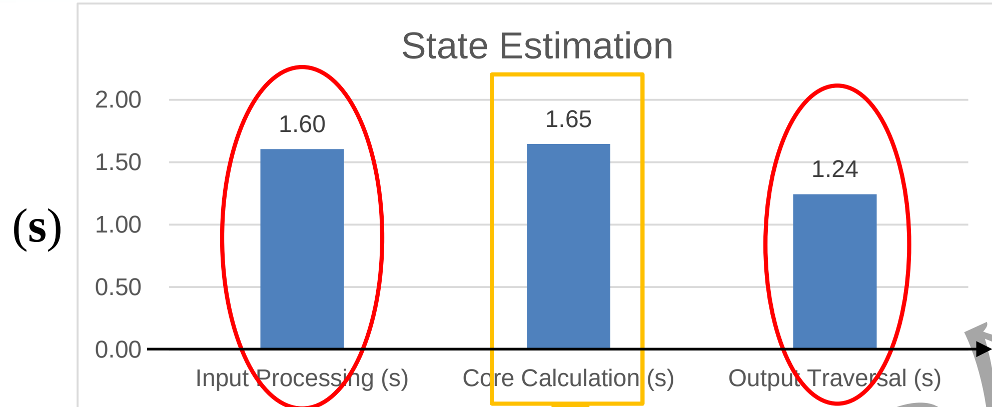


Observations:

- ❑ Matrix formation and P/Q calculation cost 45% of the total time which can be nodal parallelized.
- ❑ Matrix factorization and F/B substitution take 51% of the total time which can be hierarchical parallelized.

Graph Potentials
- Parallel Computing - SE

STATE GRID
GEIRI NORTH AMERICA
全球能源互联网美国研究院

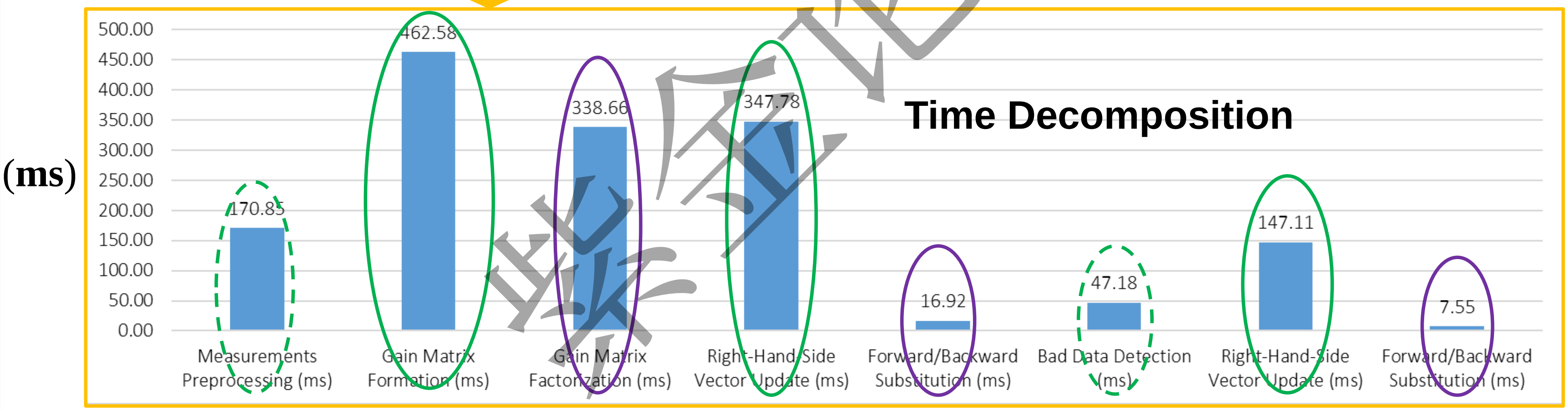


Using graph database model to save the time of

- ❑ **Input Processing**
- ❑ **Ouptut Traversal**

Further improve Core Calculation efficiency by

- ❑ **Graph based Nodal Parallel Computing**
- ❑ **Parallel LU Solver Hierarchical Parallel Computing**

