

Parallel Programming (English)

(Week 6)

Weifeng Liu

Department of Computer Science and Technology

China University of Petroleum - Beijing



Final project: Sparse Deep Neural Network GraphChallenge



[Home](#)

Challenges

GraphChallenge seeks input from diverse communities to develop graph challenges that take the best of what has been learned from groundbreaking efforts such as [GraphAnalysis](#), [Graph500](#), [FireHose](#), [MiniTri](#), and [GraphBLAS](#) to create a new set of challenges to move the community forward.

[NEW] Sparse Deep Neural Network Graph Challenge This challenge performs neural network inference on a variety of sparse deep neural networks.

- Specification: [slides](#), [paper](#), [example serial code](#), [example data sets](#)

Static Graph Challenge: Subgraph Isomorphism This challenge seeks to identify a given sub-graph in a larger graph.

- Specification: [slides](#), [paper](#), [example serial code](#), [example data sets](#), [Amazon instructions](#)

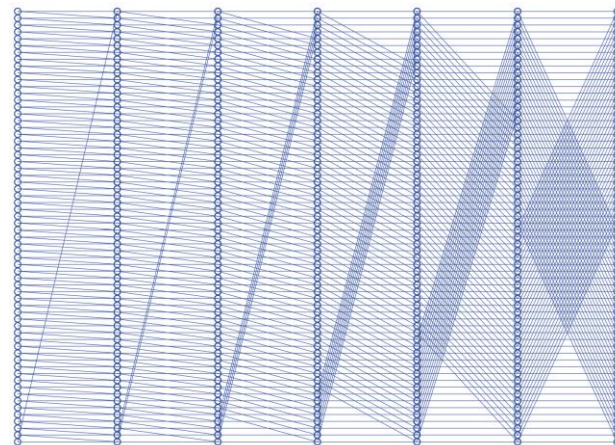
Streaming Graph Challenge: Stochastic Block Partition This challenge seeks to identify optimal blocks (or clusters) in a larger graph.

- Specification: [slides](#), [paper](#), [example serial code](#), [example data sets](#), [Amazon instructions](#)

Input data



MNIST dataset



RadiX-Net

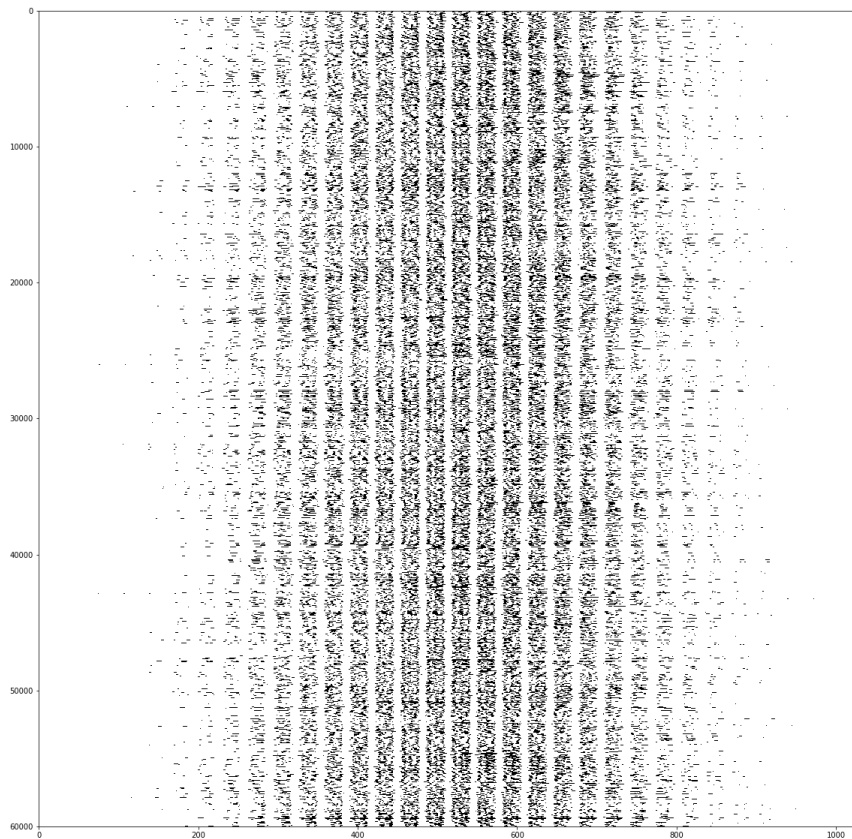
Number of
weight
matrices

Layers	Neurons per layer			
	1024	4096	16384	65536
120	3,932,160	15,728,640	62,914,560	251,658,240
480	15,728,640	62,914,560	251,658,240	1,006,632,960
1920	62,914,560	251,658,240	1,006,632,960	4,026,531,840

