CREST 戦略的創造研究推進事業
Core Research for Evolutional Science and Technology

JST Japan Science and Technology Agency

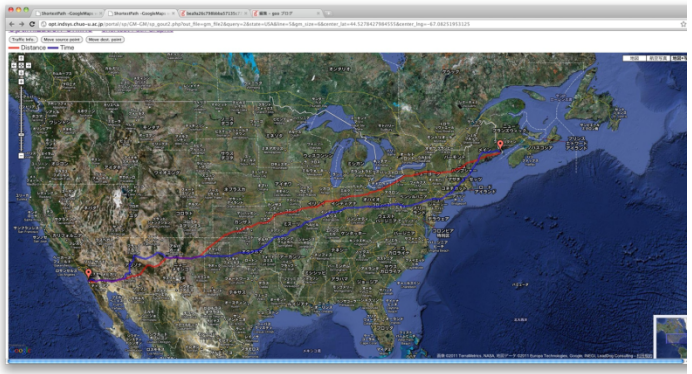
Advanced Computing and Optimization Infrastructure for Extremely Large-Scale Graphs on Post Peta-Scale Supercomputers

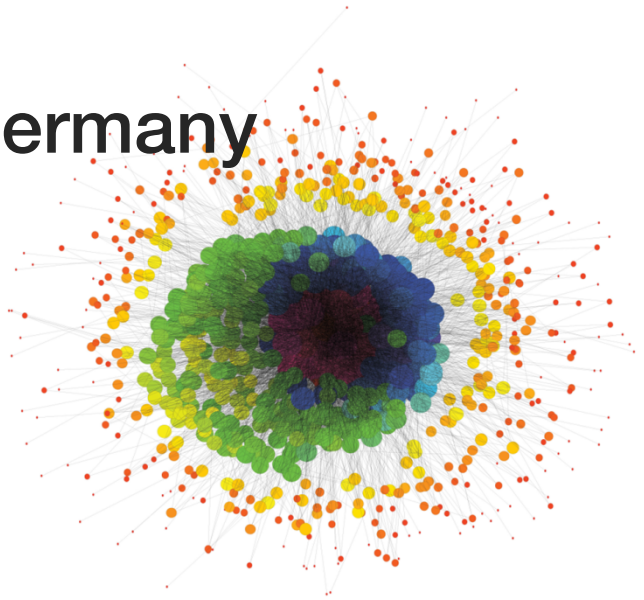
Katsuki Fujisawa

The Institute of Mathematics for Industry,
Kyushu University, Fukuoka, Japan

July 12, 2016

ICMS2016, ZIB, Berlin, Germany



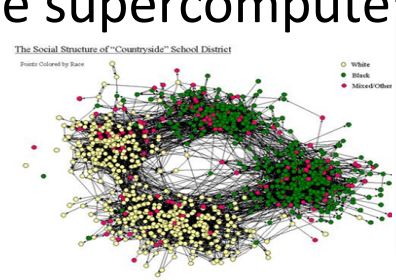


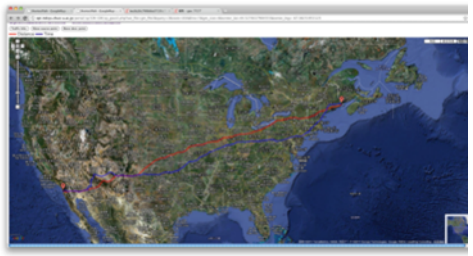
Advanced Computing and Optimization Infrastructure for Extremely Large-Scale Graphs on Post Peta-Scale Supercomputers

- **JST**(Japan Science and Technology Agency) **CREST**(Core Research for Evolutionally Science and Technology) **Project** (Oct, 2011 ～ March, 2017)
- **Winner of 8^{th} ,10^{th}, 11^{th} and 12^{th} Graph 500 benchmarks, and 1^{st} ～ 6^{th} Green Graph 500 benchmarks**
- **Collaborative research**

Panasonic NTT HITACHI Inspire the Next SUMITOMO ELECTRIC NEC sgi Hewlett Packard Enterprise

- **Innovative Algorithms and implementations**
  - Optimization, Searching, Clustering, Network flow, etc.
- Extreme Big Graph Data for emerging applications
  - **2^{30} ~ 2^{42} nodes** and **2^{40} ~ 2^{46} edges**
  - **Over 1M threads** are required for real-time analysis
- Many applications on post peta-scale supercomputers
  - Cyber security and social networks
  - Optimizing smart grid networks
  - Health care and medical science

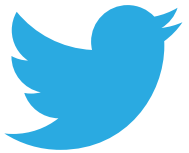




Background

- **The extremely large-scale graphs that have recently emerged in various application fields**
  - US Road network : 58 **million** edges
  - Twitter fellow-ship : 1.47 **billion** edges
  - Neuronal network : 100 **trillion** edges

