Preemptive Thread Block Scheduling with Online Structural Runtime Prediction for Concurrent GPGPU Kernels

Sreepathi Pai[1], R. Govindarajan[2] and Matthew J. Thazhuthaveetil[2]

[1]The University of Texas at Austin [2]Indian Institute of Science, Bangalore

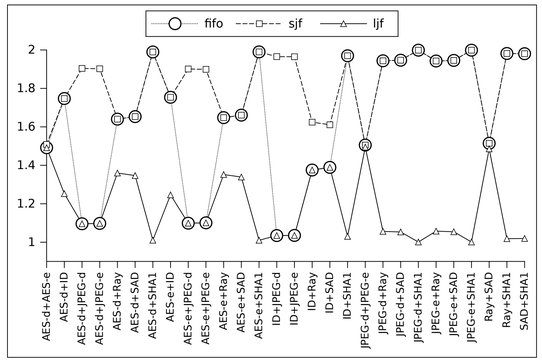
Concurrent Kernels

Current GPUs support concurrent execution of kernels

- GPU Concurrency is *space-sharing* not *time-sharing*
- Works only for small kernels
- Large kernels execute serially in FIFO order

The Problem with FIFO Scheduling

As an example, consider FIFO scheduling on 2-program workloads



- FIFO behaves like Shortest Job First (SJF) or Longest Job First (LJF)
- FIFO throughput is lower (average 15%) than SJF

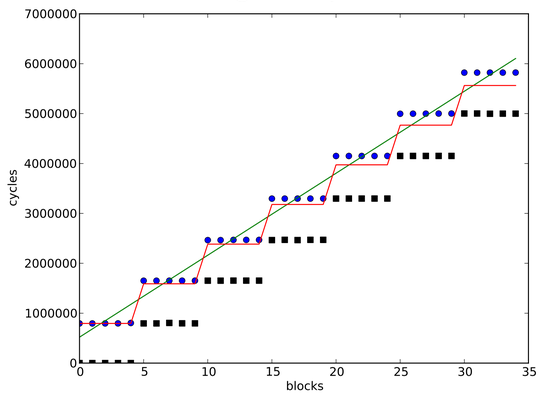
Replacing FIFO with SRTF

- GPU Kernels execute as discrete *grids* of thread blocks
- Thread blocks are independent units of execution
- Kernels can be pre-empted easily at thread block boundaries

But SRTF requires runtime of kernels.

Predicting Runtimes by Harnessing Regularity

SGEMM on one Streaming Multiprocessor of a GPU



Green Line = Linear fit, Red Line = Model fit

Structural Runtime Prediction

Structural Prediction treats the execution of n thread blocks as n repeated executions of the same code.

- Observe runtime for first few thread blocks of kernel
- Use model and observations to predict runtime for kernel

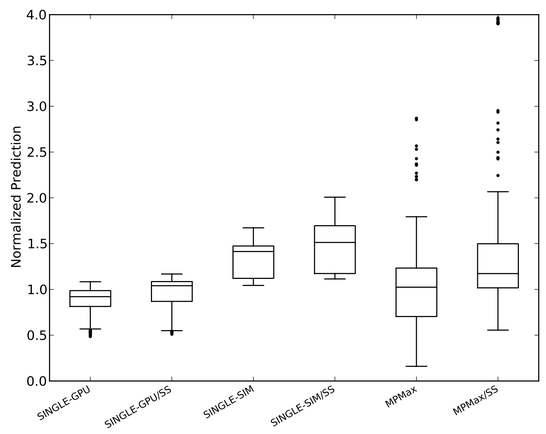
Simple Slicing Predictor

$$Pred\_Cycles = Active\_Kernel\_Cycles + \frac{(Total\_Blocks - Done\_Blocks) \times t}{Resident\_Blocks}$$

where:

$Active\_Kernel\_Cycles$	executed cycles
t	duration of thread block
$Total\_Blocks$	total thread blocks
$Done\_Blocks$	blocks until now
$Resident\_Blocks$	kernel residency
$Pred\_Cycles$	prediction

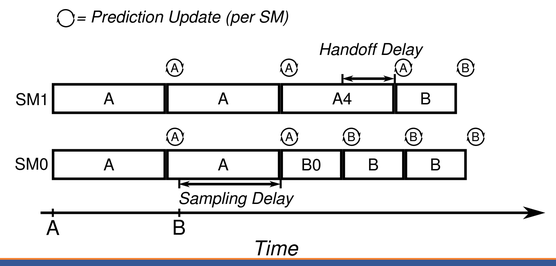
Accuracy of Predictor



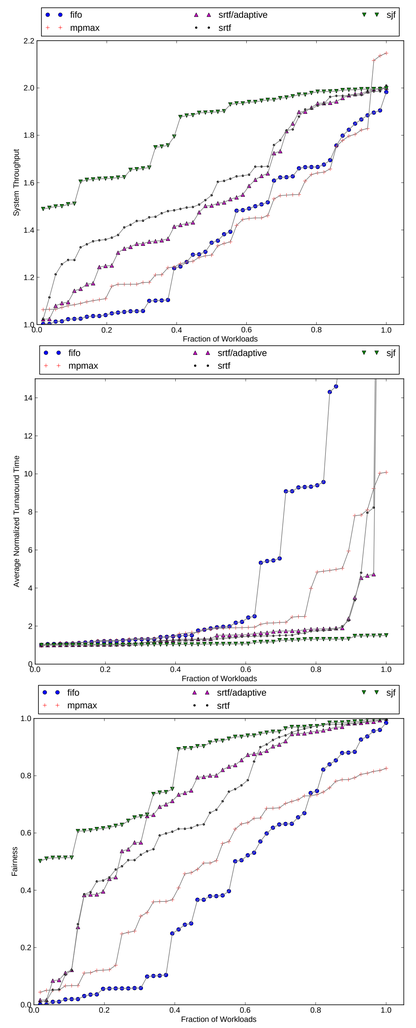
For single-gpu, the predictions are between 0.48x to 1.08x of actual runtime after observing 1 thread block.

SRTF Thread Block Scheduler (TBS)

Our TBS samples t for waiting kernels, predicts their runtime and pre-empts running kernel per SRTF policy.



Results on all 2-program ERCBench workloads



Scheduler	STP	ANTT	Fairness
FIFO	1.35	3.66	0.19
MPMax	1.37	2.15	0.36
SRTF	1.59	1.63	0.52
SRTF/Adaptive	1.51	1.64	0.56
SJF	1.82	1.13	0.80

Table : Geomean System Throughput (STP), Average Normalized Turnaround Time (ANTT) and Fairness for various scheduling policies. Note that ANTT is a lower-is-better metric.

Conclusion

- SRTF is superior in terms of system throughput, turnaround time and fairness.
- SRTF improved STP by 1.18x and ANTT by 2.25x compared to FIFO
- SRTF also outperformed resource-sharing MPMax by 1.16x (STP) and 1.3x (ANTT).
- SRTF bridges 49% of the gap between FIFO and SJF, approaching to within 12.64% of SJF's throughput.