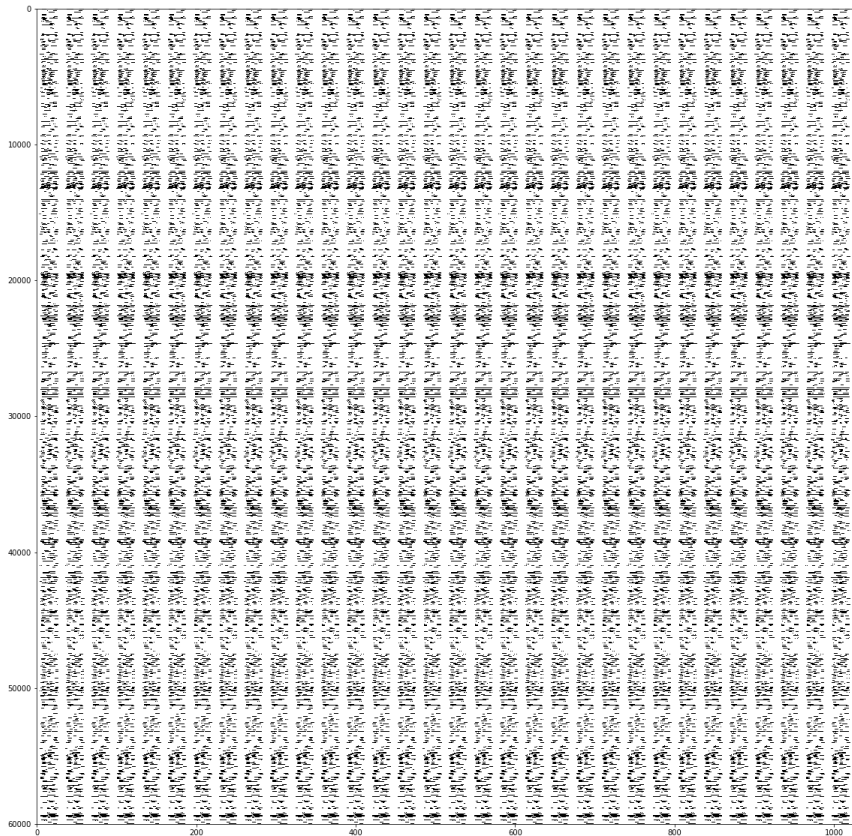
Size
of
weight
matrices

12 different sizes

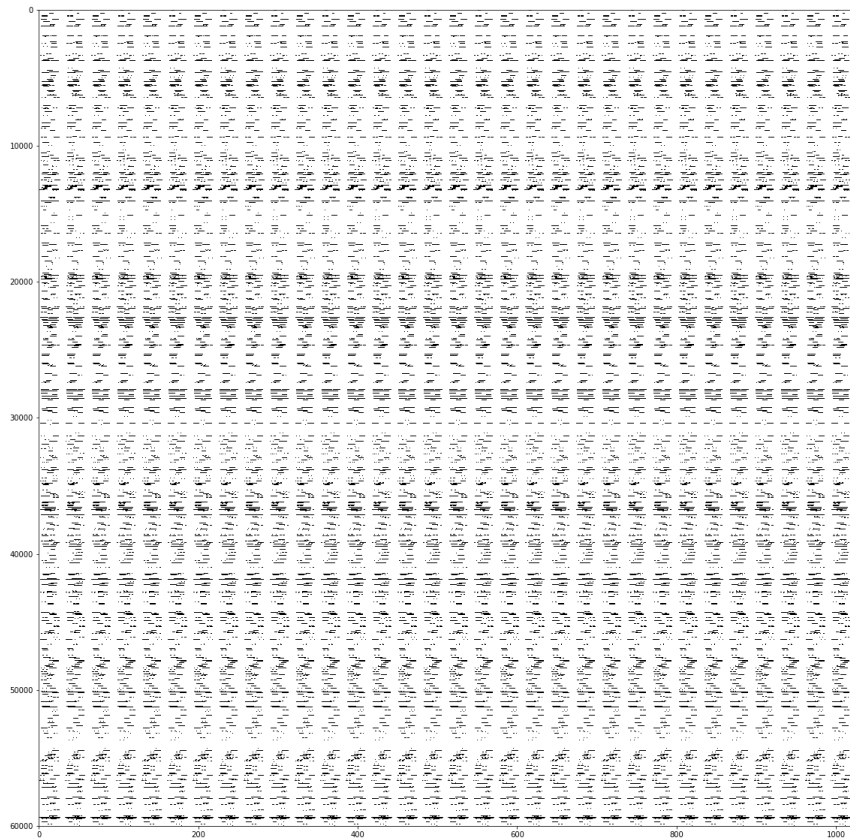
Data changes (layer 0)