Social network

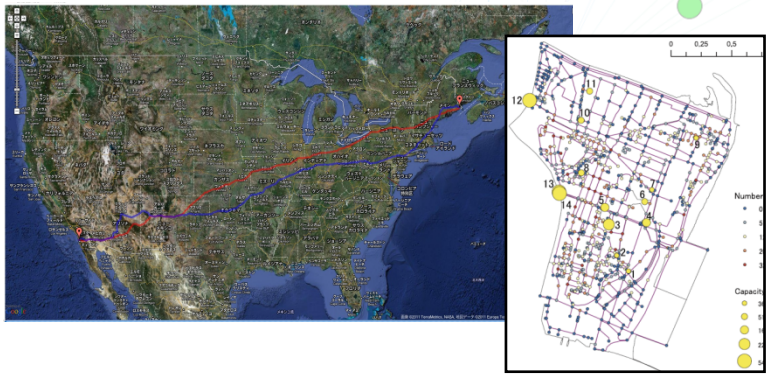


Twitter

61.6 million nodes & 1.47 billion edges

- **Fast and scalable graph processing** by using **HPC**

US road network
24 million nodes & 58 million edges



Cyber-security
15 billion log entries / day

Neuronal network
89 billion nodes & 100 trillion edges

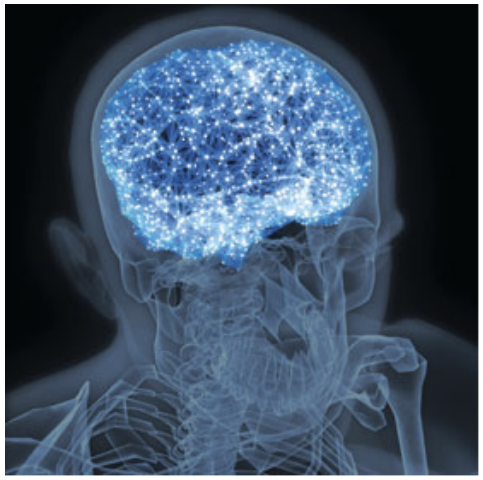
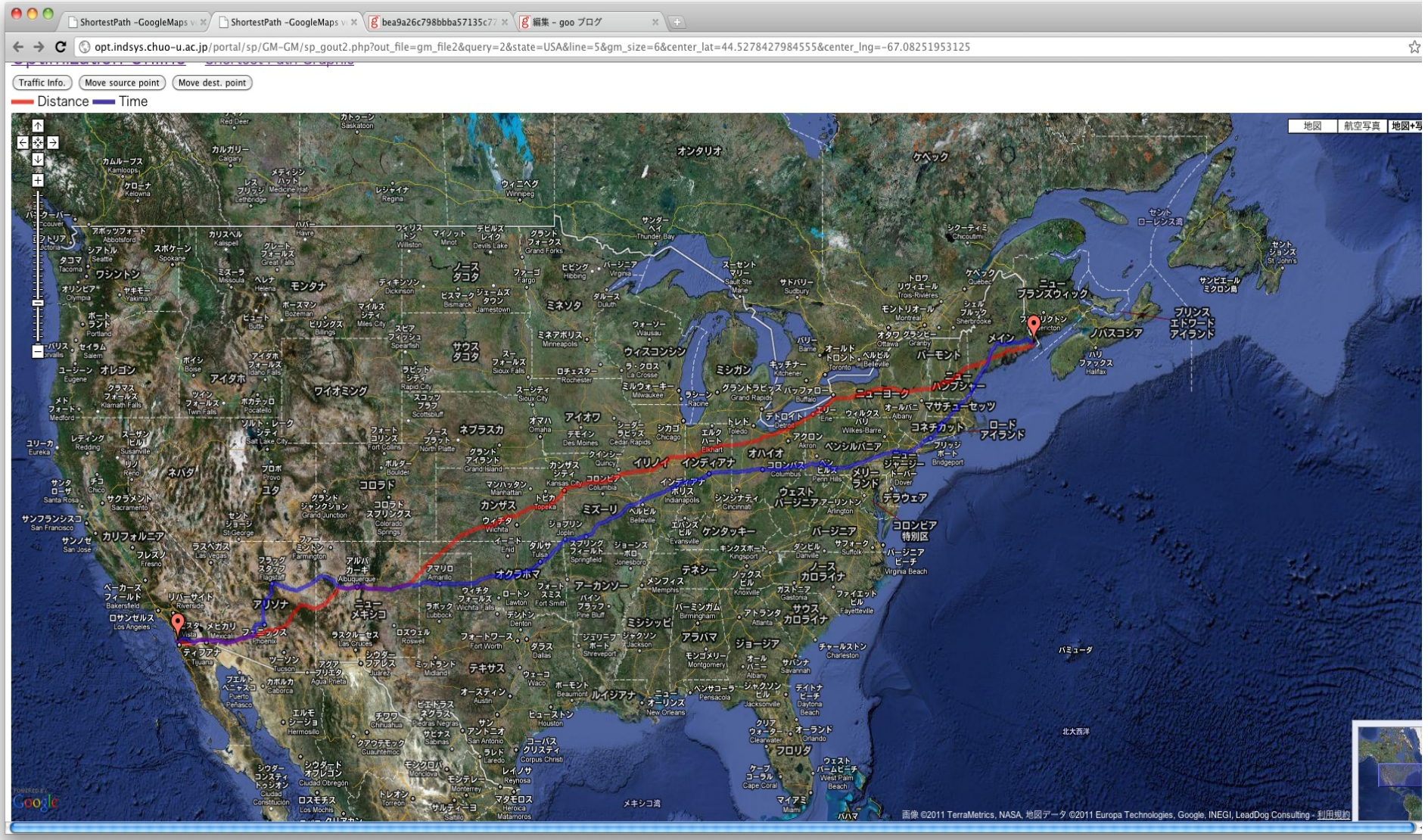


Image: Illustration by Mirko Ilic

USA road network : **24 million vertices & 58 million edges**

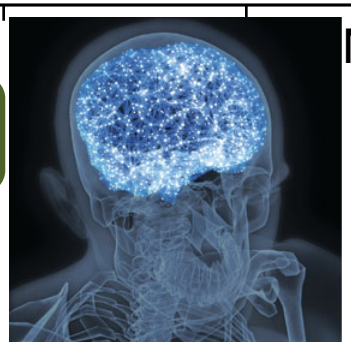
A shortest path from San Diego to Augusta

(**Red** : distance、**Blue** : time) : **http://opt.imi.kyushu-u.ac.jp/**



The size of graphs

K computer: 82944 nodes
Graph500: 38621 GTEPS



Neuronal Network

Symbolic Network

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

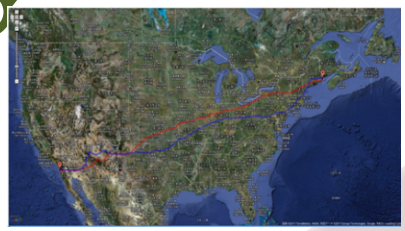
Graph500 (Small)

Graph500 (Mini)

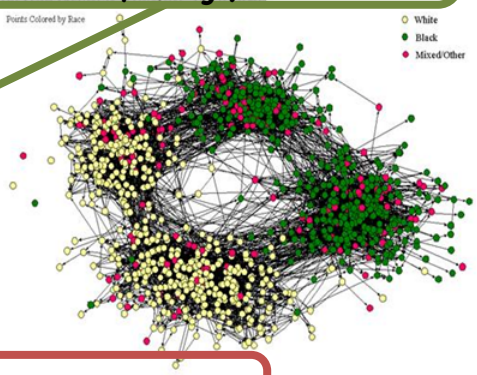
Graph500 (Toy)

1 trillion edges

1 billion edges



Twitter (tweets/day):2012



USA-road-d.USA.gr

USA Road Network

USA-road-d.LKS.gr

USA-road-d.NY.gr

1 billion nodes

1 trillion nodes

No. of edges

$\log_2(m)$

45
40
35
30
25
20

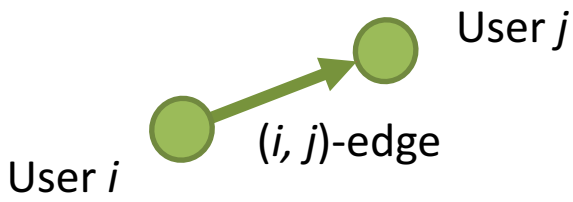
15
20
25
30
35
40
45

$\log_2(n)$

No. of nodes

Twitter network (Application of BFS)

Follow-ship network 2009



41 million vertices and 1.47 billion edges

Our NUMA-optimized BFS
on 4-way Xeon system

70 ms / BFS

⇒ 21.28 GTEPS

Six-degrees of separation

Frontier size in BFS

with source as User 21,804,357

Lv	Frontier size	Freq. (%)	Cum. Freq. (%)
0	1	0.00	0.00
1	7	0.00	0.00
2	6,188	0.01	0.01
3	510,515	1.23	1.24
4	29,526,508	70.89	72.13
5	11,314,238	27.16	99.29
6	282,456	0.68	99.97
7	11536	0.03	100.00
8	673	0.00	100.00
9	68	0.00	100.00
10	19	0.00	100.00
11	10	0.00	100.00
12	5	0.00	100.00
13	2	0.00	100.00
14	2	0.00	100.00
15	2	0.00	100.00
Total	41,652,230	100.00	-

Graph 500 and Green Graph500 Benchmarks

- **New Graph Search Based Benchmarks for Ranking Supercomputers**
- BFS (Breadth First Search) from a single node on a static, undirected Kronecker graph with average vertex degree edgegactor (=16).
- No. of Nodes = 2^{SCALE}, Average degree = 16
- Performance Metrics :

  TEPS(Traversed Edges per Second) : Graph 500

  TEPS/W (Traversed Edges per Second / Watt) : Green Graph500

Step.

