Yuankun Fu

Seeking a full-time position Research Scientist / SWE starting from May 2021.

Personal Information

Personal Site: qoofyk.github.io
Linkedin: linkedin.com/in/yuankun-fu
Github: github.com/qoofyk
email: fu121@purdue.edu
Phone: 317-373-7240

Research Interests

HPC: Parallel Programming Models & Runtime Systems, Advanced parallel algorithms, Automated performance analysis & optimization, In-situ analysis & visualization, GPU acceleration;
Cloud: Software defined networking (SDN); Distributed workflow; Data Storage
Artificial intelligence: Distributed machine learning, Self-driving, Virtual reality.
CFD: Lattice Boltzmann Method (LBM), Molecule dynamics, Large Eddy Simulation;

Education

- Ph.D., Computer Science, Purdue University, 08/2014∼04/2021 (Targeted), GPA 3.7/4
- M.S., Computer Science, University of Chinese Academy of Sciences, 09/2011∼06/2014, GPA 3.5/4
- B.S., Electronic Engineering, Northeastern University, China, 09/2007∼06/2011, GPA 3.6/4, Top 3%

Work & Service experiences

- Research Assistant Intern, Los Alamos National Laboratory, 05/18∼08/18
  Programming model team in CCS-7 group
- Research Assistant, Institute of Computing Technology, Beijing, China, 09/2011∼06/2014
- Teaching Assistant, Purdue University, *08/2014 ∼ Present*
  Operating System: Fall 2016, Spring 2019, Fall 2019
  Computer Architecture: Spring 2017, Fall 2018, Spring 2020, Fall 2020, Spring 2021
- Student Volunteer, SC'2018 (Dallas, US), SC'2020 (Atlanta, US)

Publications

1. Yuankun Fu, Fengguang Song, *"Designing a 3D Parallel Memory-Aware Lattice Boltzmann Algorithm on Manycore Systems"*, **Euro-Par'21**, Lisbon Portugal, 09/2021. (In submission)
2. Feng Li, Yuankun Fu, etc., *"Accelerating complex hydrology workflows using Vistrails and LaunchAgent coupling local and HPC/Cloud resources on-demand"*, **PEARC'21**, 07/2021. (In submission)
3. Slaughter Elliott, Wei Wu, Yuankun Fu, etc.*"Task Bench: A Parameterized Benchmark for Evaluating Parallel Runtime Performance"*, **SC'20**, Atlanta, Georgia, 11/2020. [PDF]
4. Yuankun Fu, Feng Li, Fengguang Song, Luoding Zhu, *"Designing a Parallel Memory-Aware Lattice Boltzmann Algorithm on Manycore Systems"*, **SBAC-PAD'18**, Lyon, France, 09/2018. [PDF]
5. Yuankun Fu, Feng Li, Fengguang Song, Zizhong Chen, *"Performance Analysis and Optimization of In-situ Integration of Simulation with Data Analysis: Zipping Applications Up"*, **HPDC'18**, Tempe, Arizona, 06/2018. [PDF]
6. Yuankun Fu, Fengguang Song, *"SDN helps Big-Data to optimize access to data"*, chapter 14, 297-318(504), Big Data and Software Defined Networks, Stevenage, UK, 03/2018. [PDF]
7. Yuankun Fu, Fengguang Song, Luoding Zhu, *"Modeling and Implementation of an Asynchronous Approach to Integrating HPC and Big Data Analysi"*, **ICCS'16**, San Diego, CA, 06/2016. [PDF]

Skills

- HPC, C, C++, MPI, OpenMP, Pthreads, GPU/CUDA/HIP, Python, Roofline, Tracing, ADIOS, SLURM
- CFD, Palabos, LAMMPS, OpenLB, Paraview/Catalyst, OpenFoam, Fieldview
- AWS, GCP, Kubernetes, Openstack, Docker, Ansible, Hadoop, Spark, SQL, Django
- Machine Learning, Deep Learning, Reinforcement Learning, TensorFlow, Keras, Horovod
- Linux, Bash, Git, CMake, Scons, Json, Java, Matlab, LaTeX
- Leadership, Public speaking, Interpersonal communication