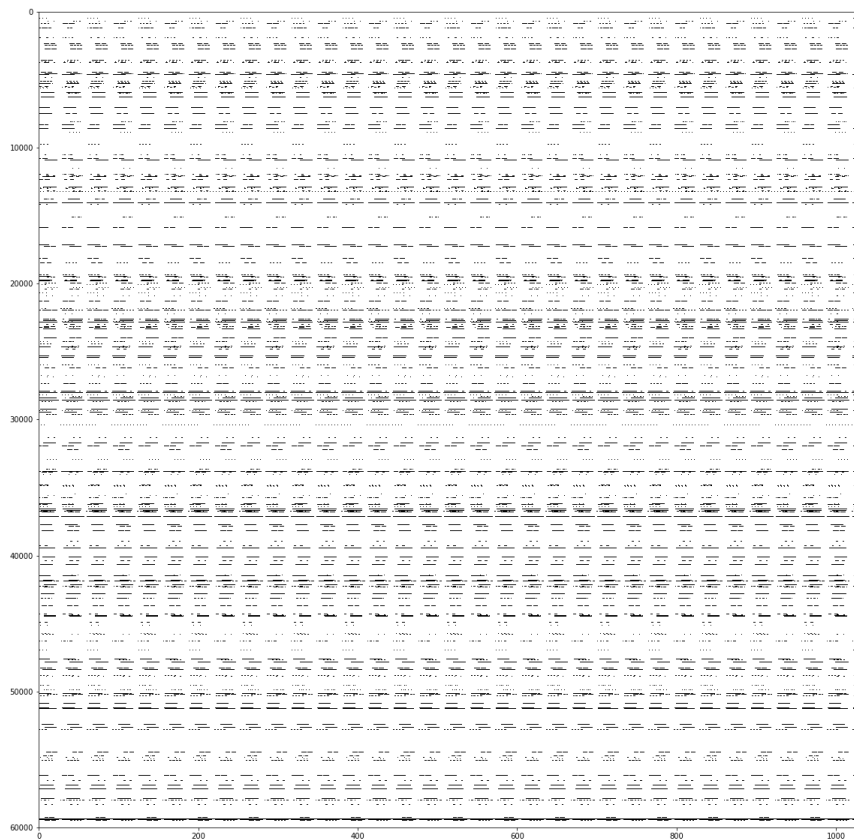
Data changes (layer 2)



