1st R-CCS International Symposium @ Kobe, Japan
Feb. 18, 2019

# Using Artificial Intelligence and Transprecision Computing for Accelerating Finite-Element Urban Earthquake Simulation

Tsuyoshi Ichimura, Kohei Fujita, Takuma Yamaguchi, Akira Naruse, Jack C. Wells,
Thomas C. Schulthess, Tjerk P. Straatsma, Christopher J. Zimmer,
Maxime Martinasso, Kengo Nakajima, Muneo Hori, Lalith Maddegedara


THE UNIVERSITY OF TOKYO



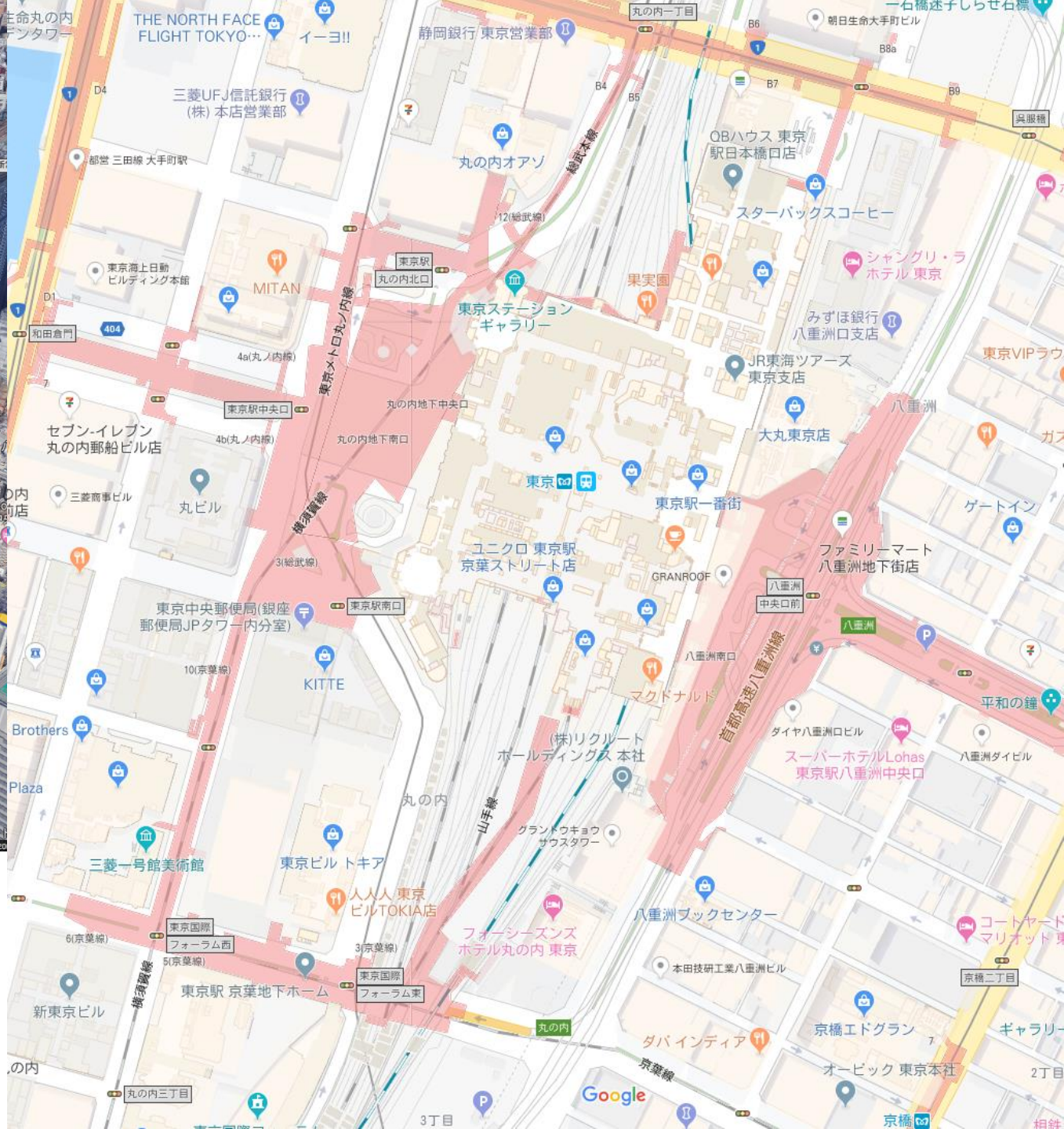
OAK RIDGE National Laboratory

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

# Smart cities

- Controlling cities based on real-time data for higher efficiency
- Computer modeling via high-performance computing is expected as key enabling tool
- Disaster resiliency is requirement; however, not established yet

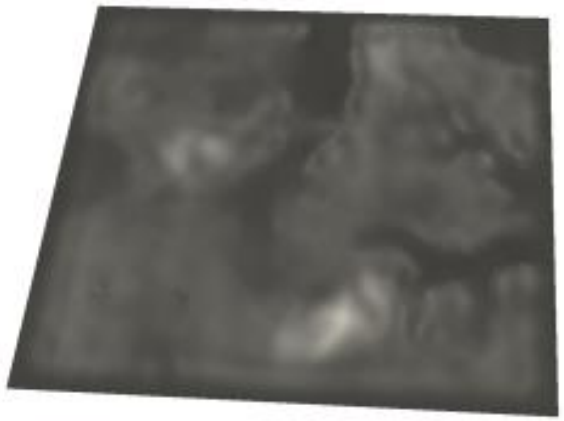

Example of highly dense city: Tokyo Station district

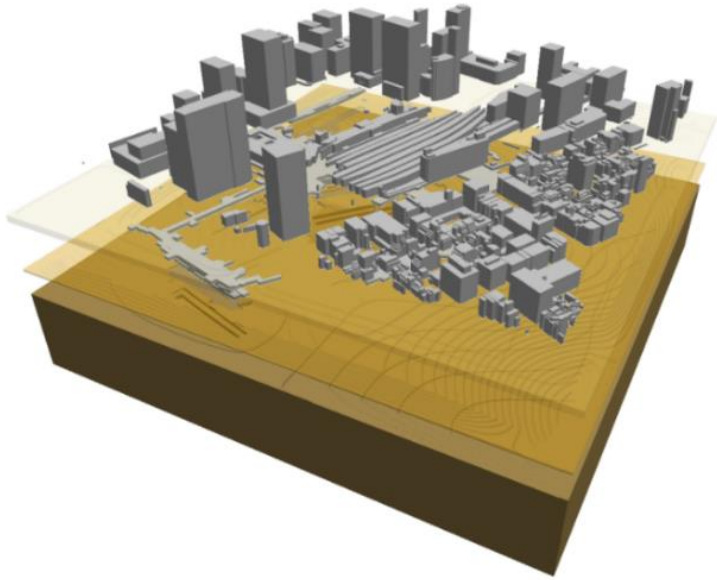

Fully coupled aboveground/underground earthquake simulation required for resilient smart city