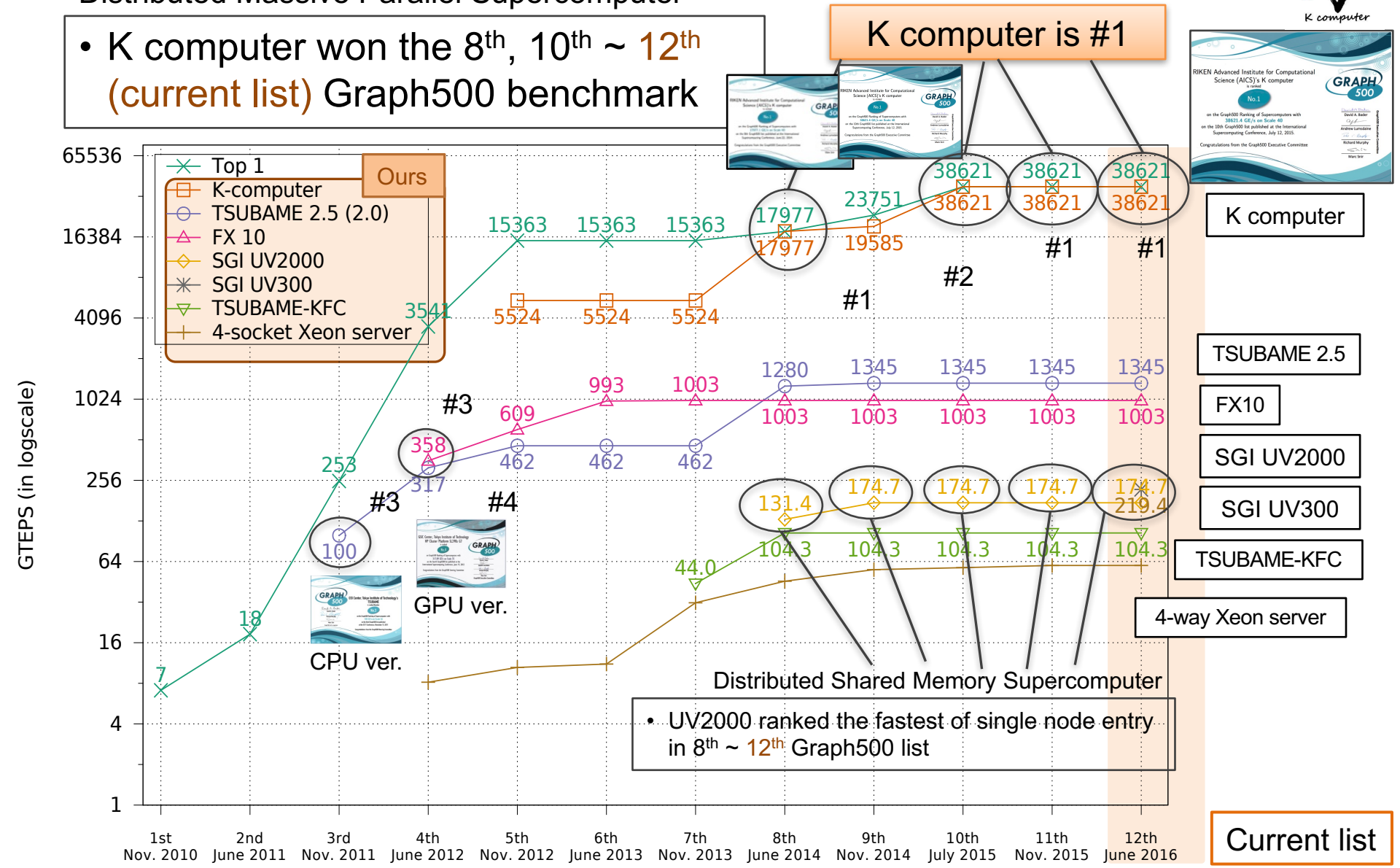
1. Generate edgelist
2. Construct Graph（CSR format）
3. BFS
4. Validation

} Repeat 64 times for randomly selected source vertices

Our achievements in Graph500

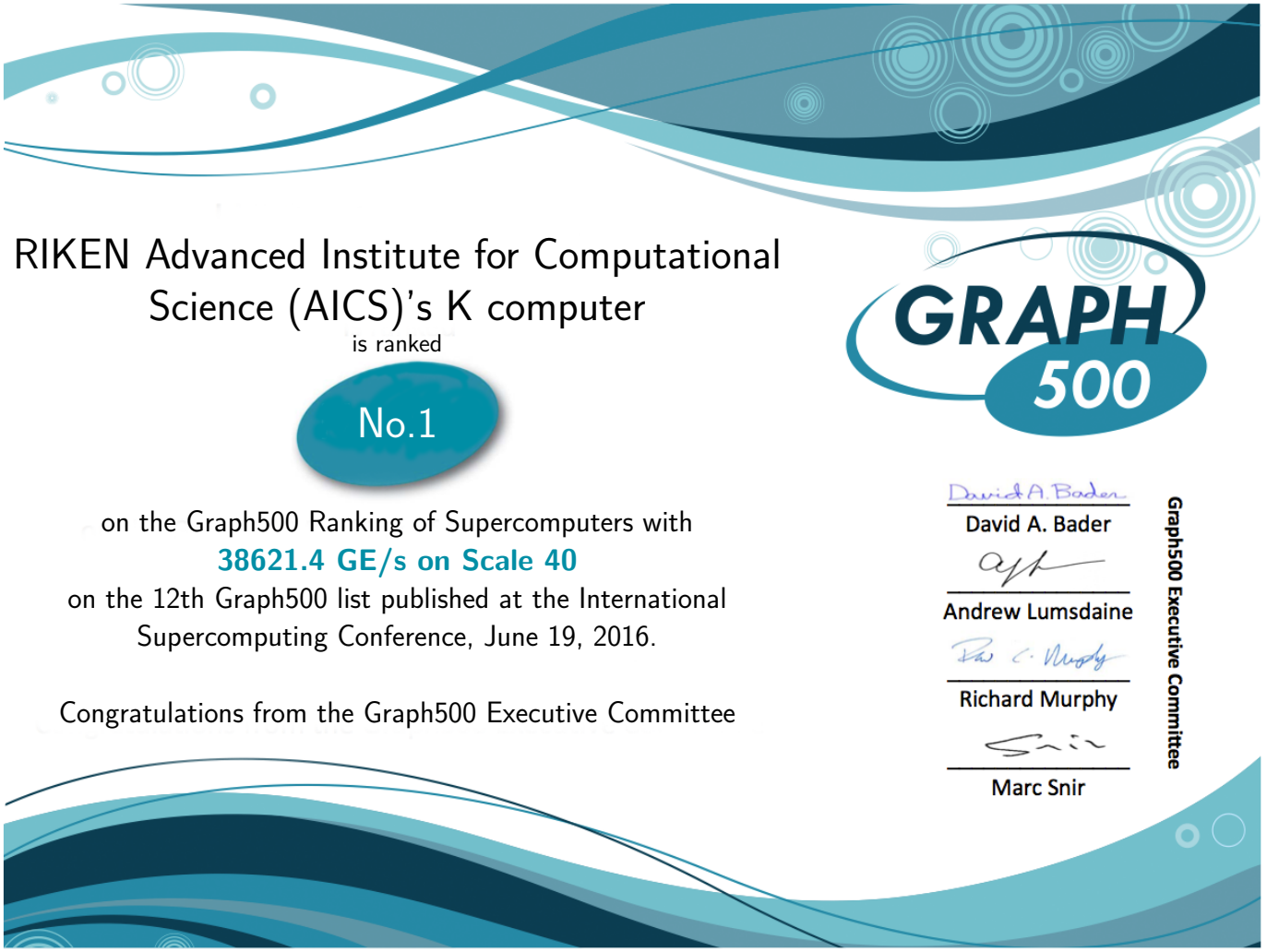
Distributed Massive Parallel Supercomputer