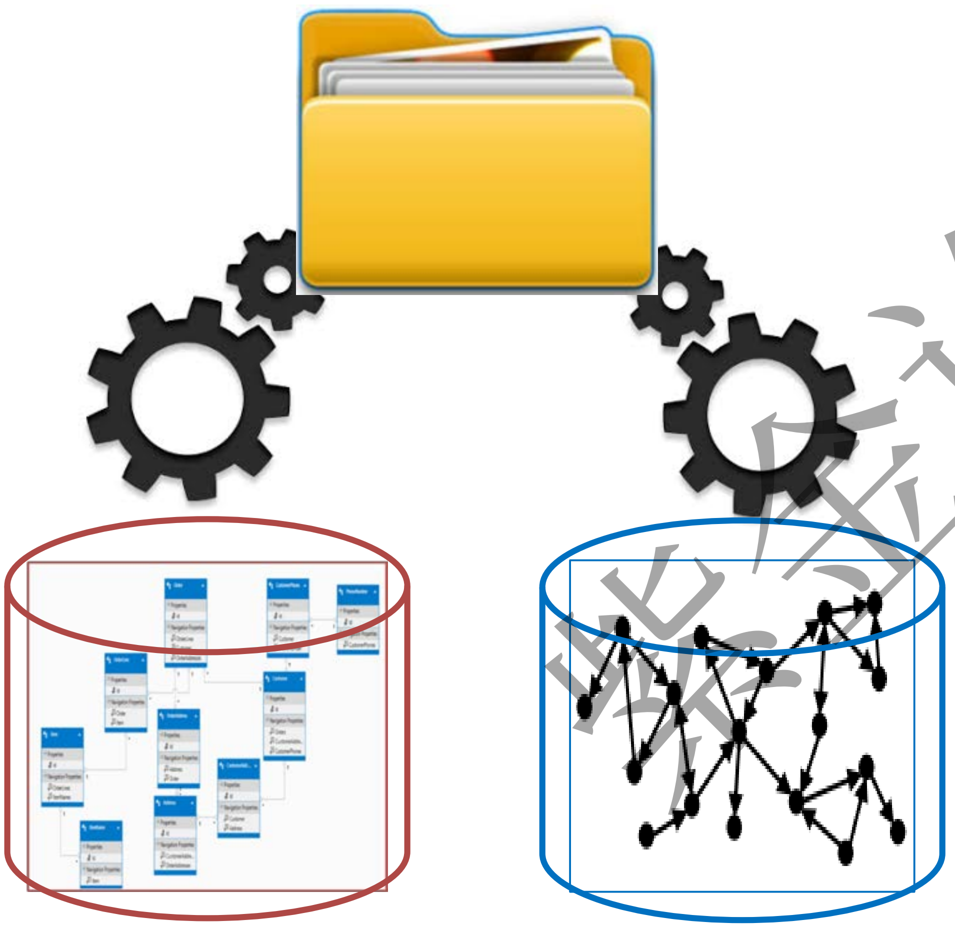


Observations:

- ❑ Gain matrix formulation and right-hand-side vector update take ~60% of core computation time which can be implemented by nodal parallel computing.
- ❑ Gain-matrix factorization and forward/backward substitution cost ~20% of time which can be hierarchically parallelized.

GDB for Power System Modeling

Original Data Document



Relational Database:
Data model is a collection of interlinked tables.

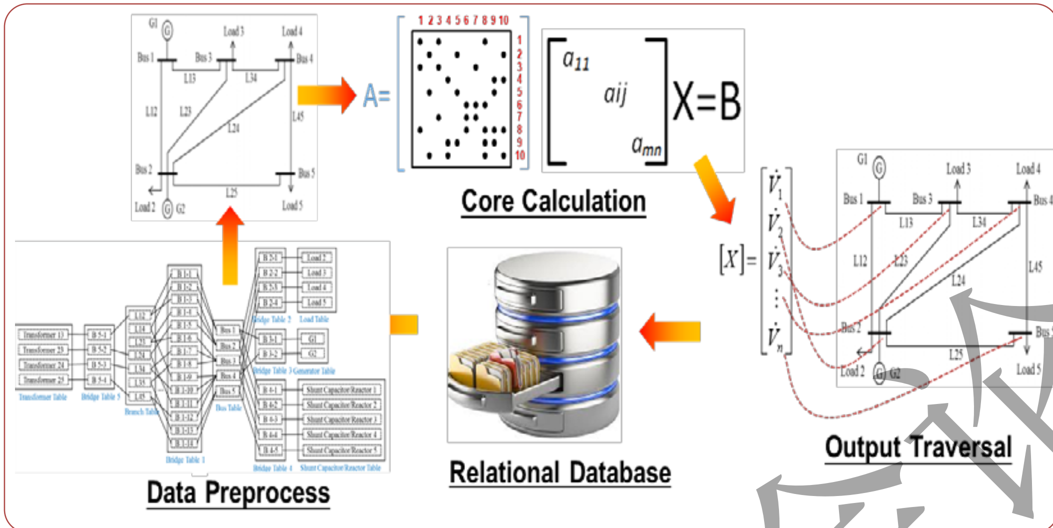
Graph Database:
Data model is a multi-relational graph.