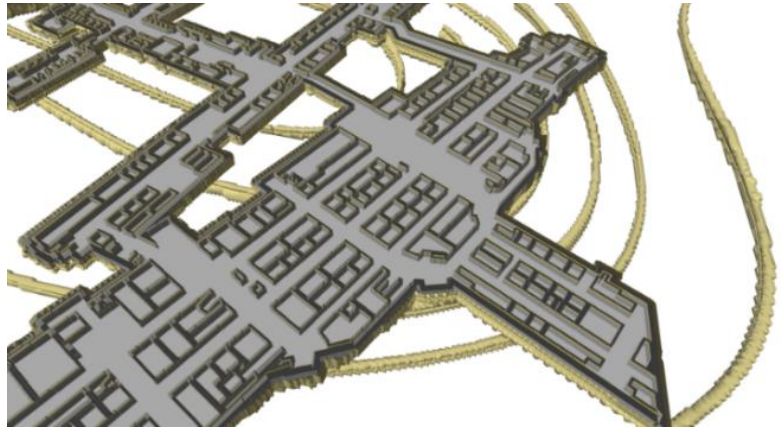
# Earthquake modeling of smart cities

- Unstructured mesh with implicit solvers required for urban earthquake modeling
  - We have been developing high-performance implicit unstructured finite-element solvers (SC14 & SC15 Gordon Bell Prize Finalist, SC16 best poster)
- However, simulation for smart cities requires full coupling in super-fine resolution
  - Traditional physics-based modeling too costly
  - Can we combine use of data analytics to solve this problem?



SC14, SC15 & SC16 solvers:
ground simulation only

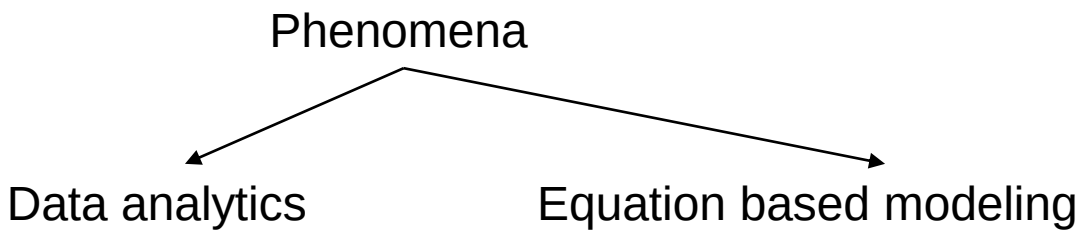




Fully coupled ground-structure simulation with underground structures

# Data analytics and equation based modeling

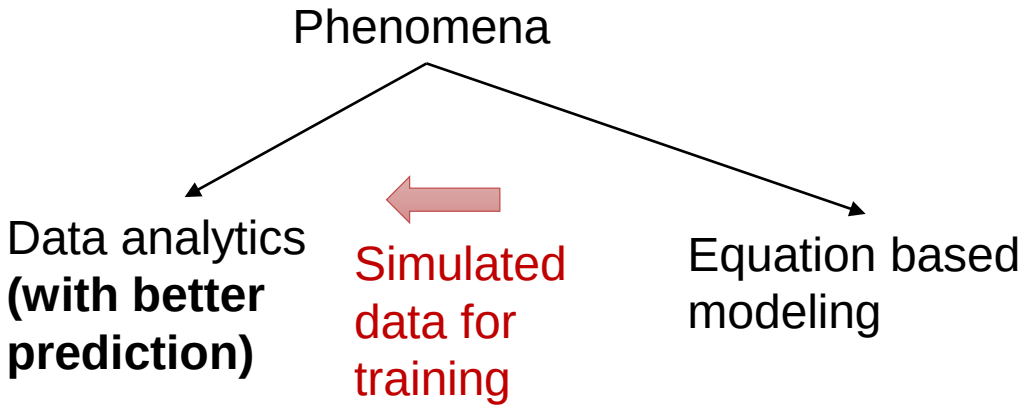
- Equation based modeling
  - Highly precise, but costly
- Data analytics
  - Fast inferencing, but accuracy not as high
- Use both methods to complement each other



# Integration of data analytics and equation based modeling

- First step: use data generated by equation based modeling for data analytics training
  - Use of high-performance computing in equation based modeling enables generating very large amounts of high quality data
  - We developed earthquake intensity prediction method using this approach (SC17 Best Poster)

**SC17**

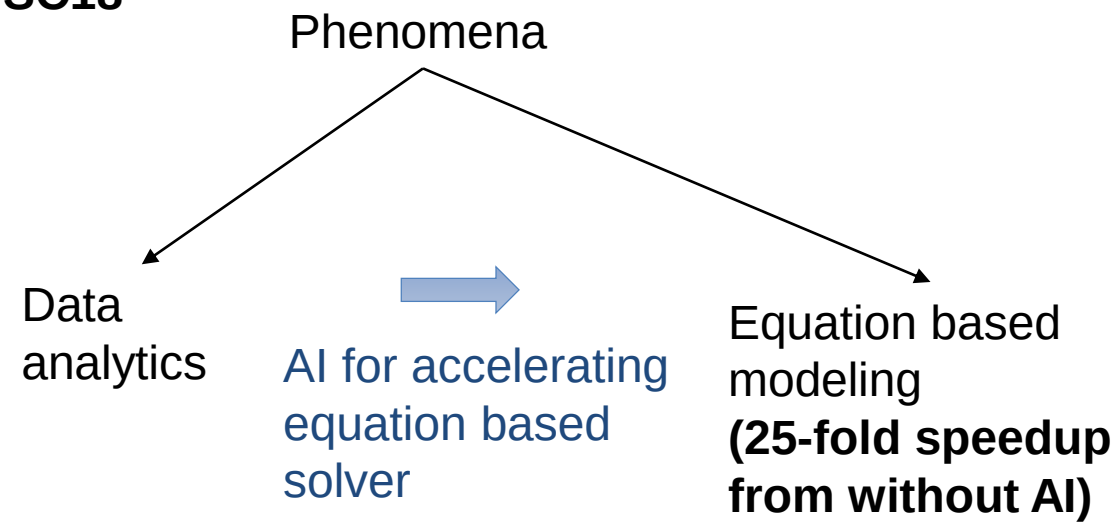



- SC14: equation based modeling
- SC15: equation based modeling
- SC16: equation based modeling
- **SC17: equation based modeling for AI**

