

Санкт-Петербургский государственный университет
Математическое обеспечение и администрирование
информационных систем

Системное программирование

КУРСОВАЯ РАБОТА

Реализация программной транзакционной памяти на архитектуре CUDA

СТУДЕНТ 344 ГРУППЫ
АЛЕКСАНДР МИСОНИЖНИК

НАУЧНЫЙ РУКОВОДИТЕЛЬ:
СТАНИСЛАВ ЮРЬЕВИЧ САРТАСОВ

Введение

Транзакционная память — технология синхронизации конкурентных потоков

- Упрощает параллельное программирование, позволяя выделять группы инструкций в атомарные транзакции
- Конкурентные потоки работают параллельно, пока не начинают модифицировать один и тот же участок памяти

Цель работы: реализация алгоритма STM на архитектуре CUDA

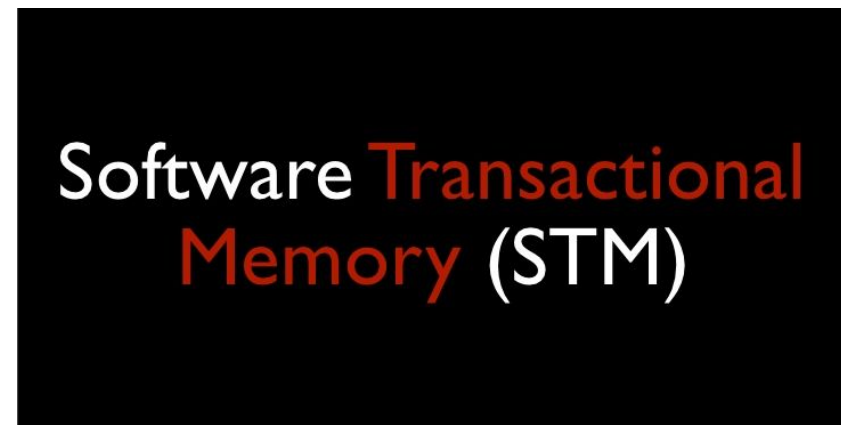
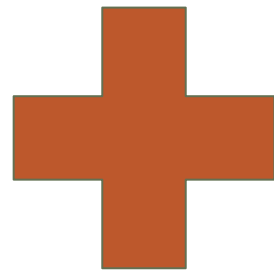
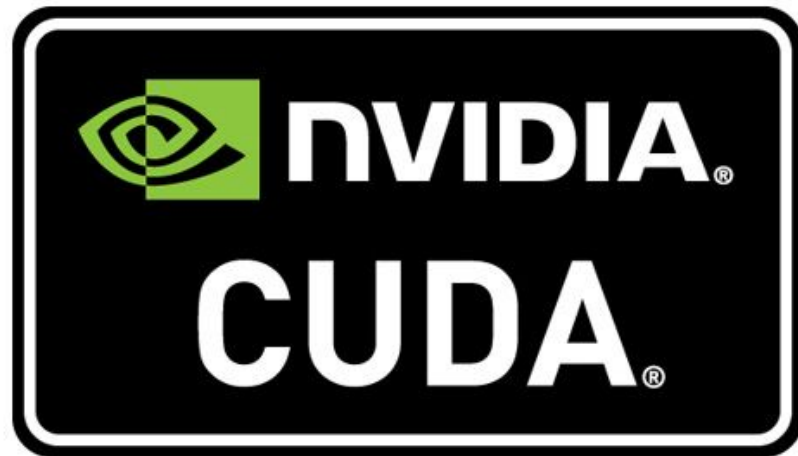
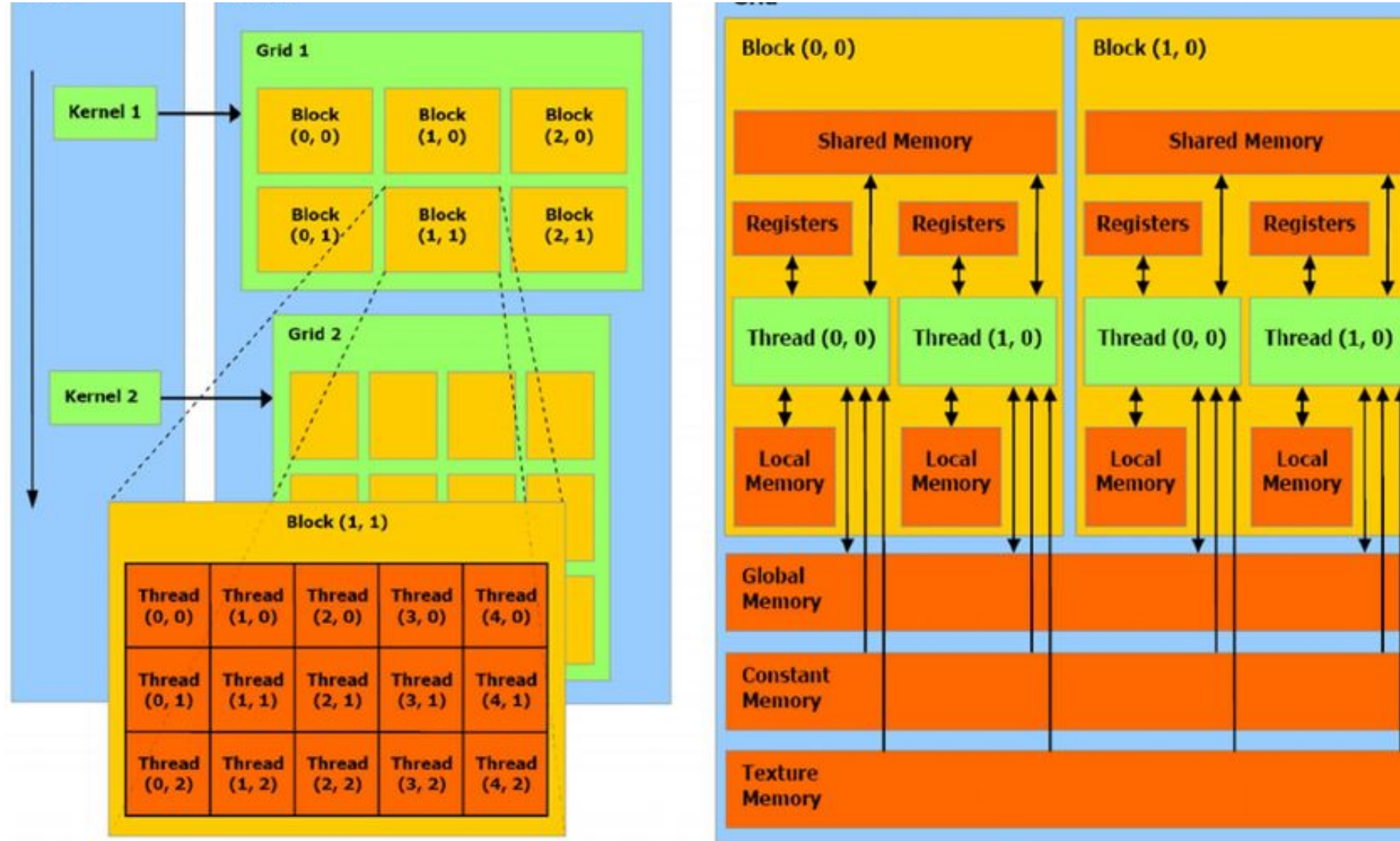