Relational Database
- ❑ Use table structure
- ❑ Attribute relations modelled by separated tables
- ❑ Need to update multiple table to maintain system
- ❑ Hard to support recursive queries and parallel queries

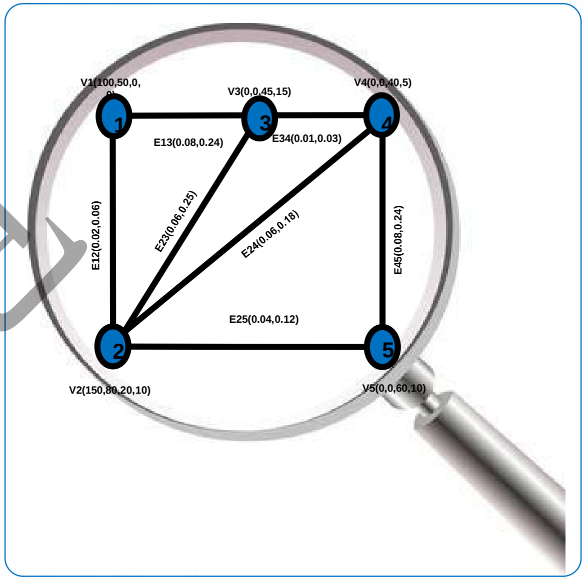
Physical System
- ❑ Edges are connecting by nodes
- ❑ Connectivity is naturally a graph

Graph Database
- ❑ Use graph structure with edges and nodes
- ❑ Store data by attributes of nodes and edges
- ❑ Support parallel computing
- ❑ Easy to maintain large system

