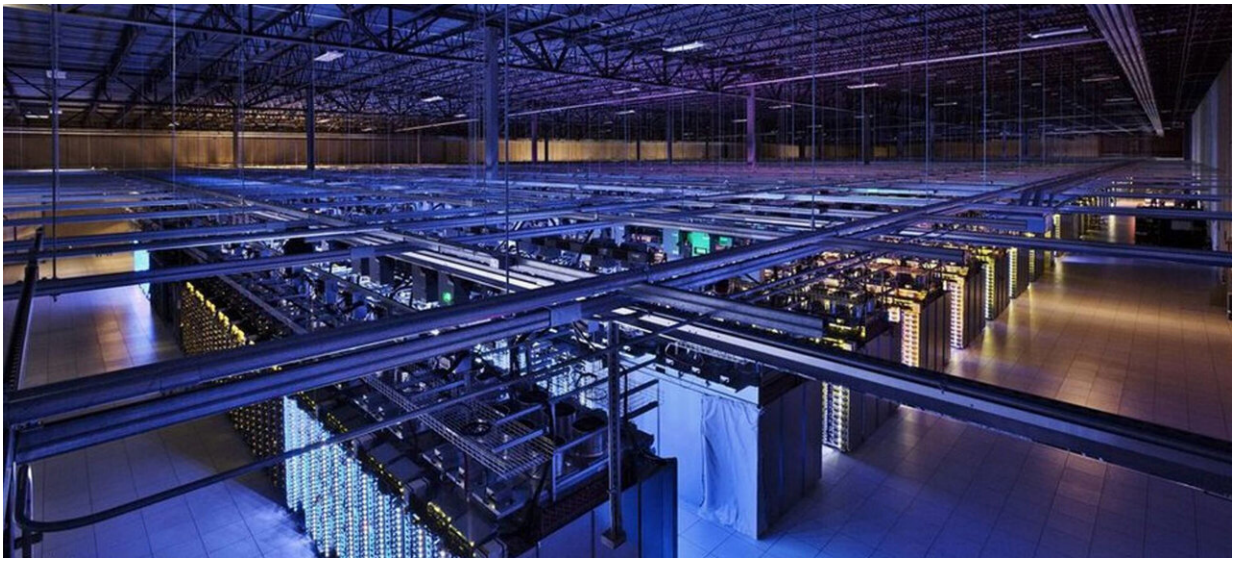
Enabling fairer data clusters for machine learning

July 20 2020, by Steve Crang



Credit: Google

Research published recently by CSE investigators can make training machine learning (ML) models fairer and faster. With a tool called AlloX, Prof. Mosharaf Chowdhury and a team from Stony Brook University developed a new way to fairly schedule high volumes of ML jobs in data centers that make use of multiple different types of computing hardware, like CPUs, GPUs, and specialized accelerators. As these so-called heterogeneous clusters grow to be the norm, fair scheduling systems like AlloX will become essential to their efficient

operation.

This project is a new step for Chowdhury's lab, which has recently published a number of tools aimed at speeding up the process of training and testing ML models. Their past projects Tiresias and Salus sped up GPU resource sharing at multiple scales: both within a single GPU (Salus) and across many GPUs in a cluster (Tiresias).