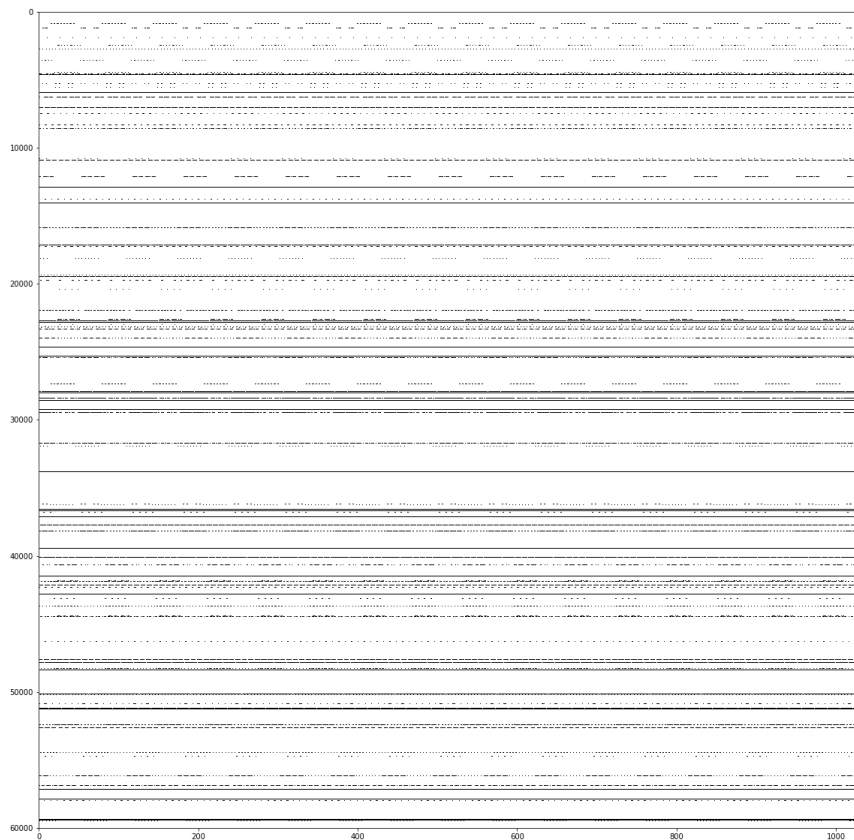
Data changes (layer 3)



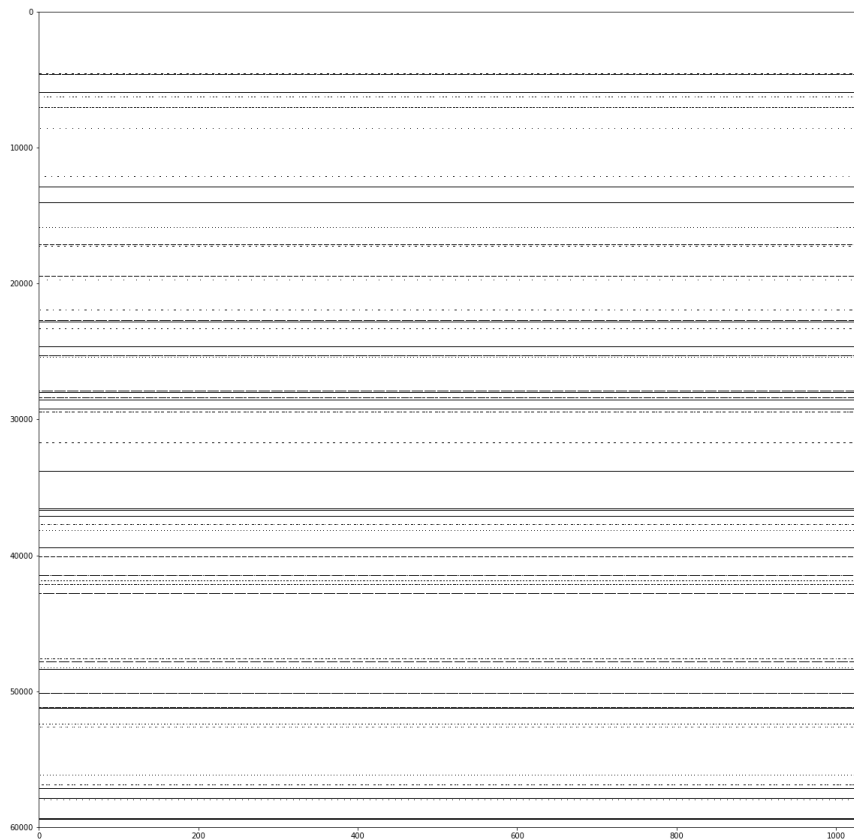
Data changes (layer 4)