Featured Research & Projects

1. **World's fastest in-situ workflow system** on HPC.
   09/2014∼01/2018, Purdue University, Ph.D. Research. Related Publications: [5,6,7]
   Advised by Prof. Fengguang Song, Luoding Zhu, Zhiyuan Li, Zizhong Chen
   - Built **the first in-situ workflow benchmark** in the scientific workflow community. It integrated 7 state-of-the-art in-situ workflow systems with 3 synthetic apps, 2 real-world apps (LBM & LAMMPS Molecule dynamics), and real-world data analysis. Then detailed *performance analysis* using **trace profiling tools** showed that even the fastest existing in-situ system still had 42% overhead.
   - Developed a novel **minimized end-to-end** in-situ workflow system, **Zipper**. It supported the fine-grain *asynchronous and pipeline* task parallelism, *hybrid* node proximity, fault tolerance, and achieved 96% efficiency of the ideal workflow performance model. Scalability experiments on 2 HPC systems using up to 13,056 cores showed that Zipper outperformed the existing fastest in-situ systems by up to 2.2 times.
   - Invented a novel **concurrent data transfer method**, which used two channels, i.e., network and parallel file system, to transfer data with a *multi-threaded work-stealing algorithm*. This method reduced the data transfer time by up to 32% when the simulation was stalled. The speedup reason was then investigated by using *OmniPath network tools*, which showed that the network congestion was alleviated by up to 80%.
2. **World's fastest** Lattice Boltzmann method (LBM) algorithm.
   11/15∼Present, Purdue University, Ph.D. Research, Related Publications: [1,4]
   Advised by Prof. Fengguang Song, Luoding Zhu, Zhiyuan Li, Xavier Trichoche
   - Designed a novel **sequential / parallel 2D & 3D memory-aware LBM** algorithms to accelerate the memory-bound LBM simulation efficiently. It combined features of loop fusion, swap (reducing half of the data storage cost in 3D), spatial blocking (Tile in 2D, Prism Traversal in 3D), and temporal blocking (merging K time steps of LBM computation). Strong scalability experiments on 3 architectures showed that 2D & 3D memory-aware LBM outperformed the existing fastest LBM by up to 5 times & 1.9 times, respectively.
   - Investigated the speedup reason using both theoretical algorithm analysis and experimental **Roofline** analysis. The arithmetic intensity (AI) of memory-aware LBM outperformed the fastest existing LBM by up to 4.6 times, and it transformed **the critical memory-bound problem into a compute-bound problem** on the Roofline chart of current CPU architectures.
   - Used **Paraview/Catalyst** to visualize the the simulation results of Karman vortex street.
3. **Task Bench**: evaluating the performance of the state-of-the-art parallel and distributed programming systems.
   05/2018∼08/2018, Los Alamos National Laboratory, Intern project, Related Publications: [2]
   Advised by Dr. Wei Wu, Elliott Slaughter, Patrick McCormick
   - Designed a **parameterized** benchmark **Task Bench** implemented in **15 parallel and distributed programming systems**, including traditional HPC message passing models (MPI, MPI+OpenMP/CUDA), PGAS / Actor models (Chapel, Charm++, X10), task-based models (OmpSs,

OpenMP, PaRSEC, Realm, Regent, StarPU), and models for data analytics, machine learning and workflows (Dask, Spark, Swift/T, TensorFlow). For N benchmarks and M programming systems, it reduced the efforts of benchmarking and comparison from **O(NM)** to **O(N+M)**.