- K computer won the 8th, 10th ~ 12th (current list) Graph500 benchmark



The 10th, 11th and 12th Graph500 Lists : The winner is K computer

Scale 40 : 38621.4 GTEPS (July, November 2015 and June 2016)



The 10th,11th and 12th Graph500 Lists : The winner is K computer

Scale 40 : 38621.4 GTEPS

========== Result ==========

SCALE: 40

edgefactor: 16

NBFS: 64

graph_generation: 197.379

num_mpi_processes: 82944

construction_time: 609.395

min_time: 0.395321105141

firstquartile_time: 0.409624118358

median_time: 0.455501377815

thirdquartile_time: 0.566996186739

max_time: 1.95167534612

mean_time: 0.562004323256

stddev_time: 0.311699826145

min_nedge: 1.7592103987e+13

firstquartile_nedge: 1.7592103987e+13

median_nedge: 1.7592103987e+13

thirdquartile_nedge: 1.7592103987e+13

max_nedge: 1.7592103987e+13

mean_nedge: 1.7592103987e+13

stddev_nedge: 0

min_TEPS: 9.01384752422e+12

firstquartile_TEPS: 3.10268470903e+13

median_TEPS: 3.86214067477e+13

thirdquartile_TEPS: 4.29469437914e+13

max_TEPS: 4.45007963348e+13

harmonic_mean_TEPS: 3.1302435335e+13

harmonic_stddev_TEPS: 2.18728188393e+12

min_validate: 43.201660905

firstquartile_validate: 43.4925568579

median_validate: 44.3293765394

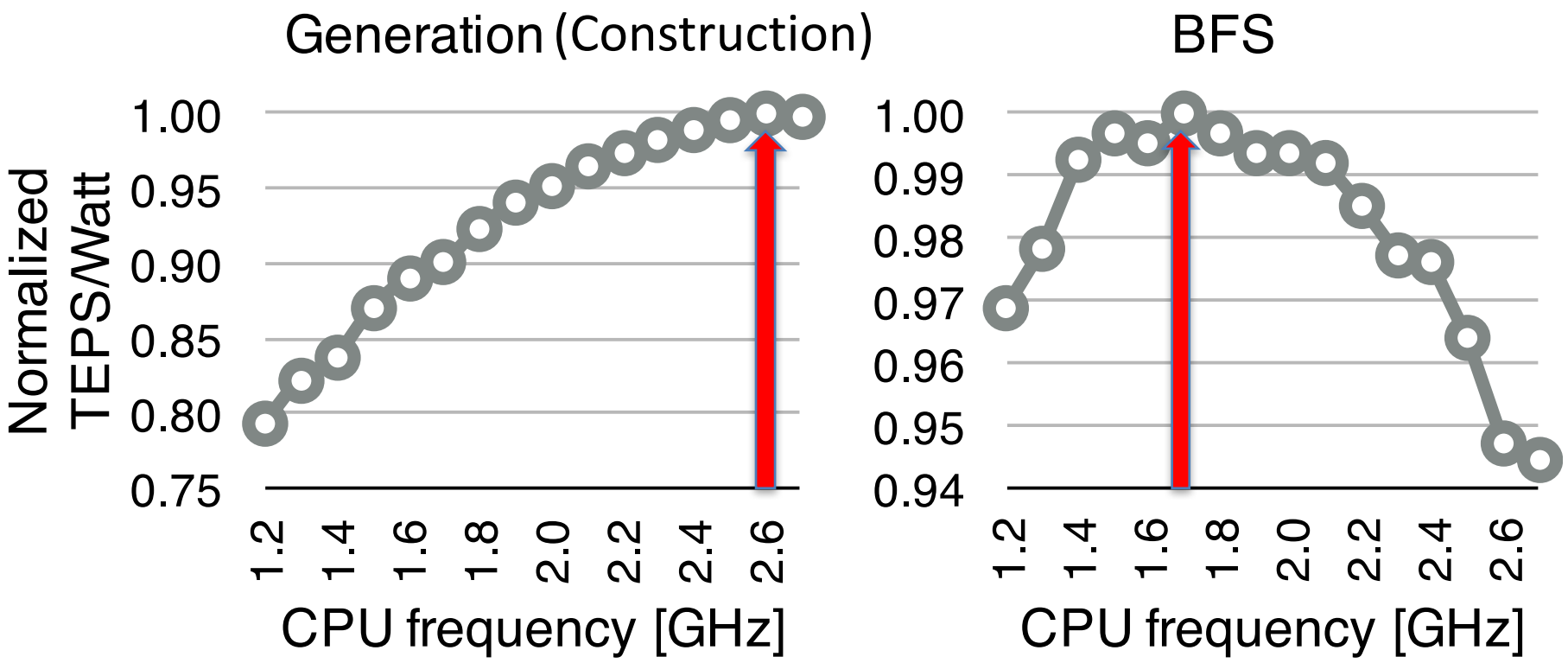
thirdquartile_validate: 45.4055157886

max_validate: 50.040661654

mean_validate: 44.6539914012

stddev_validate: 1.39768976422

Power efficiency evaluation of two components of the multi-nodeGraph500 implementation with varying CPU frequency



TOP500 List : June 2016

TOP 10 Sites for June 2016

For more information about the sites and systems in the list, click on the links or view the complete list.

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	National Supercomputing Center in Wuxi China	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC	10,649,600	93,014.6	125,435.9	15,371
2	National Super Computer Center in Guangzhou China	**Tianhe-2 (MilkyWay-2)** - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
3	DOE/SC/Oak Ridge National Laboratory United States	**Titan** - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
4	DOE/NNSA/LLNL United States	**Sequoia** - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890
5	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660