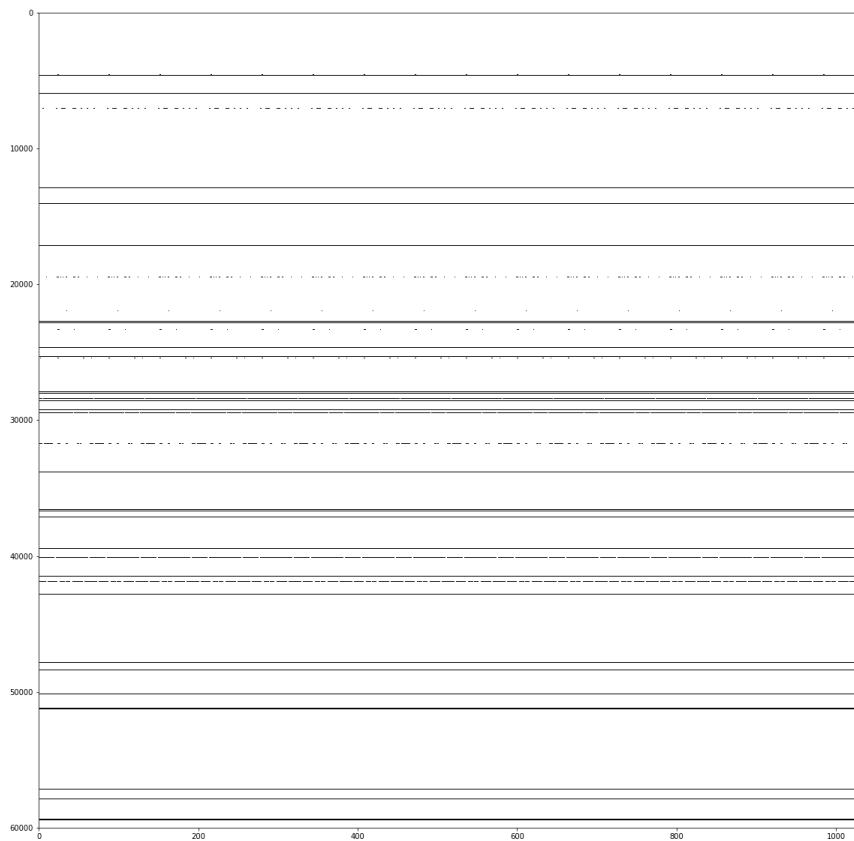
Data changes (layer 5)



Data changes (layer 6)



Data changes (layer 7)



Data changes (layer 8)



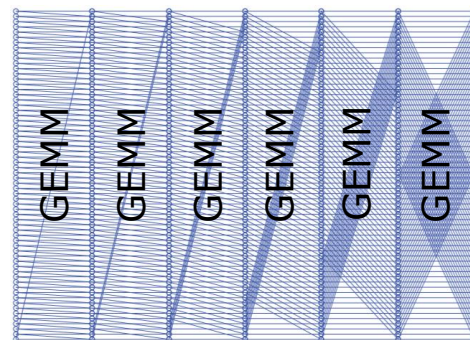
Data changes (layer 9)



Timing results of our reference code

```
File Edit View Search Terminal Help
[weifeng@localhost finalproject-SDNN]$ make
gcc -O3 -fopenmp -lm main.c -o sdnn
[weifeng@localhost finalproject-SDNN]$ ./sdnn
input matrix A: ( 60000, 1024 ) nnz = 6374505
Weight matrix load time: 5361.136000 ms
k = 1, GEMM time: 143391.96100 ms, Bias+ReLU time: 95.38200 ms
k = 2, GEMM time: 103282.16000 ms, Bias+ReLU time: 79.75200 ms
k = 3, GEMM time: 103290.31400 ms, Bias+ReLU time: 72.38000 ms
k = 4, GEMM time: 103296.50600 ms, Bias+ReLU time: 68.92000 ms
k = 5, GEMM time: 103292.89000 ms, Bias+ReLU time: 67.13200 ms
k = 6, GEMM time: 103464.63000 ms, Bias+ReLU time: 64.62200 ms
k = 7, GEMM time: 103278.84900 ms, Bias+ReLU time: 64.36500 ms
k = 8, GEMM time: 103288.94800 ms, Bias+ReLU time: 63.74300 ms
...
k = 111, GEMM time: 103268.54500 ms, Bias+ReLU time: 63.45400 ms
k = 112, GEMM time: 103239.78700 ms, Bias+ReLU time: 63.39600 ms
k = 113, GEMM time: 103267.48800 ms, Bias+ReLU time: 63.43300 ms
k = 114, GEMM time: 103254.53500 ms, Bias+ReLU time: 63.42000 ms
k = 115, GEMM time: 103252.22300 ms, Bias+ReLU time: 63.38700 ms
k = 116, GEMM time: 103261.00200 ms, Bias+ReLU time: 63.42600 ms
k = 117, GEMM time: 103257.97300 ms, Bias+ReLU time: 63.41300 ms
k = 118, GEMM time: 103235.65300 ms, Bias+ReLU time: 63.44100 ms
k = 119, GEMM time: 103265.48700 ms, Bias+ReLU time: 63.51100 ms
k = 120, GEMM time: 103431.26500 ms, Bias+ReLU time: 63.57800 ms
Inference time: 12446776.514000 ms
test
judge:0
CHALLENGE PASSED
[weifeng@localhost finalproject-SDNN]$
```