GDB based Power System Computing



RDB Based Computing

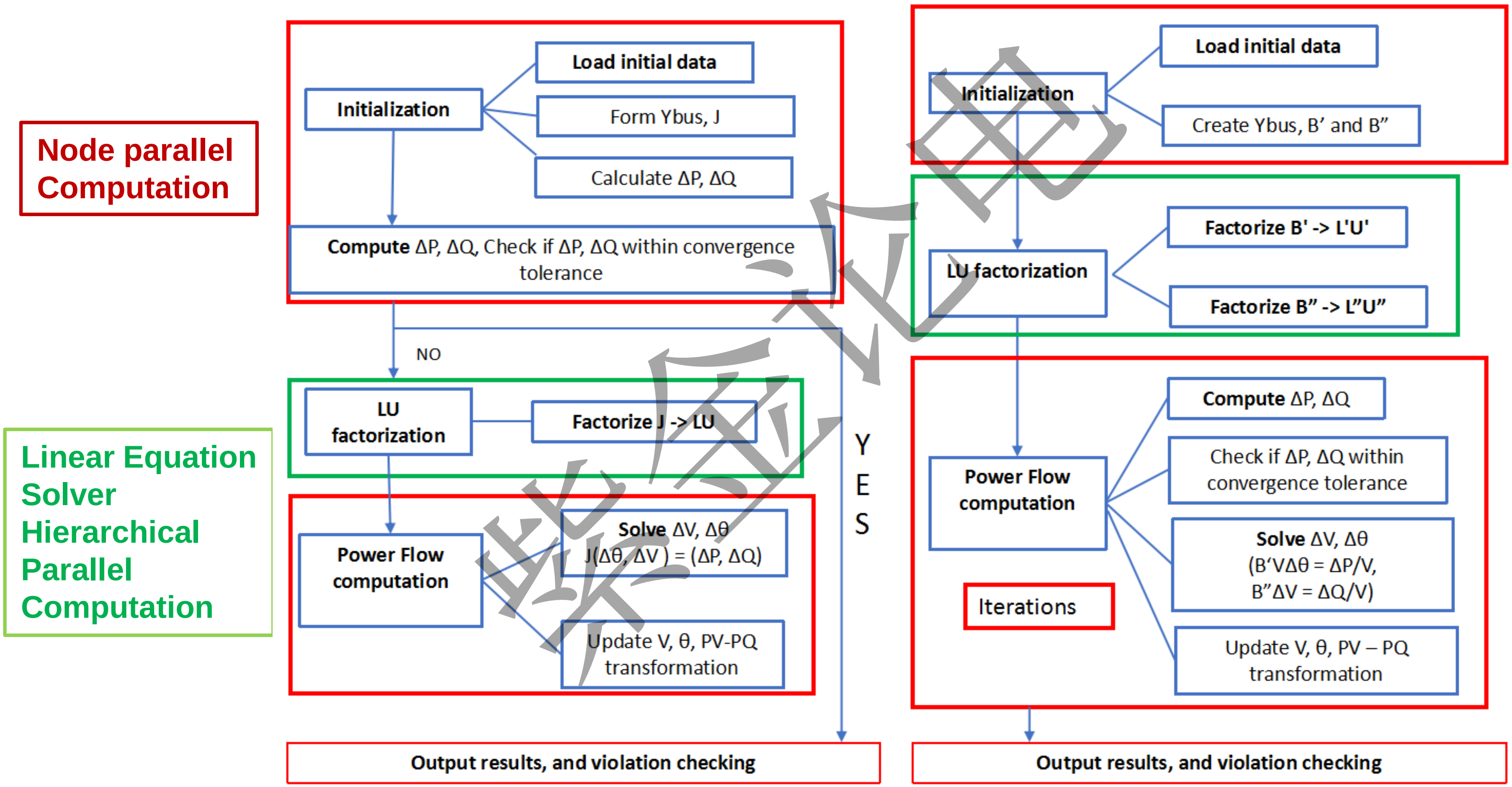


GDB Based Computing

Advantages of Graph Computing for Power System

- Integrate system modeling, core computing, and result visualization in graph database
- Data input preprocessing and result output traversal are not need
- Change calculation approach from serial computing to parallel queries
- Implement a suite of computation queries as library for power system applications

Graph Based N-R and F-D Power Flow Algorithms



Newton-Raphson **Fast Decoupled**

Graph Based State Estimation Algorithm

$$\begin{cases} G_{AA}\cdot\Delta\theta = {H_{AA}}^{T}R_A^{-1}(z_A - h_A(x)) \\ G_{RR}\cdot\Delta|V| = {H_{RR}}^{T}R_R^{-1}(z_R - h_R(x)) \end{cases}$$

$$\begin{cases} G_{AA} = {H_{AA}}^{T}R_A^{-1}H_{AA} = \sum_{i}^{n-1} {H_{AA,i}}^{T}\cdot {R_{A,i}}^{-1}\cdot H_{AA,i} = \sum_{i=1}^{n-1} G_{AA,i} \\ G_{RR} = {H_{RR}}^{T}R_R^{-1}H_{RR} = \sum_{i}^{n} {H_{RR,i}}^{T}\cdot {R_{R,i}}^{-1}\cdot H_{RR,i} = \sum_{i=1}^{n} G_{RR,i} \end{cases}$$

Page-Rank Node parallel Computation

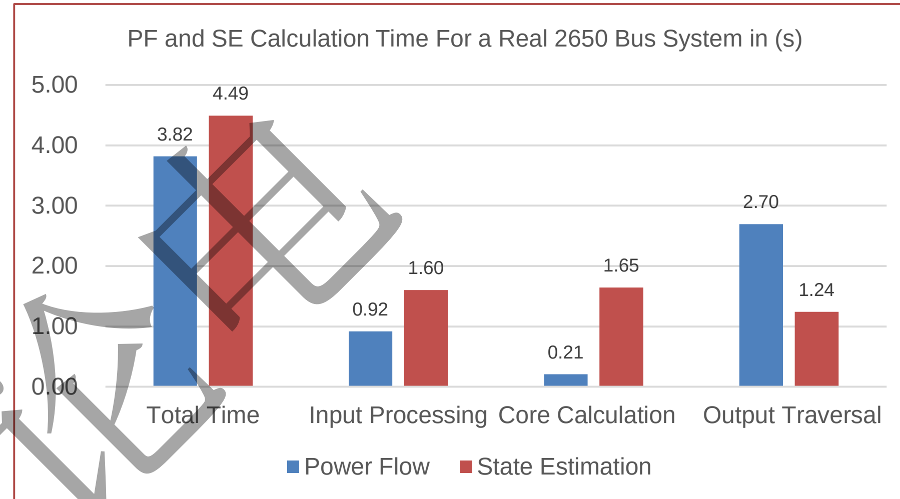
Linear Equation Solver Hierarchical Parallel Computation