Graph500 List : June 2016

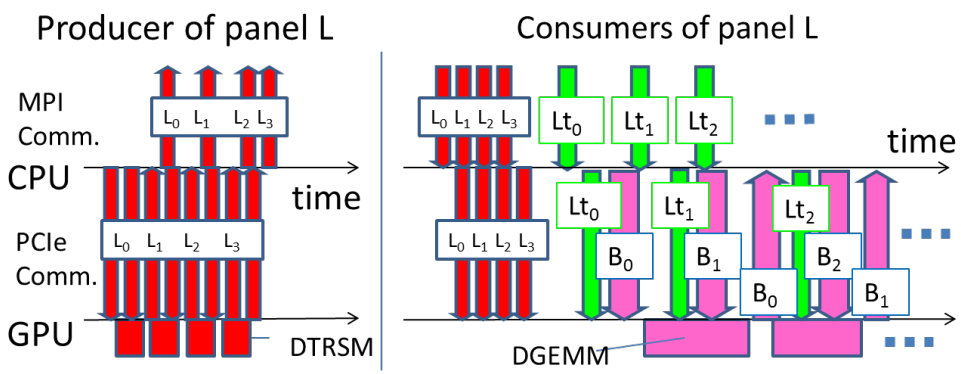
June 2016

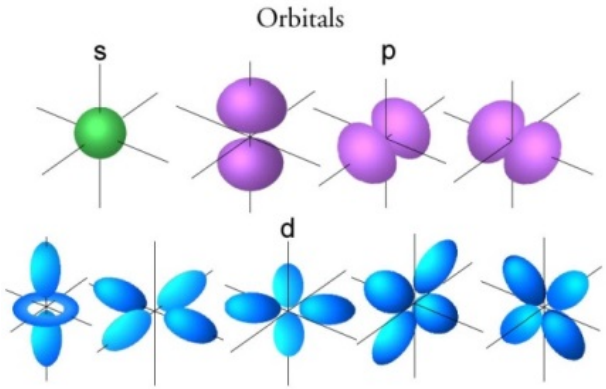
No.	Rank	Machine	Installation Site	Number of nodes	Number of cores	Problem scale	GTEPS
1	1	K computer (Fujitsu - Custom)	RIKEN Advanced Institute for Computational Science (AICS)	82944	663552	40	38621.4
2	2	Sunway TaihuLight (NRCPC - Sunway MPP)	National Supercomputing Center in Wuxi	40768	10599680	40	23755.7
3	3	DOE/NNSA/LLNL Sequoia (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	Lawrence Livermore National Laboratory	98304	1572864	41	23751
4	4	DOE/SC/Argonne National Laboratory Mira (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	Argonne National Laboratory	49152	786432	40	14982
5	5	JUQUEEN (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	Forschungszentrum Juelich (FZJ)	16384	262144	38	5848

CREST 戦略的創造研究推進事業
Core Research for Evolutional Science and Technology
JST Japan Science and Technology Agency

Petascale General Solver for Semidefinite Programming Problems with over Two Million Constraints

Katsuki Fujisawa & Yuichiro Yasui & Hayato Waki
Kyushu University, Japan
Toshio Endo & Hitoshi Sato & Naoki Matsuzawa & Satoshi Matsuoka
Tokyo Institute of Technology, Japan





SDP(SemiDefinite Programming)

Primal

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{A}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{A}_p \bullet \boldsymbol{X} = b_p \ (p = 1, 2, \ldots, m), \ \boldsymbol{X} \in \mathcal{S}^n_+ \end{array}\right\} \quad (1)$$

Dual

$$\left.\begin{array}{ll} \text{maximize} & \displaystyle\sum_{p=1}^{m} b_p z_p \\ \text{subject to} & \displaystyle\sum_{p=1}^{m} \boldsymbol{A}_p z_p + \boldsymbol{Y} = \boldsymbol{A}_0, \ \boldsymbol{Y} \in \mathcal{S}^n_+ \end{array}\right\} \quad (2)$$

$\mathcal{S}^n$: $n \times n$ symmetric matrices

$\boldsymbol{X} \bullet \boldsymbol{Y} = \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij} Y_{ij}$

$\boldsymbol{X} \in \mathcal{S}^n_+$ $(\mathcal{S}^n_{++})$: $\boldsymbol{X} \in \mathcal{S}^n$ is positive semidefinite (or positive definite)

$\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 0, 1, \ldots, m)$ and $\boldsymbol{b} \in \mathbb{R}^m$

SDPA 1.x(1995) could solve SDPs with n = m = 20 ~30 (very small size!!)

Primal-Dual Interior-Point Methods

1. Choose an initial point $(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{z})$
   with $\boldsymbol{X} \succ \boldsymbol{O}, \boldsymbol{Y} \succ \boldsymbol{O}$.
2. Compute Search Direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z})$.
3. Compute Step Length α_p, α_d.
   $\boldsymbol{X} + \alpha_p d\boldsymbol{X} \succ \boldsymbol{O}, \boldsymbol{Y} + \alpha_d d\boldsymbol{Y} \succ \boldsymbol{O}$
4. Update $(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{z})$
   $\leftarrow (\boldsymbol{X} + \alpha_p d\boldsymbol{X}, \boldsymbol{Y} + \alpha_d d\boldsymbol{Y}, \boldsymbol{z} + \alpha_d d\boldsymbol{z})$.
5. Goto 2 if $(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{z})$ is not close to optimal.

Step 2 : Compute the search direction $(d\boldsymbol{x}, d\boldsymbol{X}, d\boldsymbol{Y})$.

Step 2a Compute the SCM $\boldsymbol{B}$ using the formula

$$B_{ij} = \left((\boldsymbol{X}^s)^{-1} \boldsymbol{F}_i \boldsymbol{Y}^s \right) \bullet \boldsymbol{F}_j. \qquad \textbf{ELEMENTS}$$

Step 2b Apply Cholesky factorization to $\boldsymbol{B}$ and obtain a lower triangular matrix $\boldsymbol{L}$ such that $\boldsymbol{B} = \boldsymbol{L}\boldsymbol{L}^T$.

CHOLESKY

The major bottleneck parts (80% - 90% of total execution time)

- **ELEMENTS** : Computation of the SCM
  - ✓ Memory Access-intensive
  - ✓ Time-complexity: $O(mn^3 + m^2n^2)$

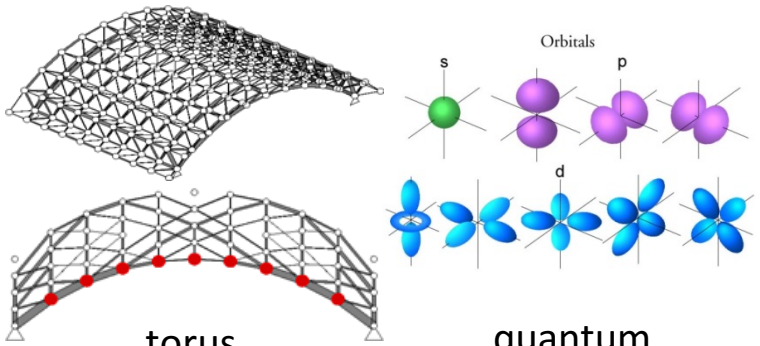
n : matrix size
m: # of constraints

or

- **CHOLESKY** : Cholesky factorizations of the SCM
  - ✓ Compute-intensive
  - ✓ Time-complexity: $O(m^3)$

ELEMENTS-bound SDP problems

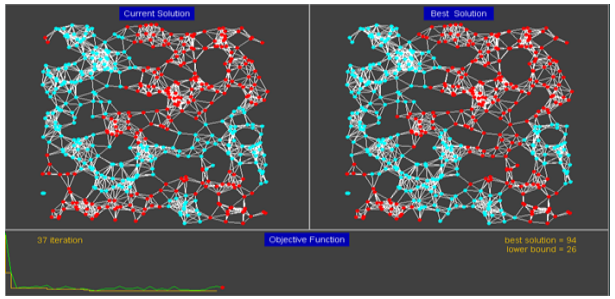
m < n (not m >> n), and **Fully Dense SCM**



torus quantum

CHOLESKY-bound SDP problems

m >> n, and **Fully Dense SCM**



Comb. Opt. Quad. assignment

Fast computation of sparse SCM is future work.