GEMM dominates the overall runtime.



The whole process takes ~207 minutes.

References (work awarded in 2019 and 2020)

2019 Champions

- *One Quadrillion Triangles Queried on One Million Processors* - Roger Pearce, Trevor Steil, Benjamin Priest, Geoffrey Sanders (Lawrence Livermore National Laboratory)
- *H-INDEX: Hash-Indexing for Parallel Triangle Counting on GPUs* - Santosh Pandey (Stevens Institute of Tech.), Xiaoye Li, Aydin Buluc (Lawrence Berkeley National Laboratory), Jiejun Xu (HRL), Hang Liu (Stevens Institute of Tech.)
- *Exploration of Fine-Grained Parallelism for Load Balancing Eager K-truss on GPU and CPU* - Mark P Blanco (Carnegie Mellon University and Sandia National Labs), Tze Meng Low (Carnegie Mellon University), Kyungjoo Kim (Sandia National Labs)
- *A GPU Implementation of the Sparse Deep Neural Network Graph Challenge* - Mauro Bisson, Massimiliano Fatica (NVIDIA)
- *Write Quick, Run Fast: Sparse Deep Neural Network in 20 Minutes of Development Time via SuiteSparse: GraphBLAS* - Timothy A Davis, Mohsen Aznaveh, Scott Kolodziej (Texas A&M University)

2019 Innovation Awards

- *Linear Algebra-Based Triangle Counting via Fine-Grained Tasking on Heterogeneous Environments* - Abdurrahman Yaşar (Georgia Institute of Technology), Siva Rajamanickam, Jonathan Berry, Michael M Wolf (Sandia National Laboratories), Jeffrey Young, Ümit V. Çatalyürek (Georgia Institute of Technology)
- *Scalable Triangle Counting on Distributed-Memory Systems* - Seher Acer (Sandia National Laboratories), Abdurrahman Yaşar (Georgia Institute of Technology), Siva Rajamanickam, Michael M Wolf (Sandia National Laboratories), Ümit V. Çatalyürek (Georgia Institute of Technology)
- *Scaling and Quality of Modularity Optimization Methods for Graph Clustering* - Sayan Ghosh, Mahantesh Halappanavar, Antonino Tumeo (Pacific Northwest National Laboratory), Ananth Kalyanaraman (Washington State University)
- *Distributed Direction-Optimizing Label Propagation for Community Detection* - Xu Liu (Washington State University); Jesun S Firoz, Marcin Zalewski, Mahantesh Halappanavar, Kevin Barker, Andrew Lumsdaine (Pacific Northwest National Laboratory), Assefaw H Gebremedhin (Washington State University)
- *Scalable Inference for Sparse Deep Neural Networks using Kokkos Kernels* - John A Ellis, Sivasankaran Rajamanickam (Sandia National Laboratories)

References (work awarded in 2019 and 2020)