1. Start Iterations, set iteration index $k = 0$;
2. Initialize the system state vector x^k, including θ^k and $|V|^k$ (flat start or not);
3. Formulate gain matrices, G_{AA} and G_{RR}, based on Page-Rank node parallel computing;
4. Decompose G_{AA} and G_{RR} using parallel LU solver;
5. Update right-hand-side vector ${H_{AA}}^{T}R_A^{-1}\left(z_A - h_A(x^k)\right)$ based on Page-Rank node parallel computing, solve $\Delta\theta^k$, and update $\theta^{k+1} = \theta^k + \Delta\theta^k$;
6. Check convergence: max $|\Delta x^k| \le \epsilon$? If yes, output θ^{k+1} and $|V|^k$; If no, go to step 7;
7. Update right-hand-side vector ${H_{RR}}^{T}R_R^{-1}\left(z_R - h_R(x^k)\right)$ based on Page-Rank node parallel computing, solve $\Delta|V|^k$, and update $|V|^{k+1} = |V|^k + \Delta|V|^k$;
8. Check convergence: max $|\Delta x^k| \le \epsilon$? If yes, output θ^{k+1} and $|V|^{k+1}$; If no, $k = k + 1$, go to step 5;

Graph Computing: A Real Case

Relational Database

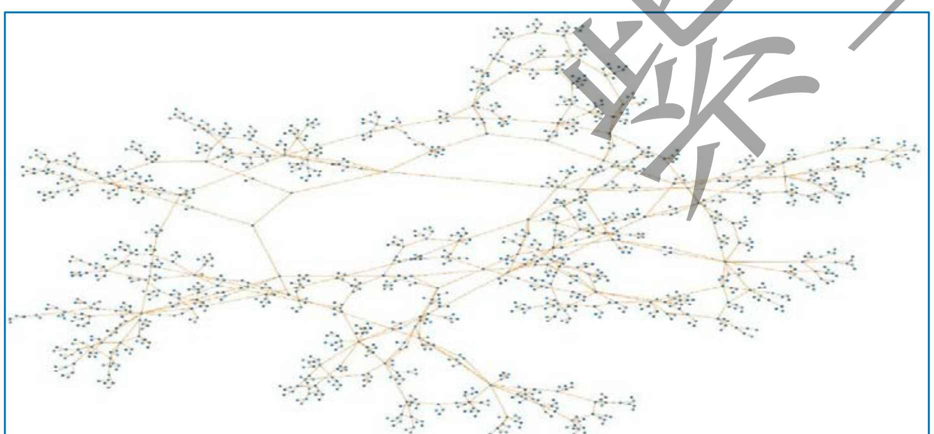
- ❑ Preprocessing: Search bus tables and branch tables to find connectivities.
- ❑ Output Traversal:Map solved variables to bus voltage and branch flow.



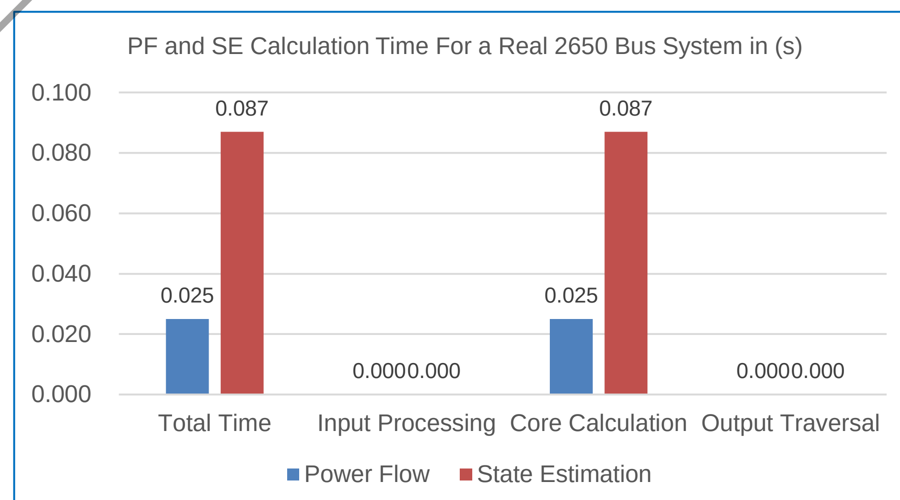
Relational Database

Graph Database

- ❑ **No preprocessing**. Connectivity is predefined in graph database.
- ❑ **No output traversal**. Solved bus voltage and branch flow are attributes of nodes and edges in graph database.