# Integration of data analytics and equation based modeling

- We extend this concept in this paper: train AI to accelerate equation based modeling

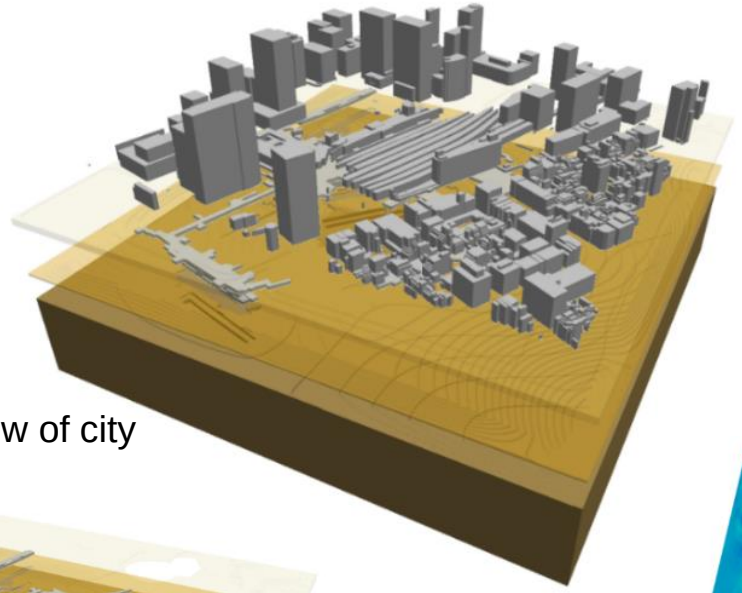
**SC18**



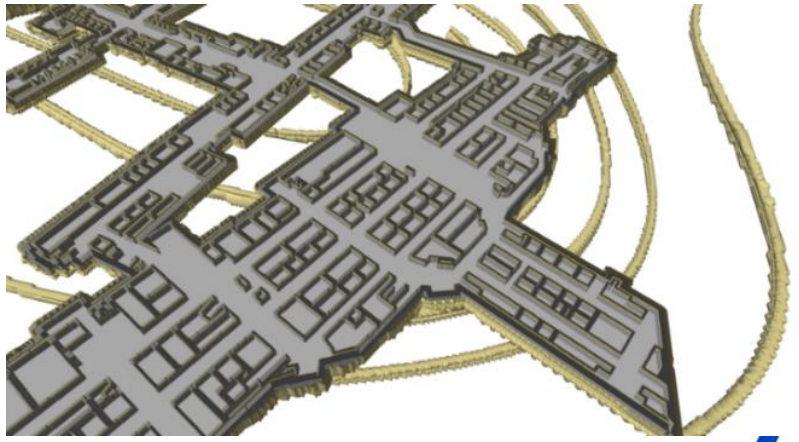

- SC14: equation based modeling
- SC15: equation based modeling
- SC16: equation based modeling
- SC17: equation based modeling for AI
- **SC18: AI for equation based modeling**

# Earthquake modeling for smart cities

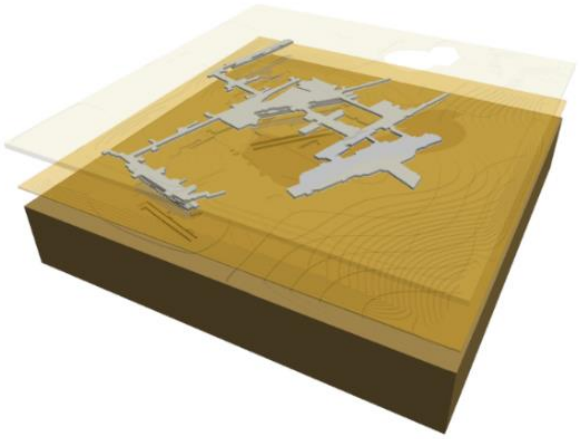
- By using AI-enhanced solver, we enabled fully coupled ground-structure simulation on Summit