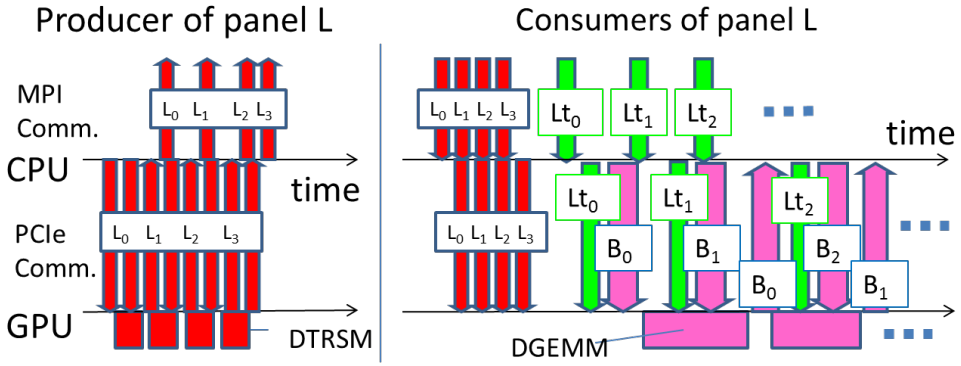
e.g.) The sensor network location problem and The polynomial optimization problem

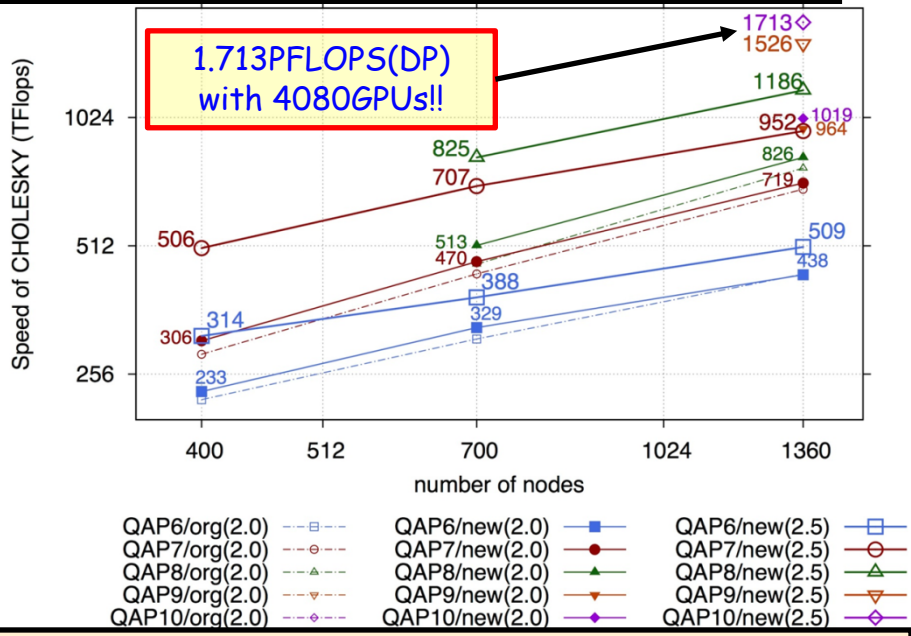
High-Performance General Solver for Extremely Large-scale Semidefinite Programming Problems

1. Mathematical Programming : one of the most important mathematical programming
2. Many Applications : combinatorial optimization, control theory, structural optimization, quantum chemistry, sensor network location, data mining, etc.

Parallel Algorithm of Cholesky Factorization

GPU computation, PCI-e communication, and
MPI communication are overlapped





- SDPARA is a parallel implementation of the interior-point method for Semidefinite Programming
  Parallel computation for two major bottlenecks
  - ELEMENTS ⇒ Computation of Schur complement matrix (SCM)
  - CHOLESKY ⇒ Cholesky factorization of Schur complement matrix (SCM)
- SDPARA could attain high scalability using 16,320 CPU cores on the TSUBAME 2.5 supercomputer and some techniques of processor affinity and memory interleaving when the computation of SCM (ELEMENTS) constituted a bottleneck.
- With 4,080 NVIDIA GPUs on the TSUBAME 2.0 & 2.5 supercomputer, our implementation achieved 1.019 PFlops(TSUBAME 2.0) & 1.774PFlops(TSUBAME 2.5: TSUBAME Grand Challenge 2015 Autumn) in double precision for a large-scale problem (CHOLESKY) with over two million constraints.

Table 1. Performance record of CHOLESKY of SDPARA

Year	Paper	n	m	CHOLESY (Flops)
2003	[7]	630	24,503	78.58 Giga
2010	[8]	10,462	76,554	2.414 Tera
2012	[9]	1,779,204	1,484,406	0.533 Peta
2014	[11]	2,752,649	2,339,331	1.713 Peta
2015	[23]	2,322,988	1,962,225	1.774 Peta

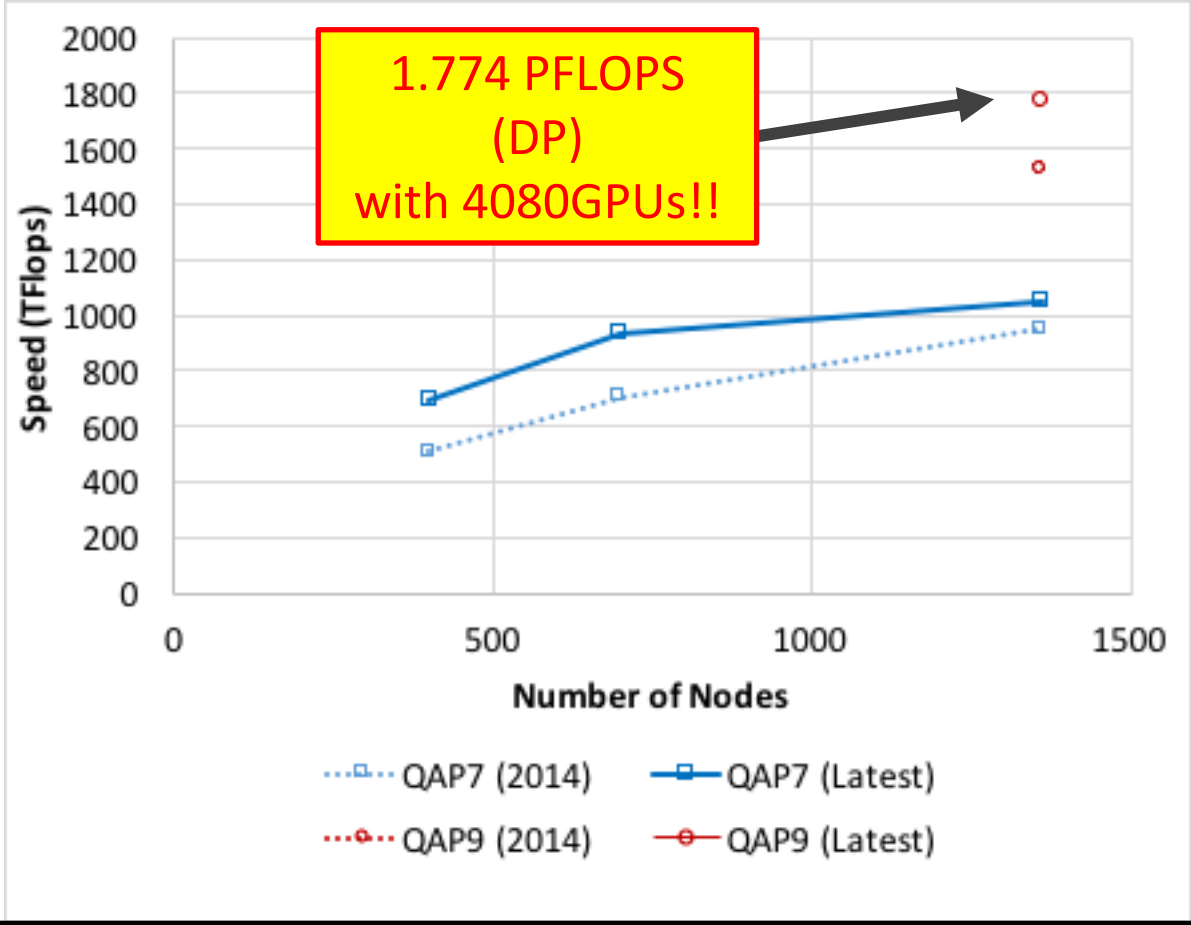
Performance of CHOLESKY for QAP on TSUBAME 2.5

Yuki Tsujita, Toshio Endo, Katsuki Fujisawa, (ESPM2, 2015)

Design Strategy

We decompose the whole computation into fine-grained tasks. Tasks are distributed among processes and executed by our data driven scheduler.