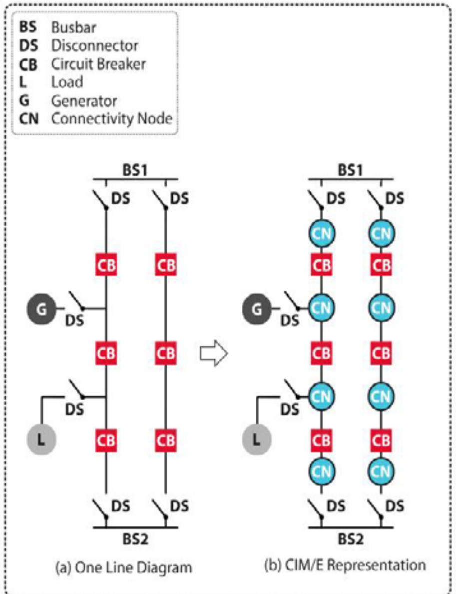
2650 Bus System Model in GDB



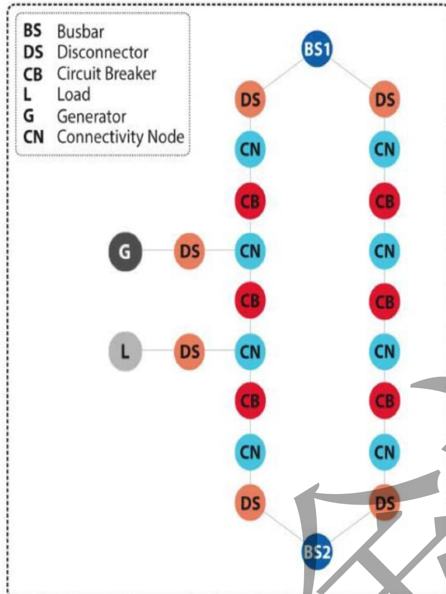
Graph Database and Computing

GDB for Node-Breaker Model

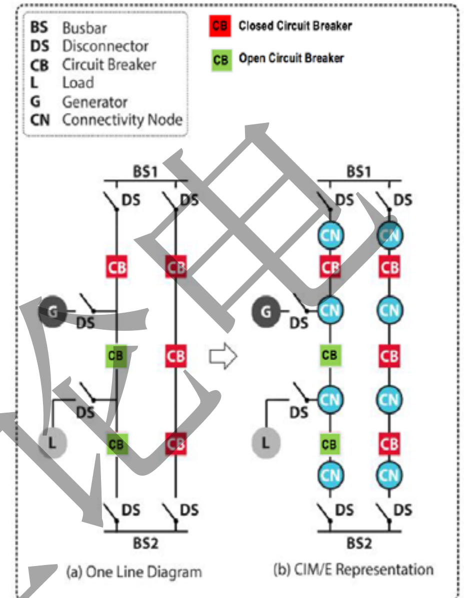
STATE GRID
GEIRI NORTH AMERICA
全球能源互联网美国研究院



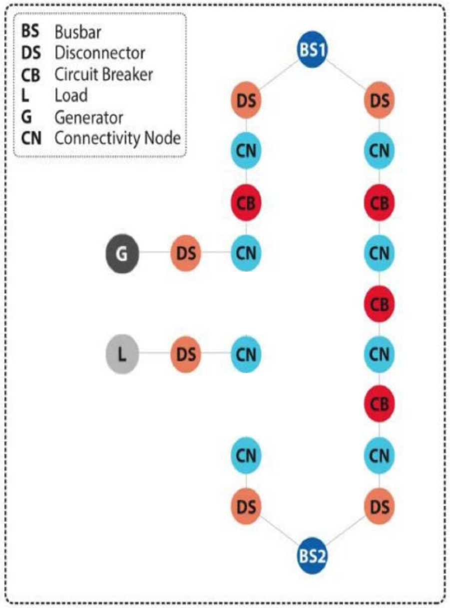
Substation representation in one-line diagram and CIM/E