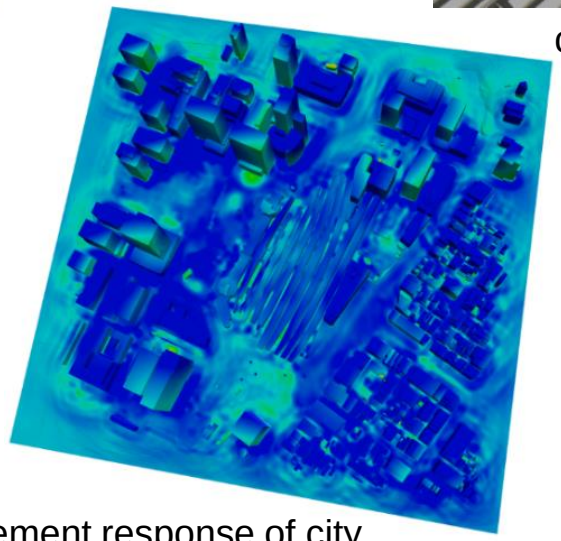
a) Overview of city model



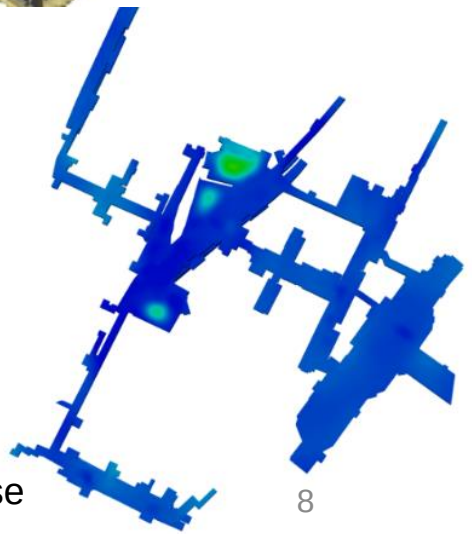
c) Close up view of city model



b) Location of underground structure



d) Displacement response of city



e) Displacement response of underground structure

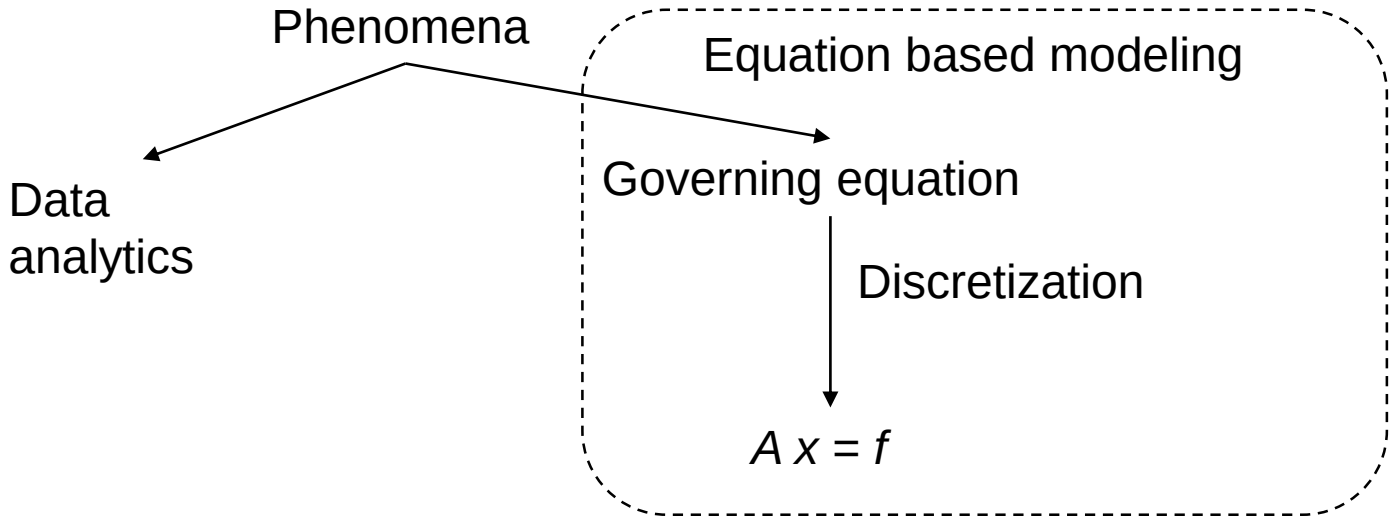
# Difficulties of using data analytics to accelerate equation based modeling

- Target: Solve $A\,x = f$
- Difficulty in using data analytics in solver
  - Data analytics results are not always accurate
  - **We need to design solver algorithm that enables robust and cost effective use of data analytics, together with uniformity for scalability on large-scale systems**
- Candidates: Guess $A^{-1}$ for use in preconditioner
  - For example, we can use data analytics to determine the fill-in of matrix; however, challenging for unstructured mesh where sparseness of matrix $A$ is nonuniform (difficult for load balancing and robustness)

  ➡ Manipulation of $A$ without additional information may be difficult…

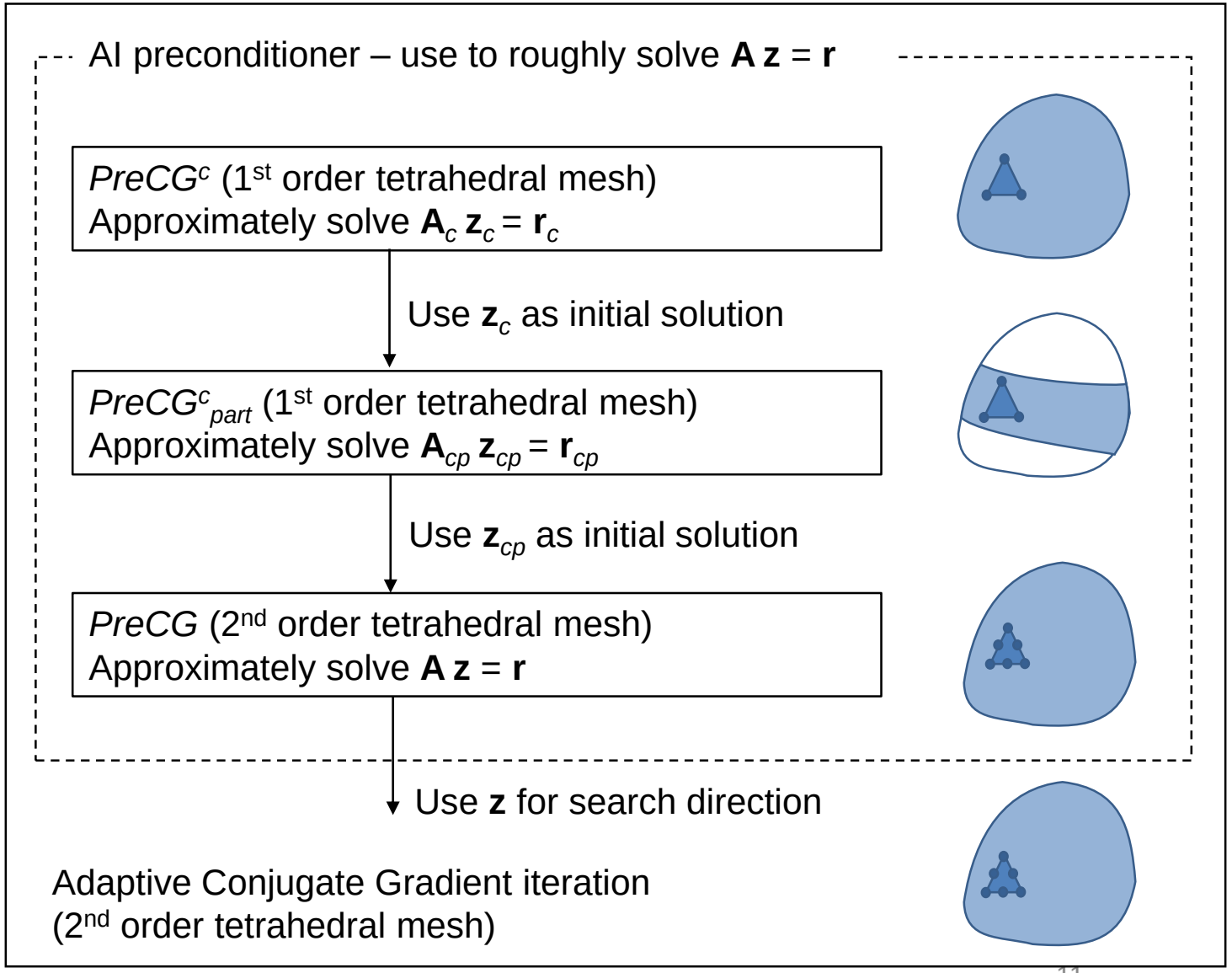
# Designing solver suitable for use with AI

- Use information of underlying governing equation
  - Governing equation's characteristics with discretization conditions should include information about the difficulty of convergence in solver
  - Extract parts with bad convergence using AI and extensively solve extracted part



