Processing Data Streams with Hard Real-Time Constraints on CPU/GPU Systems

Technion Israel Institute of Technology

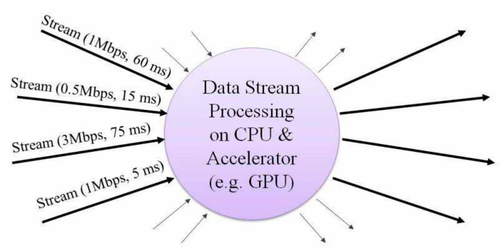
Uri Verner, Prof. Assaf Schuster, Prof. Avi Mendelson, Dr. Mark Silberstein

Technion Israel Institute of Technology

DEFINITION

Problem

Hard real-time stateful processing of multiple data streams on GPU-based systems.



Goal

High throughput framework with hard real-time guarantees.

MOTIVATION

Hard real-time streams applications:

- RT critical medical & defense systems, production control, etc.
- RT by contract SSL, VoIP, IPTV, media, games, interactive.

CHALLENGES

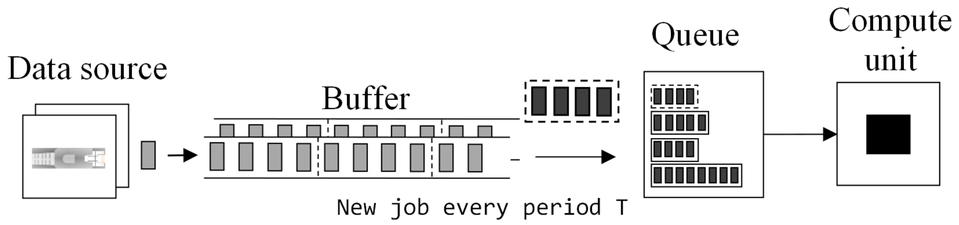
Hard real-time on GPU:

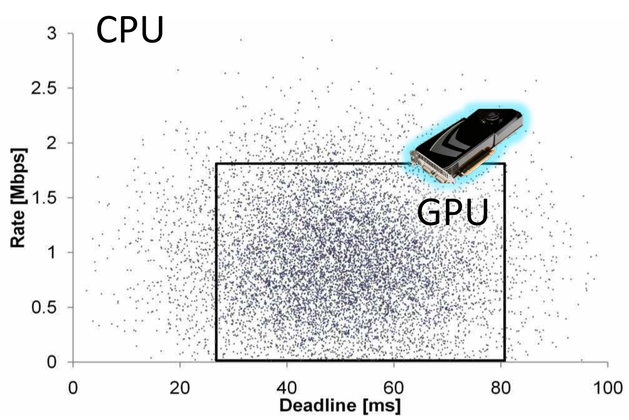
- Low per-stream performance
- Performance sensitive to workload irregularity
- Weak CPU/GPU memory model

Multi-GPU:

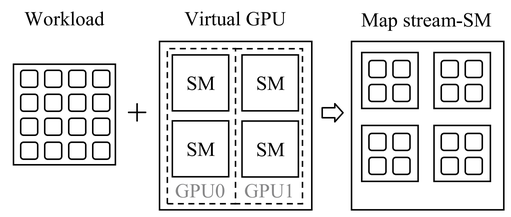
- Efficient work distribution
- Limited bandwidth of CPU/GPU interconnect
- Communication scheduling

RECTANGLE METHOD





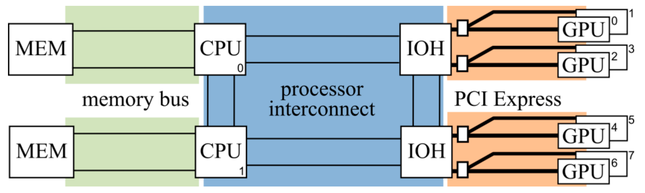
A workload of 12,000 RT streams



Multi-GPU

COMMUNICATION SCHEDULING – THE BATCH METHOD

The CPU-GPU communication time is influenced by concurrent traffic on the interconnect. As a result, the worst-case execution time of every job that is used by classic hard RT scheduling algorithms is very pessimistic.



The Batch method combines a set of CPU-GPU data transfers in a single job. The method also provides an algorithm for computing the execution time of a batch, hence a batch can be scheduled as a job using classic algorithms such as EDF and RM.

Using the Batch method, one can efficiently schedule the CPU-GPU communication by binding individual CPU-GPU data block transfers into batch messages, and scheduling the batch messages using EDF.

CONTACT INFO

uriv@cs.technion.ac.il

EVALUATION & RESULTS

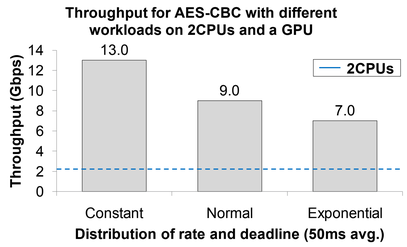
Single GPU

Application: AES-CBC Encryption

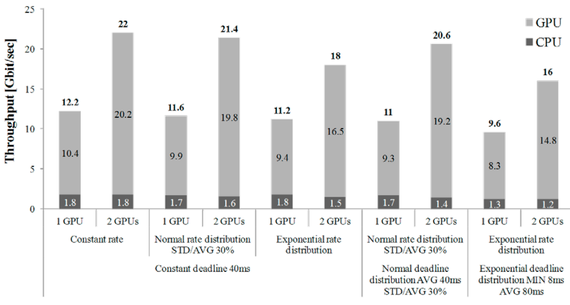
Platform: NVIDIA GTX 285 GPU + Intel Core2 Quad 2.33Ghz + 3GB/s PCIe

(1 core dedicated to data generation)

Workload: 12,000 streams, different distributions of rate & deadline



Multi-GPU



Batch Method

Tested two realistic applications on two multi-GPU systems.

Domain decomposition: 7.9x shorter execution time than bandwidth allocation

Wafer inspection: 39% higher image resolution than time division.