Схема архитектуры памяти CUDA



Известные алгоритмы STM для GPU

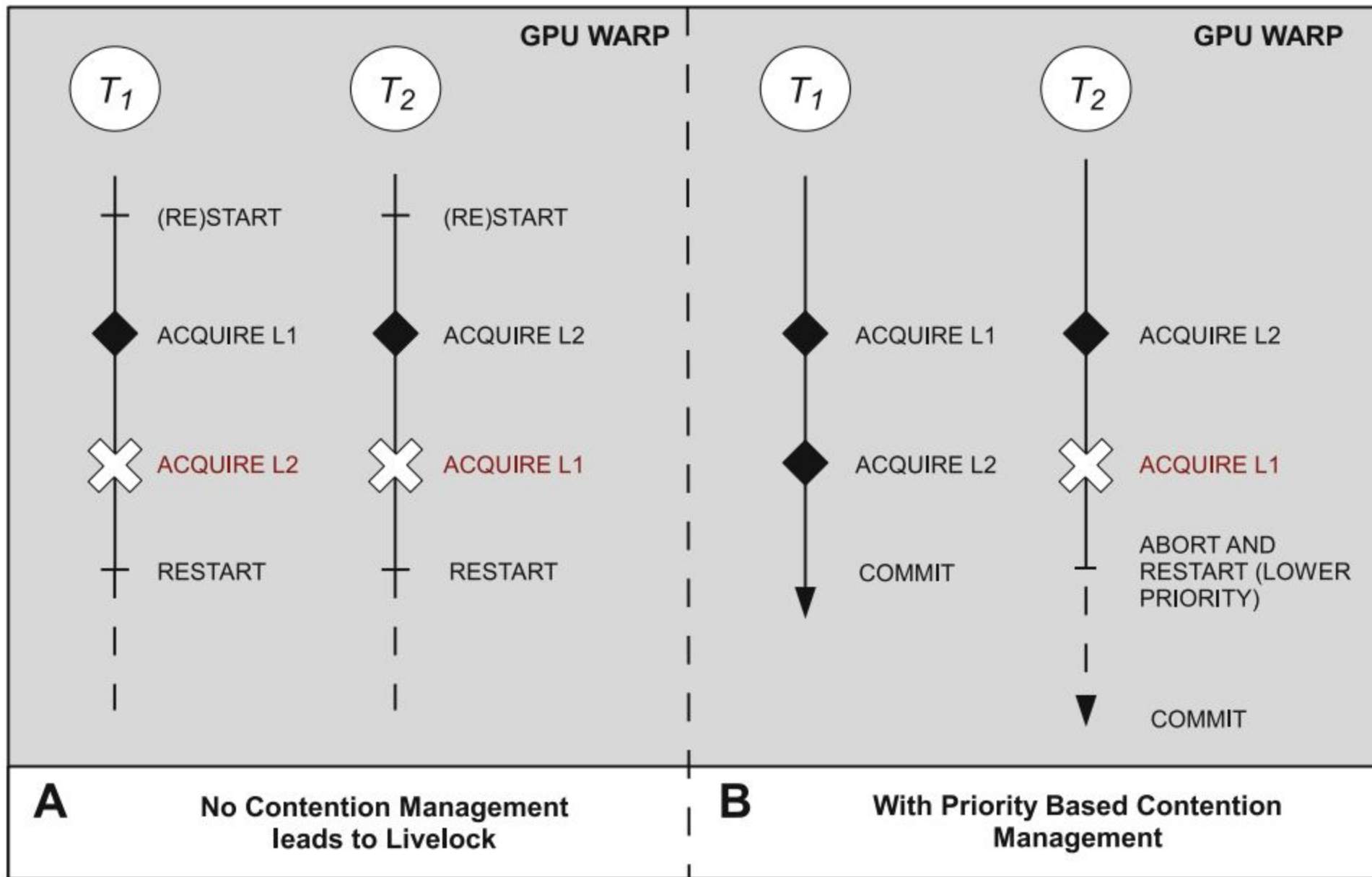
- GPU-STM - сложен в реализации, отсутствует *opacity*, присутствует нетривиальная операция с большой вычислительной сложностью
- Lightweight Software Transactions on GPUs - отсутствие универсальности
- PR-STM - относительная простота реализации, *opacity*, относительно низкие расходы на память - свобода модификации. Идеально!

Метаданные:

- Глобальные - Global Lock Table
- Локальные - Local Read Set, Local Write Set, Local Lock Set

Функции:

- txStart, txRead, txWrite, txValidate, txCommit, releaseLocks



Модификация

- Возможность использования одной Global Lock Table для разных типов данных

Результаты:

- Исследованы и проанализированы теоретические результаты в области STM для GPU
- Написана и протестирована на работоспособность одна из реализаций STM для CUDA
- При реализации были применены и сравнены различные подходы в организации данных с учетом архитектуры CUDA
- Придумана и реализована модификация для этой реализации, которая упрощает работу с STM