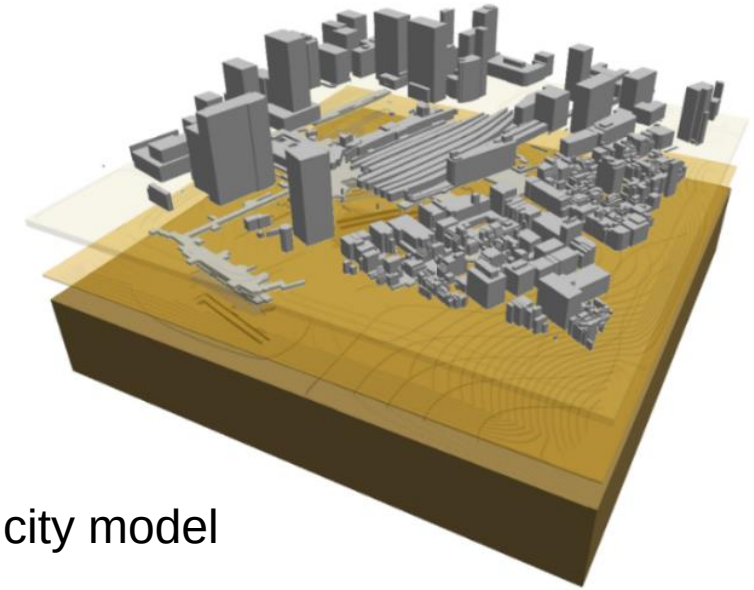
# Solver suitable for use with AI

- Transform solver such that AI can be used robustly
  - Select part of domain to be extensively solved in adaptive conjugate gradient solver
  - Based on the governing equation's properties, part of problem with bad convergence is selected using AI

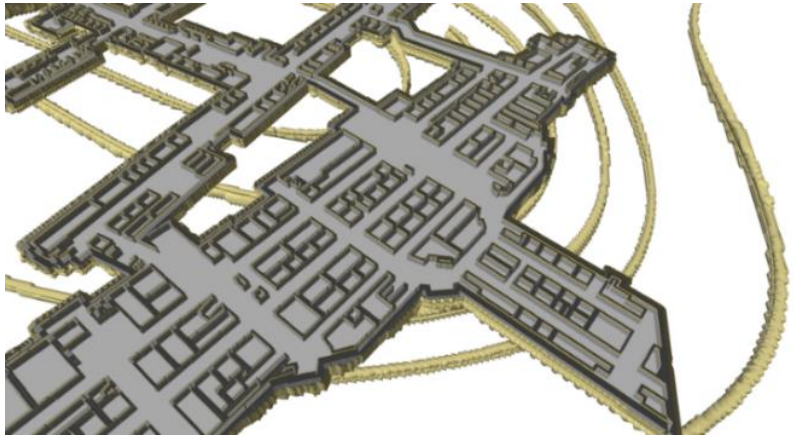


# How to select part of problem using AI

- In discretized form, governing equation becomes function of material property, element and node connectivity and coordinates
  - Train an Artificial Neural Network (ANN) to guess the degree of difficulty of convergence from these data