Substation modeling in CIMGDB



Substation representation in one-line diagram and CIM/E



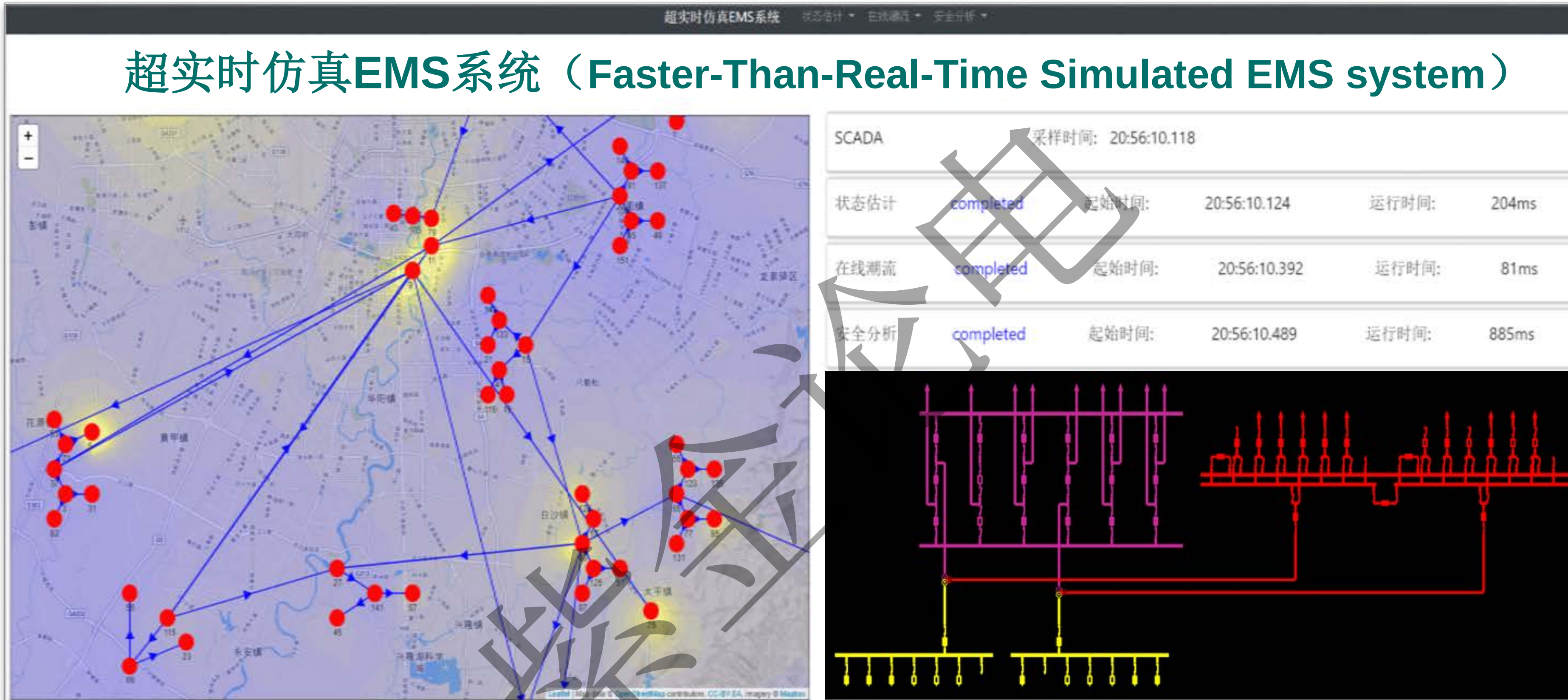
Substation modeling in CIMGDB

Node – Breaker Graph Model

- ❑ Based on Base-value, Substation, Bus, AC line, Unit, Transformer, Load, Compensator, Converter, DC line, Island, Topo-node, Breaker and Disconnector are modeled by vertices
- ❑ Common Information Model (CIM)

Graph Topology Processing Bridge Node-Breaker to Bus-Branch Model

Graph Based
Faster-Than-Real-Time EMS Platform



- ❑ Complete EMS applications at SCADA sampling rate (5 seconds)
- ❑ SE/PF/CA including topology processing is faster than real time, completed within 5 seconds
- ❑ Visualize applicatoin status, start time, and execution time
- ❑ Voltage heat map and operational serverity index show overall operational risk
- ❑ Substation diagram is automatically dynamically drawn and shows the detailed operations