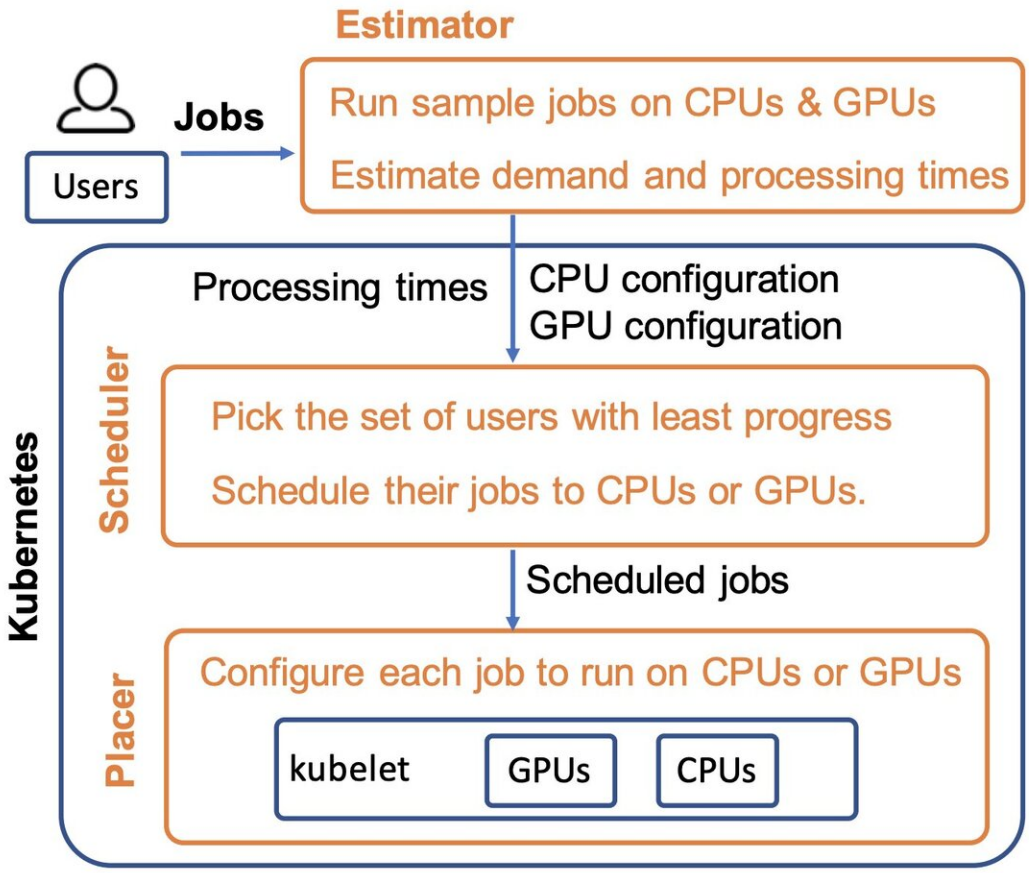
But AlloX tackles heterogeneous clusters, which carry a new problem: different hardware is best at different types of computing tasks. While there was always variety in the expectations different jobs had from the data cluster, now there's also a difference in what different hardware is best for a given job.

A number of different types of hardware may be able to run a given model, but it can have vastly different speeds on each one depending on its computation characteristics. If it requires a lot of sequential execution, then a GPU wouldn't be a good fit, but if it relies heavily on matrix multiplication then a CPU wouldn't work very well. And with new accelerators constantly under development, different common operations are always finding better options for fast execution.

"Each workload has unique characteristics that allows a unique accelerator to shine," says Chowdhury, "and that's why architecture researchers are building so many new accelerators."

This mismatched hardware raises a new aspect of the question of data cluster fairness. With a constant high volume of jobs, schedulers have to seek out not only the best average runtime to keep the center moving, but the schedule that doesn't arbitrarily tank certain jobs in pursuit of overall performance. As it stands, according to Chowdhury, the distribution of computing power leads to widely varied performance outcomes for different individual users.

"If the cluster is not fair, then some people will be penalized too much and their computation time will arbitrarily increase," he says.



The AlloX system has three main components: an estimator, a scheduler, and a placer, which together determine which hardware best serves a job and then how to most fairly fit it into what's currently available. Credit: University of Michigan

In the context of heterogeneous clusters with multiple types of computing hardware, the math involved in addressing the optimal demands of every job grows more complex.

In their solution, the researchers reformulated the problem as a bipartite matching problem, where a graph's vertices can be divided into two independent sets so that every edge connects a vertex in one to a vertex in the other. The two sets became the jobs on one hand and the total list of available computing resources on the other. They implemented this scheduler on Kubernetes, a widely used cluster manager, with a small-scale CPU-GPU hybrid cluster and large-scale simulations.

Their findings revealed that AlloX could reduce average job completion time by up to 95% when the system load is high, while still providing fairness and preventing job starvation. This performance matches or improves upon previous implementations that focus exclusively on average job runtime, without attempting to provide fairness.

Chowdhury addressed average runtimes with his group's previous system, called Tiresias. This scheduler was designed to minimize the average job completion time, taking a variety of constraints into account, but didn't account for the fairness problem. But in their paper on AlloX, the researchers demonstrate that this fast average can be achieved even with the issues of fair resource use addressed.

"We show that you can be fair and your average will still remain close to the optimized average," Chowdhury explains. "It's redistributing resources in the cluster. The average job completion time will improve, because those who were suffering the most significantly will see significant improvement."

AlloX, together with other recent projects like Salus (recently published in the MLSys conference), is one of the latest contributions of Chowdhury's group to the busy field of AI resource management. The lab's ultimate goal in this area is to optimize every step of the model hyperparameter tuning, training, testing, and inference process.

"The entire lifecycle of AI or ML, the different stages they have to go through—we want to know how best to manage every step from a resource management point of view," Chowdhury says.

AlloX was presented in the paper "AlloX: Compute Allocation in Hybrid Clusters" at the 2020 EuroSys conference.

More information: Tan N. Le et al. AlloX: Compute Allocation in Hybrid Clusters, *Proceedings of the Fifteenth European Conference on Computer Systems* (2020). DOI: 10.1145/3342195.3387547

Provided by University of Michigan