2019 Student Innovation Awards

- *DistTC: High Performance Distributed Triangle Counting* - Loc Hoang, Vishwesh Jatala, Xuhao Chen, Udit Agarwal, Roshan Dathathri, Gurbinder S Gill, Keshav Pingali (The University of Texas at Austin)
- *Fast Stochastic Block Partitioning via Sampling* - Frank D Wanye (Virginia Tech), Vitaliy Gleyzer (MIT Lincoln Laboratory), Wu-chun Feng (Virginia Tech)
- *Update on k-truss Decomposition on GPU* - Mohammad Almasri, Omer Anjum, Carl Pearson, Zaid Qureshi, Vikram Sharma Mailthody, Rakesh Nagi (University of Illinois at Urbana-Champaign), Jinjun Xiong (IBM Thomas J. Watson Research Center); Wen-Mei Hwu (UIUC)
- *Accelerating DNN Inference with GraphBLAS and the GPU* - Xiaoyun Wang, Zhongyi Lin, Carl Yang, John D Owens (University of California, Davis)

2019 Finalists

- *Fast Parallel BFS-Based Triangle Counting on GPUs* - Levuan Wang, John D Owens (University of California, Davis)
- *Performance of Training Sparse Deep Neural Networks on GPUs* - JIANZONG WANG (平安科技《深圳》有限公司); Zhangcheng Huang (Ping An Technology (Shenzhen) Co., Ltd); Lingwei Kong (PingAn Tech); Jing Xiao (Ping An Insurance (Group) Company of China); Pengyu Wang (Shanghai Jiao Tong University); Lu Zhang (Shanghai Jiao Tong University); Chao Li (Shanghai Jiaotong University)

2019 Honorable Mention

- *Fast Triangle Counting on GPU* - Chuangyi Gui, Long Zheng, Pengcheng Yao, Xiaofei Liao, Hai Jin (Huazhong University of Science and Technology)
- *Update on Triangle Counting on GPU* - Carl Pearson, Mohammad Almasri, Vikram Sharma Mailthody, Zaid Qureshi, Omer Anjum, Wen-Mei Hwu (University of Illinois at Urbana-Champaign), Jinjun Xiong (IBM Thomas J. Watson Research Center); Rakesh Nagi (UIUC)
- *Multithreaded Layer-wise Training of Sparse Deep Neural Networks using Compressed Sparse Column* - Mohammad Hasanzadeh Mofrad, Rami Melhem (University of Pittsburgh), Yousuf Ahmad, Mohammad Hammoud (Carnegie Mellon University in Qatar)
- *Accelerating Sparse Deep Neural Network on FPGA* - Sitao Huang, Carl Pearson, Rakesh Nagi, (University of Illinois at Urbana-Champaign), Jinjun Xiong (IBM Thomas J. Watson Research Center), Deming Chen, Wen-Mei Hwu (University of Illinois at Urbana-Champaign)

References (work awarded in 2019 and 2020)

2020 Champions

- *Scaling Graph Clustering with Distributed Sketches* - Benjamin Priest (LLNL), Alec Dunton (CU Boulder), Geoffrey Sanders (LLNL)
- *At-Scale Sparse Deep Neural Network Inference With Efficient GPU Implementation* - Mert Hidayetoglu, Carl Pearson, Vikram Sharma Mailthody (UIUC), Eiman Ebrahimi (Nvidia), Jinjun Xiong (IBM), Rakesh Nagi, Wen-mei W. Hwu (UIUC)
- *A Novel Inference Algorithm for Large Sparse Neural Network using Task Graph Parallelism* - Dian-Lun Lin, Tsung-Wei Huang (Univ of Utah)
- *TriC: Distributed-memory Triangle Counting by Exploiting the Graph Structure* - Sayan Ghosh, Mahantesh Halappanavar (PNNL)