Graph Based
Testing Results: Newton-Raphson PF

Testing System 1: 2650 Bus System (ms)

Server Configuration	Convergence Tolerance	# of Iteration	Segment Size	Threads
CPU: 2.0GHz, Mem: 64GB	0.05	5	8	12

Test Case	Initialization	Jacob Matrix Initialization	Order	Symbolic Factorization	Factorization and Iteration	Total
1	6.338	1.114	6.246	3.163	11.06	29.909
2	5.165	1.054	6.265	2.894	10.969	27.369
3	4.831	1.055	6.202	2.998	11.086	27.199
4	5.75	0.964	6.152	2.862	10.975	27.01
5	5.291	0.933	6.106	2.908	11.293	26.8
Average	5.475	1.024	6.194	2.965	11.077	27.657

Graph Based
Testing Results: Newton-Raphson PF

Testing System 2: MP10790 System (ms)

Server Configuration	Convergence Tolerance	# of Iteration	Segment Size	Running Threads
CPU: 2.0GHz, Mem: 64GB	0.05	4	10	12

Test Case	Initialization	Jacob Matrix Initialization	Order	Symbolic Factorization	Factorization and Iteration	Total
1	22.511	4.298	27.716	19.197	54.546	146.888
2	19.425	3.874	26.381	17.672	56.930	141.181
3	18.555	3.864	26.574	17.401	53.265	136.331
4	18.421	3.879	26.506	17.322	53.522	136.328
5	18.202	3.867	26.592	17.202	52.965	135.267
Average	19.423	3.956	26.754	17.759	54.246	139.199

Graph Based
Testing Results: Fast Decoupled PF

Testing System 1: 2650 Bus System (ms)

Server Configuration	Convergence Tolerance	# of Iteration	Segment Size	Threads
CPU: 2.0GHz, Mem: 64GB	0.05	5	8	12

Test	Initia.	B’ Matrix Factorization				B” matrix Factorization				Iteration	Total
		Order	SymFact	NumFact	SubTot	Order	SymFact	NumFact	SubTot		
1	7.059	2.457	1.106	0.699	4.262	2.177	1.203	0.585	3.965	6.421	25.848
2	6.256	2.427	1.114	0.635	4.176	2.109	1.262	0.57	3.941	6.441	24.898
3	6.505	2.499	1.157	0.756	4.412	2.187	1.322	0.642	4.151	6.736	25.589
4	5.976	2.406	1.107	0.647	4.187	2.116	1.207	0.603	3.926	6.469	24.461
5	5.825	2.392	1.053	0.597	4.042	2.138	1.267	0.547	3.952	6.535	24.050
Avg	6.324	2.436	1.107	0.666	4.216	2.145	1.252	0.589	3.987	6.520	24.969

Graph Based

Testing Results: Fast Decoupled PF

Testing System 2: MP10790 System (ms)

Server Configuration	Convergence Tolerance	# of Iteration	Segment Size	Running Threads
CPU: 2.0GHz, Mem: 64GB	0.05	5	10	12

Test	Initia.	B' Matrix Factorization				B" matrix Factorization				Iter	Total
		Order	SymFac	NumFac	SubTot	Order	SymFac	NumFac	SubTot		
1	23.446	11.417	6.665	4.482	22.564	8.962	4.807	2.475	16.244	14.144	89.266
2	22.745	11.531	6.580	4.545	22.656	8.998	4.875	2.508	16.341	14.260	88.356
3	22.256	11.470	6.426	4.497	22.393	8.960	4.644	2.556	16.160	14.086	87.092
4	22.197	11.495	6.288	4.489	22.272	8.918	4.711	2.505	16.134	14.341	86.867
5	21.916	11.292	6.308	4.485	22.085	8.953	4.584	2.529	16.066	14.335	84.903
Avg	22.512	11.441	6.453	4.499	22.394	8.938	4.724	2.514	16.189	14.233	87.297

Conclusions

- EMS cycle shall be speeded up as power system is significantly evolving to be larger and more complex with more power electronics, higher uncertainty, and faster events.
- Fast and predictive analytics are critical to respond the cascading events, avoid the severe blackouts, and enable the advanced system automatic control.
- High performance computing is critical to meet the gap on the pathway to EMS Robot.
- Graph database and graph parallel computing are promising to achieve Faster-than-Real-Time EMS.

Thank You!

Q&A