- **Memory hierarchy** is considered to support larger problems than aggregated GPU memory capacity
- **Locality-aware scheduling**
- **Scalable data transfer**, which is not supported by other schedulers



SDPARA can solve the largest SDP problem

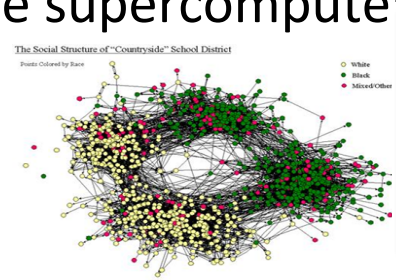
- DNN relaxation problem for QAP10 QAPLIB with 1.96 million constraints
- Using 1360 nodes, 2720 CPUs, 4080 K20X GPUs
- 1.774 PFLOPS in CHOLESKY (1.96m x 1.96m)
- The fastest and largest result as mathematical optimization problems!!

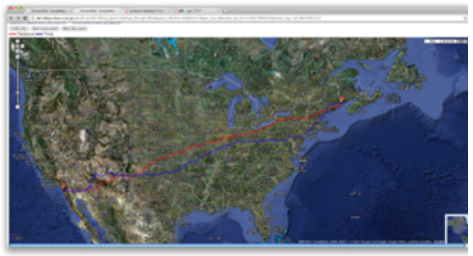
Advanced Computing and Optimization Infrastructure for Extremely Large-Scale Graphs on Post Peta-Scale Supercomputers

- **JST**(Japan Science and Technology Agency) **CREST**(Core Research for Evolutionally Science and Technology) **Project** (Oct, 2011 ～ March, 2017)
- **Winner of 8^{th} ,10^{th}, 11^{th} and 12^{th} Graph 500 benchmarks, and 1^{st} ～ 6^{th} Green Graph 500 benchmarks**
- **Collaborative research**

Panasonic NTT HITACHI Inspire the Next SUMITOMO ELECTRIC NEC sgi Hewlett Packard Enterprise

- **Innovative Algorithms and implementations**
  - Optimization, Searching, Clustering, Network flow, etc.
- Extreme Big Graph Data for emerging applications
  - **2^{30} ~ 2^{42} nodes** and **2^{40} ~ 2^{46} edges**
  - **Over 1M threads** are required for real-time analysis
- Many applications on post peta-scale supercomputers
  - Cyber security and social networks
  - Optimizing smart grid networks
  - Health care and medical science





CESS

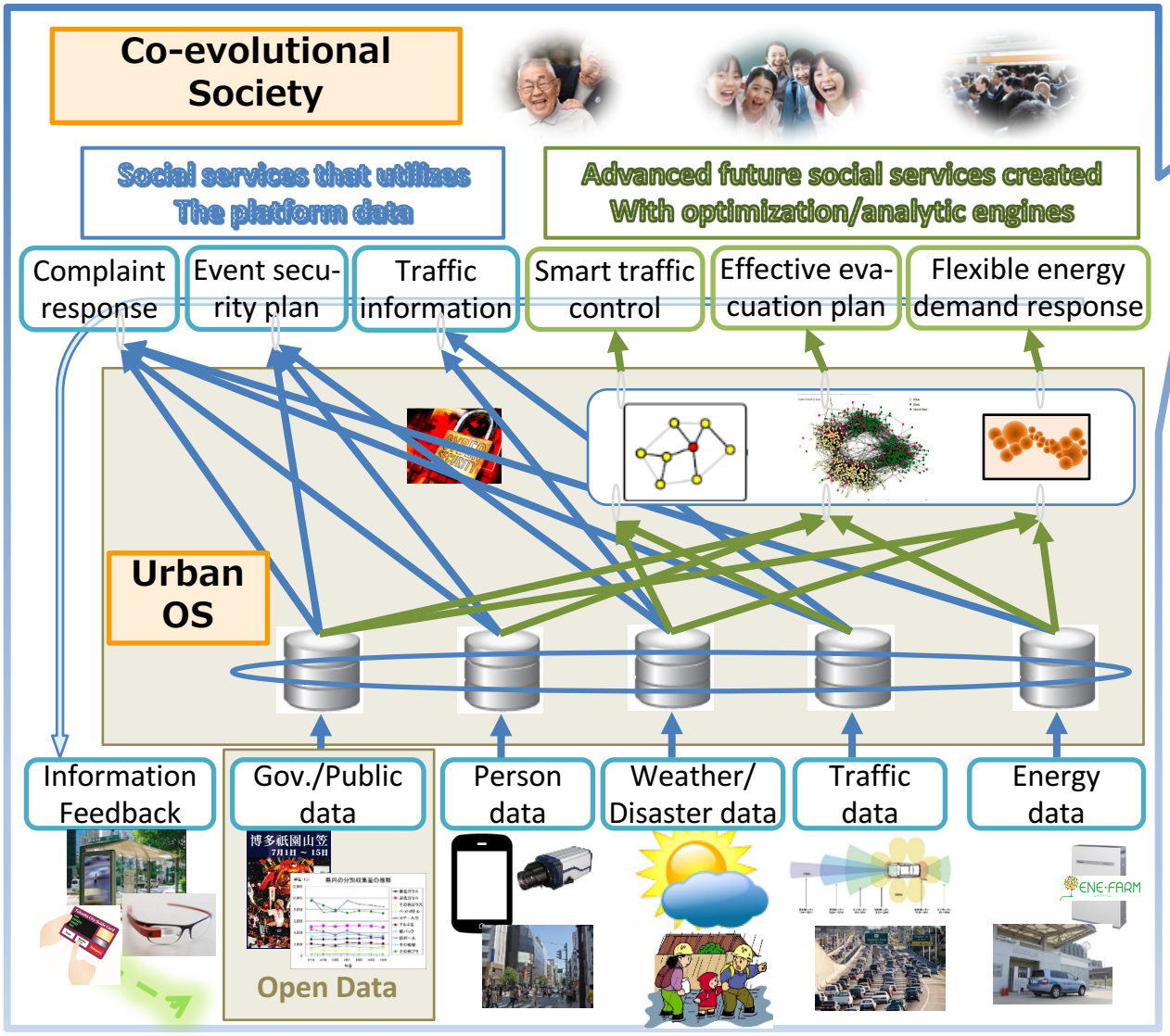
COI STREAM

Graph Analysis & High-Performance Computing Techniques for Realizing Urban OS

KYUSHU UNIVERSITY

Urban OS Functions

Open platform for advanced urban services



Application Service

Open platform for social/public/commercial applications

BODIC.org

Optimization/Analytic

Automatic optimization, control & bottleneck analysis

Data Store

Cross-utilization of various data

BODIC.org

Data

Big data / Open data
Sensor Network

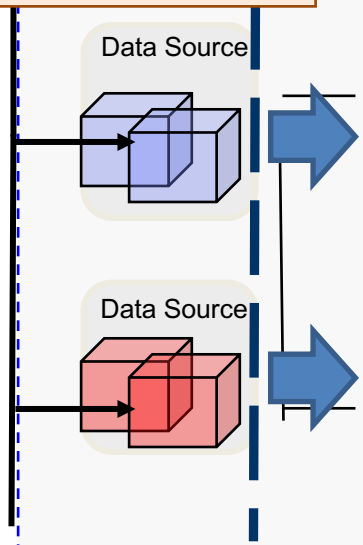
Advanced Computing and Optimization Infrastructure