- Introduced a novel efficiency metric **Minimum Effective Task Granularity** (**METG**) to quantify the runtime overhead and amount of useful work performed in each system. Running at scale up to 256 nodes on Cori HPC, **100** μs was the smallest granularity that the best system can reliably support with current technologies.
- Asynchronous systems have better overlap between computation and communication, and better support for complex task graphs. Compile-time optimization helps scaling for task-based models.

4. **LaunchAgent: On-demand distributed workflow framework**.
   06/2019∼Present, Purdue University, Cyberwater NSF project 1835817.
   Advised by Prof. Fengguang Song, Yao Liang, Sudhakar Pamidighantam
   - Developed an on-demand cross-platform distributed workflow Python framework **LaunchAgent**. It interacts with locally installed *VisTrails* workflow management software on researchers' desktop to enable conveniently requesting HPC/Cloud resources to accelerate their hydrology research process on demand. It manages authorization to HPC/Cloud resources, prepares/submits batch jobs to those resources, support fault tolerance, and monitors the quality of services for the users. This way, computation-intensive or data-intensive tasks from the It is **currently in use on 2 HPC systems and 2 Cloud systems** (*Google Cloud* and *Jetstream*).
   - Built an AWS ParallelCluster configured with Amazon FSx for Lustre and *NICE DCV* for remote visualization. It contained a front-end to the Slurm scheduler, which was composed of an HTTPS API and a serverless function (AWS Lambda) that would translate the HTTPS requests to Slurm commands and run them through a secure channel. Then users could monitor the HPC Cluster with *Prometheus*, use *Grafana* to query and visualize system status.
   - Built an elastic virtual cluster on XSEDE *Jetstream* resource in an *Openstack* environment. The basic structure is to have a single image act as headnode, with compute nodes managed by SLURM via the openstack API. The compute nodes use a basic CentOS 7 image, followed by an *Ansible* playbook to add software, mounts, users, config files, etc.

5. **Bio-detection** using unsupervised feature learning on limited biological dataset.
   02/2015∼08/2015, Purdue University.
   Advised by Prof. Murat Dundar
   - Object-detection on total 1833 images with three different wavelengths for 4 classes of bacteria during 3 days. The classes exhibits multi-modal distributions. Random effects due to daily variations in experimental settings can be significant. The descriptive features across different bacteria classes can be subtle and not easy to define.
   - Invented a **Kmeans-based three-stage unsupervised feature learning method** with data augmentation. First, it used multiple image scales and patch sizes to extract patches with normalization, then runs Kmeans to learn k centroids. Secondly, it used triangle encoding to obtain k dimensional assignment vectors, and sumpools these vectors, to obtain feature vectors representing images. Then it concatenated these feature vectors obtained at different patch sizes and image scales. Thirdly, it trained a SVM classifier using data on day 1, tuned its parameters using data on day 2, and tested data on day 3. Using only 400 descriptors, it achieved validation and test accuracy of **99%** and **96.4%**, respectively. This method mimicking the deep-learning behavior of multi-layer networks with improved stability, and can better tolerate random effects.
   - Compared performance with **deep learning** techniques with **Theano** implementations of convolutional neural nets, Deep neural nets with dropout, and Stacked denoising autoencoder. Even using data augmentation, they achieve **52%**, **49%** and **56%** test accuracy, respectively. They did not perform well as they rely on complex networks that are highly

prone to **overfitting** when data are limited. Two conventional handcrafted features learning methods (Zernike moments and Haralick texture features) were also conducted and achieved **78%** and **64%** accuracy. Their high sensitivity to random effects makes them less ideal for classification with multi-day data sets.