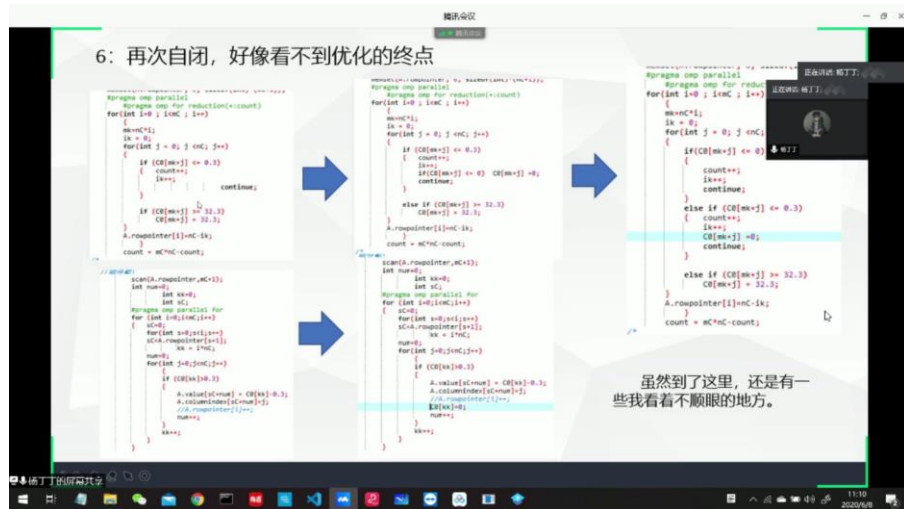
2020 Innovation Awards

- *Combinatorial Tiling for Sparse Neural Networks* - Filip Pawłowski (ENS Lyon), Rob H. Bisseling (Utrecht), Bora Ucar (CNRS), Albert-Jan Yzelman (Huawei)
- *Studying the Effects of Hashing of Sparse Deep Neural Networks on Data and Model Parallelisms* - Mohammad Hasanzadeh Mofrad, Rami Melhem (Univ of Pittsburgh), Yousuf Ahmad, Mohammad Hammoud (CMU Qatar)
- *Incremental Streaming Graph Partitioning* - Lisa Durbeck, Peter Athanas (Virginia Tech)

2020 Honorable Mention







- *KTRUSSEXPLORER: Exploring the Design Space of K-truss Decomposition Optimizations on GPUs* - Safaa Diab, Mhd Ghaith Olabi, Izzat El Hajj (American University of Beirut)
- *Analysis of Floating-Point Round-Off Error in Linear Algebra Routines for Graph Clustering* - L. Minah Yang (CU Boulder), Alyson Fox (LLNL)

References (presented by AI17 students last year)



链接: <https://pan.baidu.com/s/1R4VLpRcbGlcJZTPURRRGhQ> 提取码: 79iz

Reference code and papers

<div><div><div><</div><div>></div><div>🔄</div></div><div>我的网盘 > 2020-2021春季并行程序设计 (全英文) 课程录像 ></div><div>搜索我的网盘文件</div></div>				
<input type="checkbox"/> 文件名	↓	修改时间	类型	大小
<input type="checkbox"/>  20210602_并行程序设计 (全英文) w6s1(期末大作业稀疏神经网络推理参考代码)finalproject-SDNN.zip		2021-06-02 16:12	zip文件	26.42MB
<input type="checkbox"/>  20210602_并行程序设计 (全英文) w6s1(参考文献)GraphChallenge-papers19and20.zip		2021-06-02 16:47	zip文件	10.75MB
<input type="checkbox"/>  20210526_并行程序设计 (全英文) w5s1(MPI+SUMMA).mp4		2021-06-02 16:55	mp4文件	115.87MB
<input type="checkbox"/>  20210519_并行程序设计 (全英文) w4s1(CUDA+SpMM).mp4		2021-06-02 16:54	mp4文件	138.30MB
<input type="checkbox"/>  20210512_并行程序设计 (全英文) w3s1(CUDA+vecadd+GEMM).mp4		2021-06-02 16:55	mp4文件	127.07MB
<input type="checkbox"/>  20210428_并行程序设计 (全英文) w1s1(Introduction).mp4		2021-06-02 16:55	mp4文件	161.11MB

链接: https://pan.baidu.com/s/1EkQDW8skdGUNR_43fojISw 提取码: merk

Requirements of the project

- Work in group, ask TA for technical supports;
- Use **sparsity** of the matrices, and **MPI+CUDA** to program;
- Work (**data structure and algorithm design, sparsifying, MPI, CUDA, report writing**, etc.) should be clearly divided to individuals;
- All tests will be done on our 4-node 32-GPU cluster;
- Hand in your code and a final report describing your algorithm and performance.

Deadlines of the project

- **1st Deadline is June 9th**
 - Use slides to show your work assignment and your progress
 - All team members should come to the stage to say your task
 - No more than 3 minutes
- **2nd Deadline is June 16th**
 - Use slides to show your final performance results
 - All team members should come to the stage to show your contribution
 - No more than 5 minutes
 - Hand in your code and a final report of your group

Thanks!