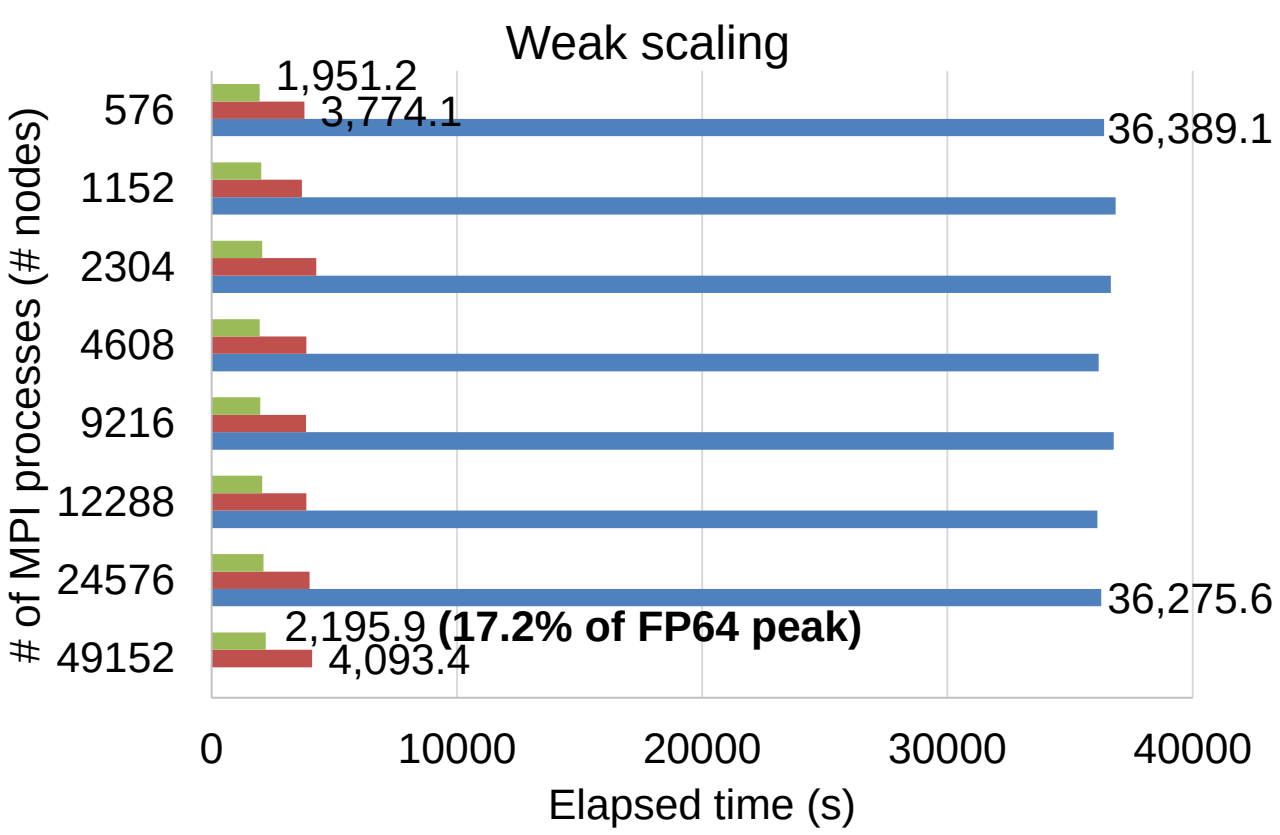
Whole city model

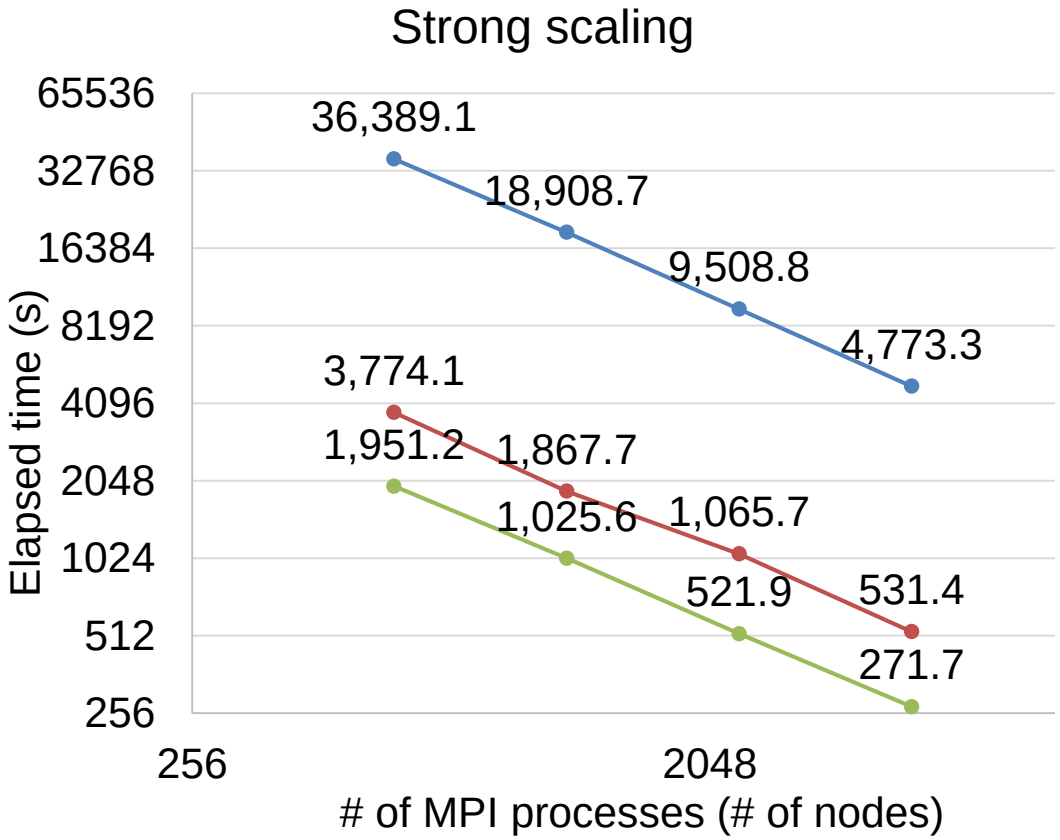


Extracted part by AI (about 1/10 of whole model)

# Performance of AI-enhanced solver on K computer

- FLOP count decreased by 5.56-times from PCGE (standard solver; Conjugate Gradient solver with block Jacobi preconditioning) and 1.32-times from SC14 Gordon Bell Prize finalist solver (with multi-grid & mixed-precision arithmetic)




Developed SC14 PCGE (Standard)

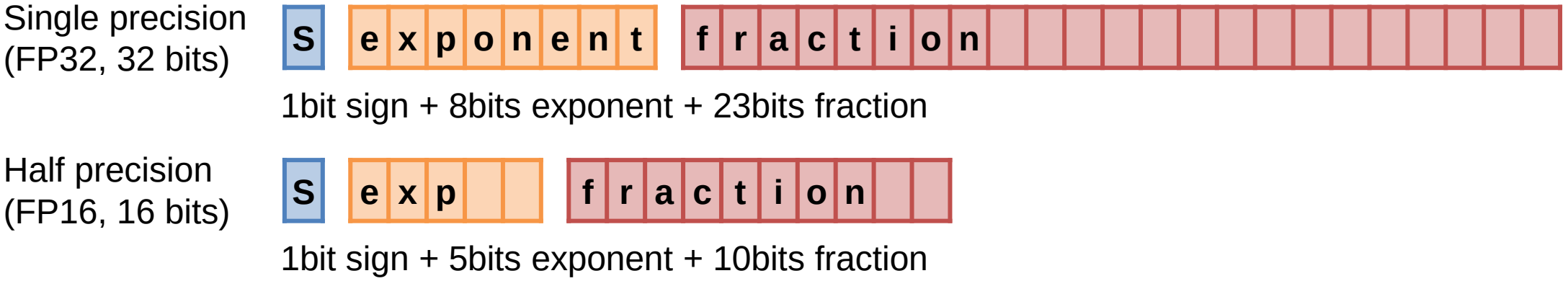
# Porting to Piz Daint/Summit

- Communication & memory bandwidth relatively lower than K computer
- Reducing data transfer required for performance
  - We have been using FP32-FP64 variables
  - Transprecision computing is available due to adaptive preconditioning

| | **K computer** | **Piz Daint** | **Summit** |
|---|---|---|---|
| CPU/node | 1 × SPARC64 VIIIfx | 1 × Intel Xeon E5-2690 v3 | 2 × IBM POWER 9 |
| GPU/node | - | 1 × NVIDIA P100 GPU | 6 × NVIDIA V100 GPU |
| Peak FP32 performance/node | 0.128 TFLOPS | 9.4 TFLOPS | 93.6 TFLOPS |
| Memory bandwidth | 512 GB/s | 720 GB/s | 5400 GB/s |
| Inter-node throughput | 5 GB/s<br>in each direction | 10.2 GB/s | 25 GB/s |

# Introduction of FP16 variables

- Half precision can be used for reduction of data transfer size




- Using FP16 for whole matrix or vector causes overflow/underflow or fails to converge
  - Smaller exponent bits → small dynamic range
  - Smaller fraction bits → no more than 4-digit accuracy

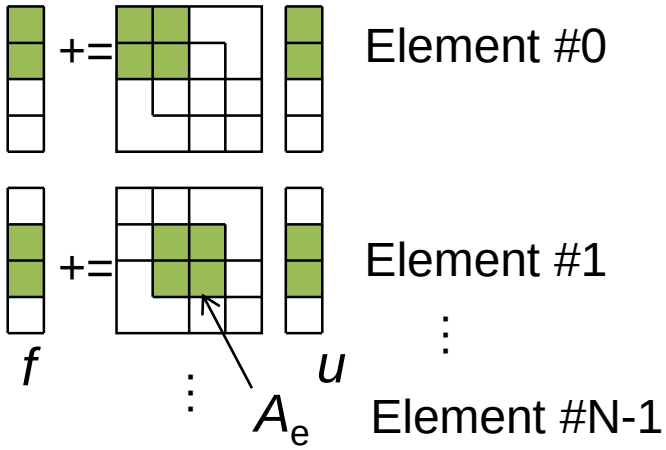
# FP16 computation in Element-by-Element method