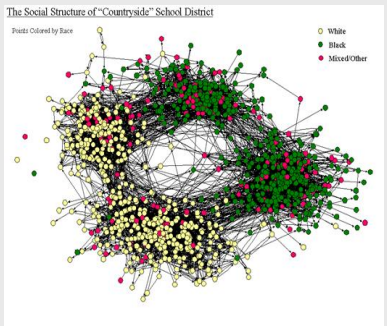
- Lower layer : Graph and network analysis algorithms
  - **Dijkstra algorithm (Single source shortest path problem with 2-heap) ,BFS (Breath first search algorithm) :** Shortest path, Centrality(BC etc.), Clustering problem
- Upper layer : Mathematical Optimization algorithms for NP-hard problems
  - **MIP(Mixed integer problem), SDP** : Facility location problem, Set covering (partitioning) problem, Scheduling, Evacuation Program

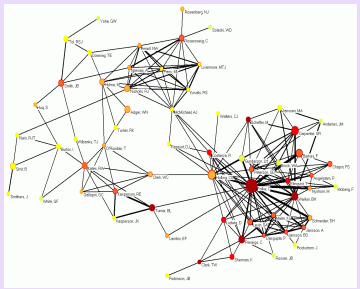
Large Sensor
- **Monitoring Data**
- **Smart Grid**
- **Traffic Transportation**
- **SNS (Twitter)**

Reduction of data size by utilizing graph analysis algorithms without loss of important components

Applying mathematical optimization algorithms to reduced data







HPC Technologies (CPU & Accelerator＋Data Store)

Betweenness centrality (BC)

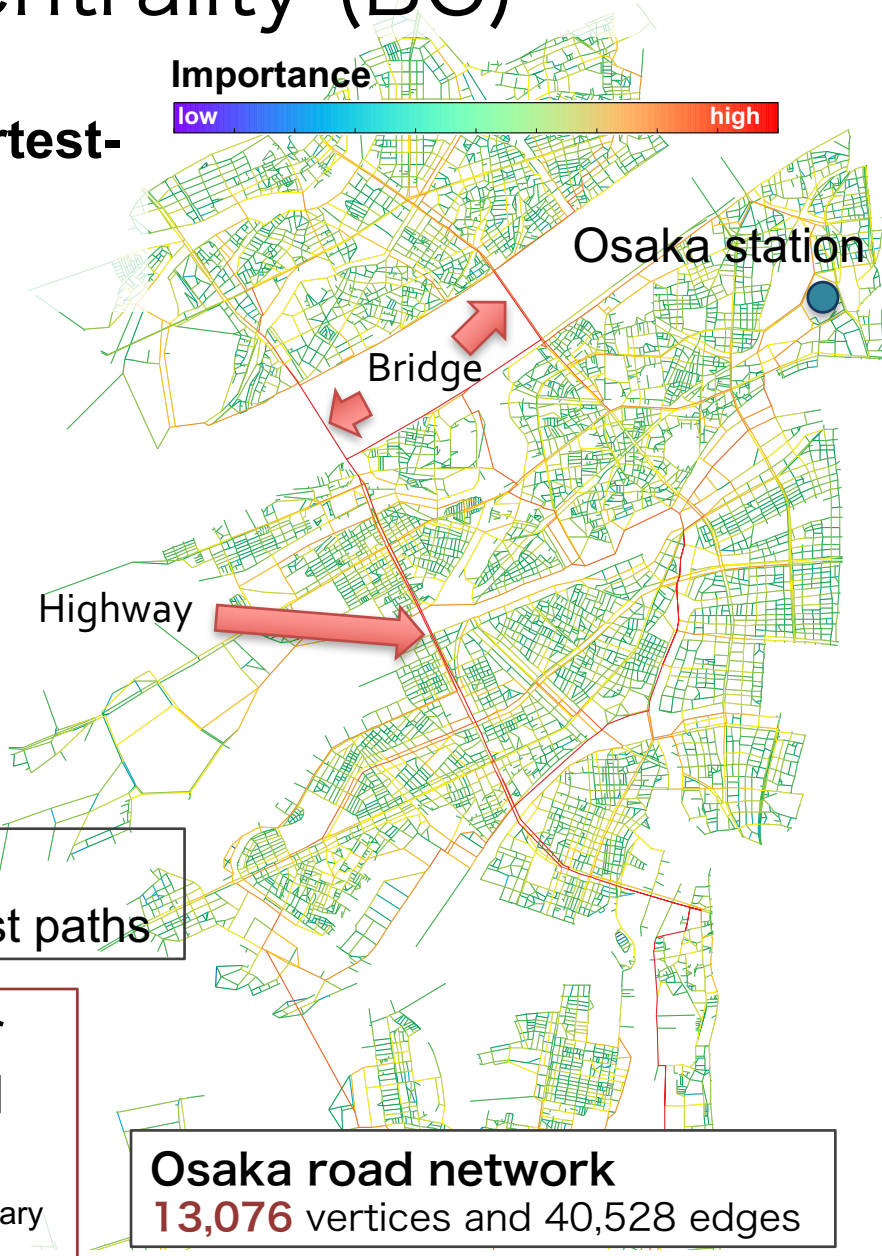
- Computes an **importance index** for each vertices and edges utilizing **all-to-all shortest-paths (breadth-first search)** w/o vertex coordinates

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

σ_{st} : # of (s, t)-shortest paths

$\sigma_{st}(v)$: # of (s, t)-shortest paths passing throw v

- BC requires *#vertices*-times BFS, because BFS obtains one-to-all shortest paths



Osaka road network
13,076 vertices and 40,528 edges

Our software "NETAL" can solves BC for Osaka road network **within one second**

Y. Yasui, K. Fujisawa, K. Goto, N. Kamiyama, and M. Takamatsu: NETAL: High-performance Implementation of Network Analysis Library Considering Computer Memory Hierarchy, JORSJ, Vol. 54-4, 2011.

Betweenness centrality

Tokyo Area in Japan
40 million people

Open Street Map
https://mapzen.com/metro-extracts

Graph

nodes 6,509,809
edges 14,460,834

Computation Time
98h 27m 37s

Huawei RH5885H V3
CPU ：Intel Xeon E7-4890 x 4
Memory ：2.0TB (32GB LRDIMM x 64 DIMMs)

Real-time Emergency Evacuation Planning

- catastrophic disasters by massive earthquakes are increasing in the world, and disaster management is required more than ever

Universally Quickest Flow(UQF) → Not simulation But Optimization Problem

UQF simultaneously maximizes the cumulative number of evacuees at an arbitrary time.

Evacuation planning can be reduced to UQF of a given dynamic network.

maximizes the cumulative number of evacuees

Quickest Evacuation