6. Minimal **ResNet** framework with efficient **GPU** kernels.
   01/2019∼05/2019, Purdue University, Course project
   Advised by Prof. Fengguang Song
   - Led a team of 4 graduate students to build a ResNet C++/CUDA framework. I I used *cuDNN* to implement the *convolution forward and backward* and global pooling GPU kernels on *Google C++ Test* framework. Results were verified by *Numpy* output.
   - Scalability experiments showed that the cuDNN implementation using forward-winograd & backward-implicit-gemm kernels outperformed other algorithms by up to 12 times. The experiment was tested with the CIFAR10 dataset on Nvidia Tesla K40c & Quadro P6000 GPU.
7. Distributed Multi-CPU sharing I/O Resource Pool on Dawning 7000 cloud system.
   06/2012∼06/2014, M.S. Thesis, Institute of Computing Technology, Beijing, China
   Advised by Prof. Xuejun An, Dr. Zheng Cao, Dr. Mian Wang
   - Designed a novel PCIe network controller using FPGA to combine distributed I/O resources including NICs, SSDs, etc. into a shared resource pool, which can be accessed by CPUs on different sockets using PCIe with isolation and flexible resource re-allocation policy. It implemented **MR-IOV** (Multi-Root I/O Virtualization) based on **SR-IOV** devices (Single-Root I/O Virtualization) among distributed PCIe domains. Thus any virtual function of a SR-IOV device could be reallocated to any virtual machine of any CPU at runtime.
   - The controller had three features: 1) *direct I/O virtualization* (building logic mirrors of the physical I/O devices on hardware to be compatible with existing hardware and software stack); 2) *direct I/O mapping* (ID mapping and address mapping of device functions in PCIe domain among root node and client nodes); 3) *distributed PICe routing* (routing PCIe data packet to the corresponding physical I/O device or remote root node).
   - Built a *NetDIOV* verification framework using system Verilog, and perform MR-IOV functional test on Xilinx Virtex6 ff365t FPGA. Bandwidth performance test using *iperTCP* shows that it achieves 1.6GB/s using 32 double words with 125MHz clock rate.
8. Design hardware system and routing algorithm of a Traveler Robot on a sedan chair.
   Ranking 2nd in Asia-Pacific Robot Contest (ABU Robocon) in China Region
   09/2008∼06/2009, leader of NEU robocon hardware team, Northeastern University, China
   Advised by Prof. Dehong Cong
   - Designed the traveler robot hardware circuit system including power supply, PIC microcontroller motherboard, self-driving and gyroscope and ultra-red sensors.
   - Implemented the traveler robot's routing algorithm, so that we can land the traveler robot from the sedan chair, then self-navigate to the Goal Zone and beat the drums by 3 times within 2 seconds.
9. Predicting the residence and next crime scene of a serial criminal.
   Meritorious Winner award of Mathematical Contest in Modeling
   10/2009∼02/2010, Team leader, Northeastern University, China
   Advised by Prof. Guangming Yang
   - Designed a novel criminal geographic targeting(**CGT**) model, which computed every location's relative probability of the criminal residence on a grid map and was visualized as a 3D contour.
   - Trained a back propagation (**BP**) neural network with *Levenberg-Marquardt* algorithm using 100 serial criminals' datasets. Then I used this model to predict the potential the serial

criminal's residence and his next crime location with 95% MSE accuracy (mean squared normalized error).

10. Digital Power Amplifier with balanced amplitude and frequency.
    Ranking Second Prize in National Undergraduate Electronic Design Contest. 09/2008∼11/2008, Northeastern University, China. Advised by Prof. Shi Zhang
    - Designed and implemented the hardware PCB system, including the pre-amplifier filter circuit, bandwidth-selection circuit, and the digital signal processing module using Altera FPGA. User can select continuous magnification of amplitude, power, and frequency.

Scholarships & Honors

- Meritorious Winner, US Mathematical Contest in Modeling (MCM), 02/2010
- Runner-up, Asia-Pacific Robot Contest (ABU Robocon) in China Region, 06/2009
- Second Prize, National Undergraduate Electronic Design Contest (China) , 09/2009
- Merit Graduate Student Award, University of Chinese Academy of Sciences, 2012∼2013
- First-class NEU Scholarship (top 2%), achieved 5 times, 2007∼2011
- Second-class NEU Scholarship (top 5%), achieved 3 times, 2007∼2011