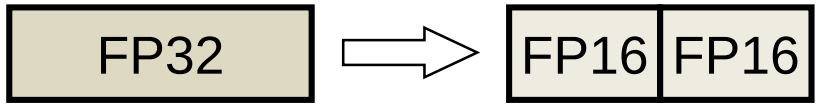
- Matrix-free matrix-vector multiplication
  - Compute element-wise multiplication
  - Add into the global vector
- Normalization of variables per element can be performed
  - Enables use of doubled width FP16 variables in element wise computation
  - Achieved 71.9% peak FP64 performance on V100 GPU
- Similar normalization used in communication between MPI partitions for FP16 communication

Element-by-Element (EBE) method

$f = \Sigma_e P_e A_e P_e^T u$

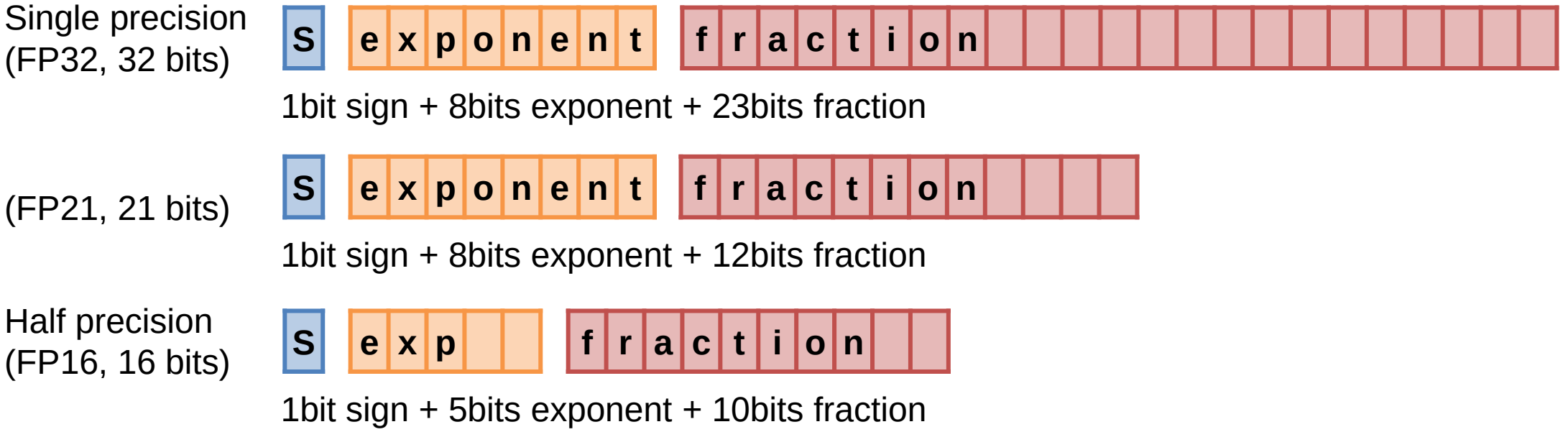
[$A_e$ is generated on-the-fly]



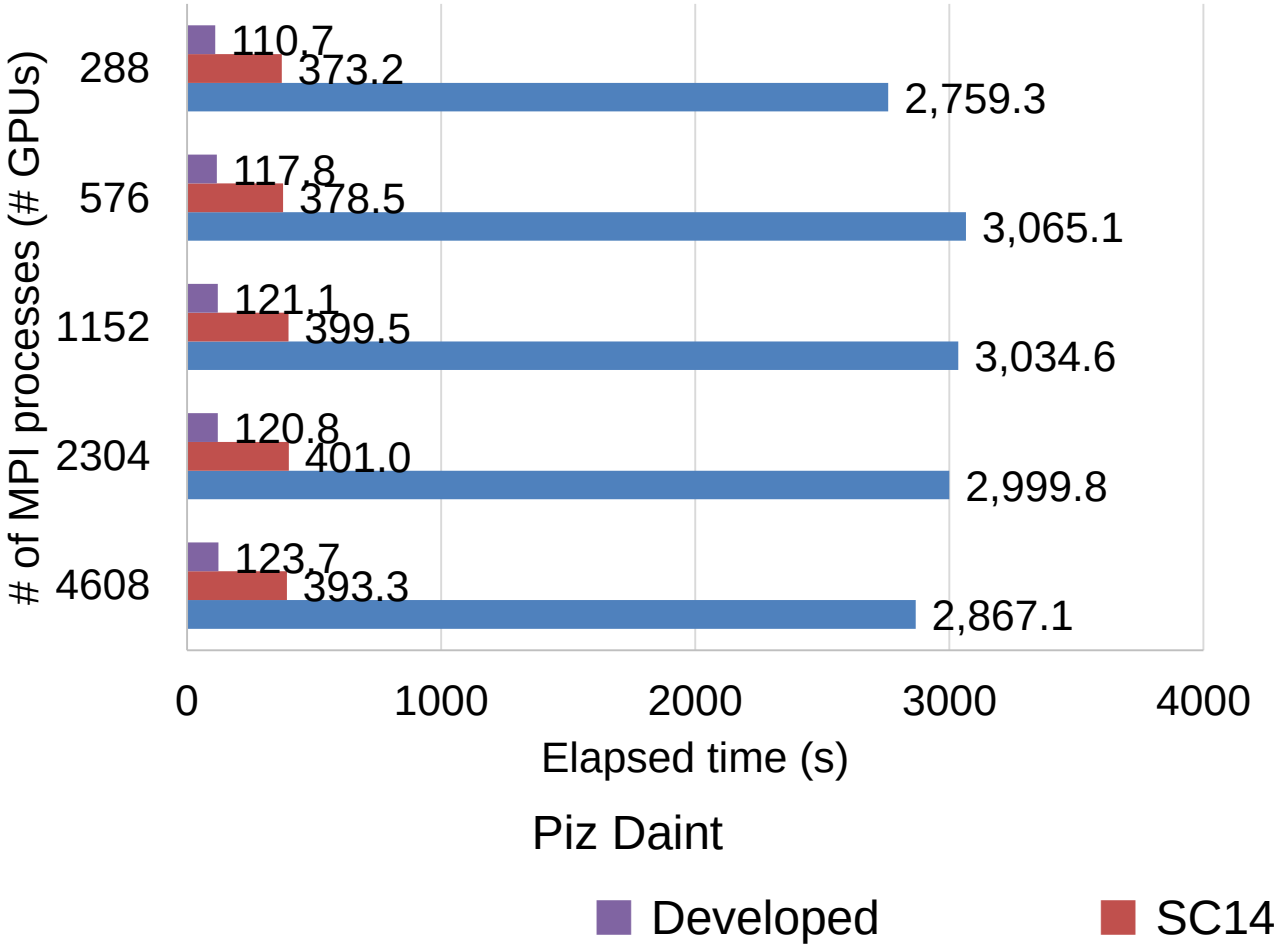
# Introduction of custom data type: FP21

