Power-Aware Heterogeneous Computing Through CPU-GPU Hybridization

Kyle Siehl Xinghui Zhao
Washington State University

Background

- GPUs capable of massive acceleration for certain tasks
- But CPU relegated to management position, leaving resources on the table
- Would be better to use the available CPU resources for actual work
- How to do so efficiently?

Balancing

- Goal is to find fraction of work (α) to offload to GPU for optimal runtime
- Given previous α and previous runtime on CPU (T_c) and GPU (T_g), can compute new α via:

$$\alpha' = \frac{W_c}{W_g + W_c}$$

- Where:

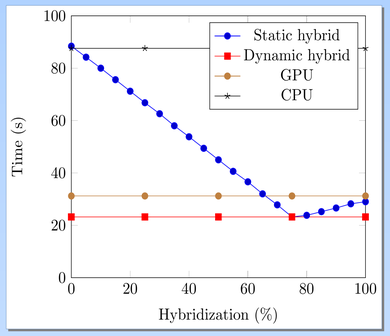
$$W_g = \frac{1}{\alpha} T_g \qquad W_c = \frac{1}{1-\alpha} T_c$$

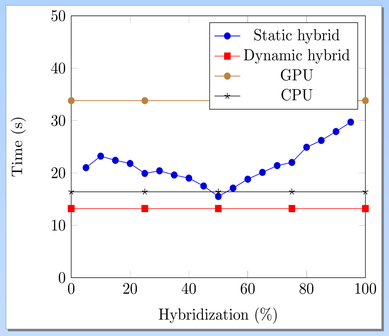
Results

1 Core CPU

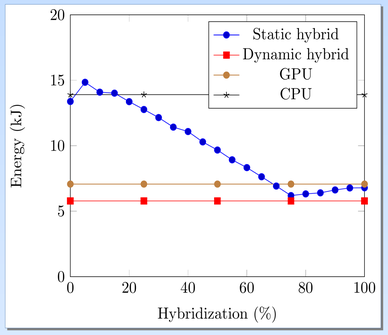
8 Core CPU

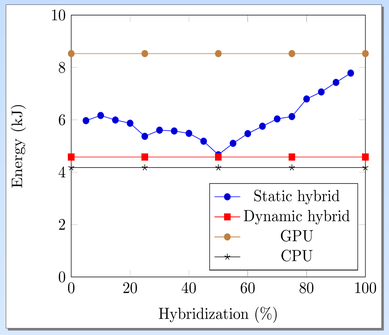
Time



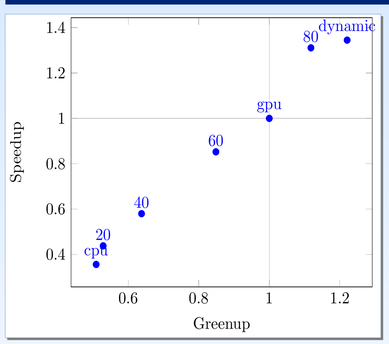


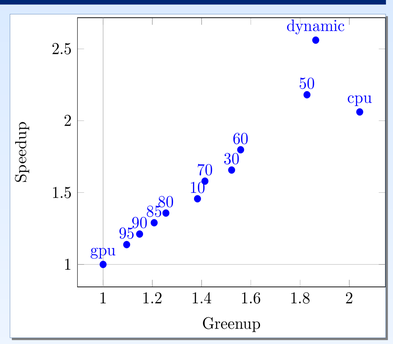
Energy



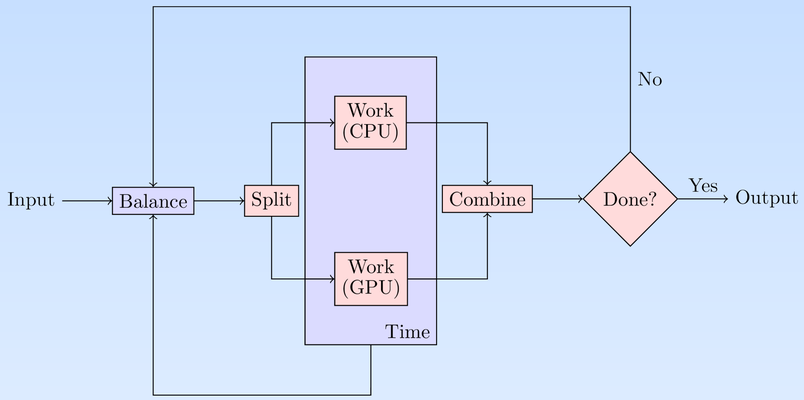


Overall Improvement





Archon System Design



- User (red) provides CPU and GPU implementations of algorithm, and splits work as necessary for problem
- Archon (blue) provides metrics and optimal load balancing

Experiments

- Using matrix multiplication as an example, we test our system's optimization in terms of both runtime and energy consumption
- We measure these results for our implementations under these circumstances:
  - using only the CPU
  - using only the GPU
  - using both the CPU and GPU with our dynamically-tuned hybridization
  - using both the CPU and GPU with statically-provided hybridization levels (from 5% GPU to 95% GPU)
- Additionally, we test each of these circumstances using two different implementation of matrix multiplication on the CPU side:
  - single-core
  - eight-core
- We apply the following metrics in each case:
  - Speedup (time taken relative to baseline GPU)
  - Greenup (energy consumed relative to baseline GPU) [Abdulsalam et al., 2015]

Hardware

- Custom machine
  - AMD FX 8350 (4.0 GHz)
  - GTX 660 (2GB GDDR5, 930 CUDA cores, 1033 MHz)
  - 32GB RAM

Conclusions

- CPU-GPU hybridization is a viable method for improving performance, without much energy cost
- Archon is a good framework for optimizing client programmer's code without much user effort