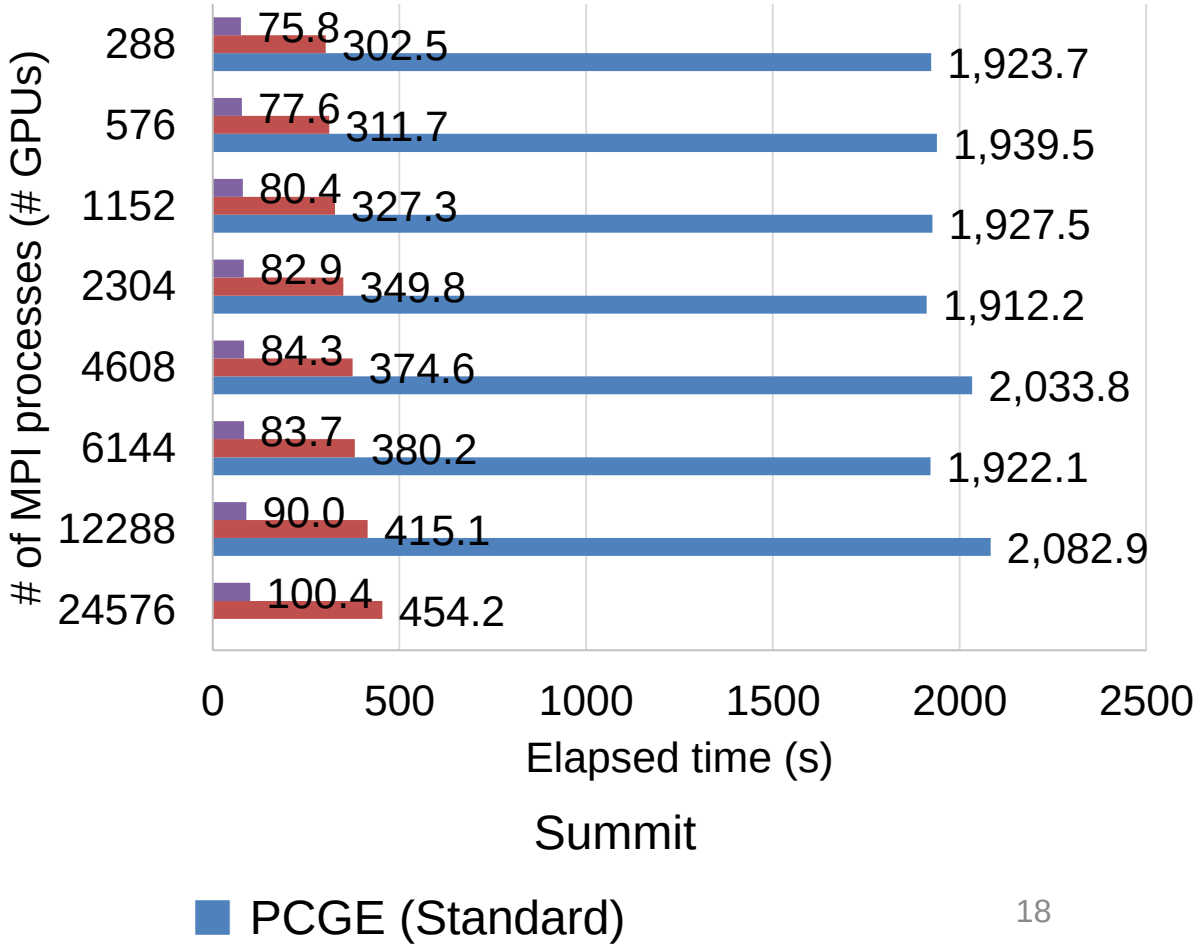
- Most computation in CG loop is memory bound
  - However, exponent of FP16 is too small for use in global vectors
- Use FP21 variables for memory bound computation
  - Only used for storing data (FP21 × 3 are stored into 64bit array)
  - Bit operations used to convert FP21 to FP32 variables for computation



# Performance on Piz Daint/Summit

- Developed solver demonstrates higher scalability compared to previous solvers
- Leads to 19.8% (nearly full Piz Daint) & 14.7% (nearly full Summit) peak FP64 performance




Developed SC14 PCGE (Standard)

# Summary and future implications

- New algorithms are required for accelerating equation based simulation by data analytics
  - We accelerated earthquake simulation by designing a scalable solver algorithm that can robustly incorporate data analytics
  - Combination with FP16-FP21-FP32-FP64 transprecision computation/communication techniques enabled high performance on recent supercomputers
- Idea of accelerating simulations with data analytics can be generalized for other types of equation based modeling
  - We plan to expand on this idea, together with transprecision computing for application development on Post-K computer

# Acknowledgments

Our results were obtained using K computer at RIKEN Center for Computational Science (R-CCS, proposal numbers: hp170249, hp180217), Piz Daint at Swiss National Supercomputing Centre (CSCS), and Summit at Oak Ridge Leadership Computing Facility, Oak Ridge National Laboratory (ORNL). We thank Yukihiko Hirano (NVIDIA) for coordination of the collaborative research project. We thank Christopher B. Fuson, Don E. Maxwell, Oscar Hernandez, Scott Atchley, Veronica Melesse-Vergara (ORNL), Jeff Larkin, Stephen Abbott (NVIDIA), Lixiang Luo (IBM), Richard Graham (Mellanox Technologies) for generous support concerning use of Summit. We thank Andreas Jocksch, Luca Marsella, Victor Holanda, Maria Grazia Giuffreda (CSCS) for generous support concerning use of Piz Daint. We thank the Operations and Computer Technologies Division of RCCS and the High Performance Computing Infrastructure helpdesk for generous support concerning use of K computer. We thank Sachiko Hayashi of Cybernet Systems Co., Ltd. for support in visualizing the application example. We acknowledge support from Post K computer project (Priority Issue 3 - Development of integrated simulation systems for hazards and disasters induced by earthquakes and tsunamis) and Japan Society for the Promotion of Science (18H05239, 26249066, 25220908, and 17K14719).