ФРТК

iSCALARE

intel

Лаборатория суперкомпьютерных технологий для биомедицины, фармакологии и малоразмерных структур

Параллельная симуляция
часть 3

Григорий Речистов

grigory.rechistov@phystech.edu

15.04.2013

- Параллельные модели МП систем
- Модели консистентности памяти и симуляция
- The state of the art

Параллельная симуляция параллельных процессоров

- Эквивалентна PDES
- Терминология меняется:
  - LP — потоки
  - События — отдельные инструкции
  - Пересылка сообщений — доступы в общую память

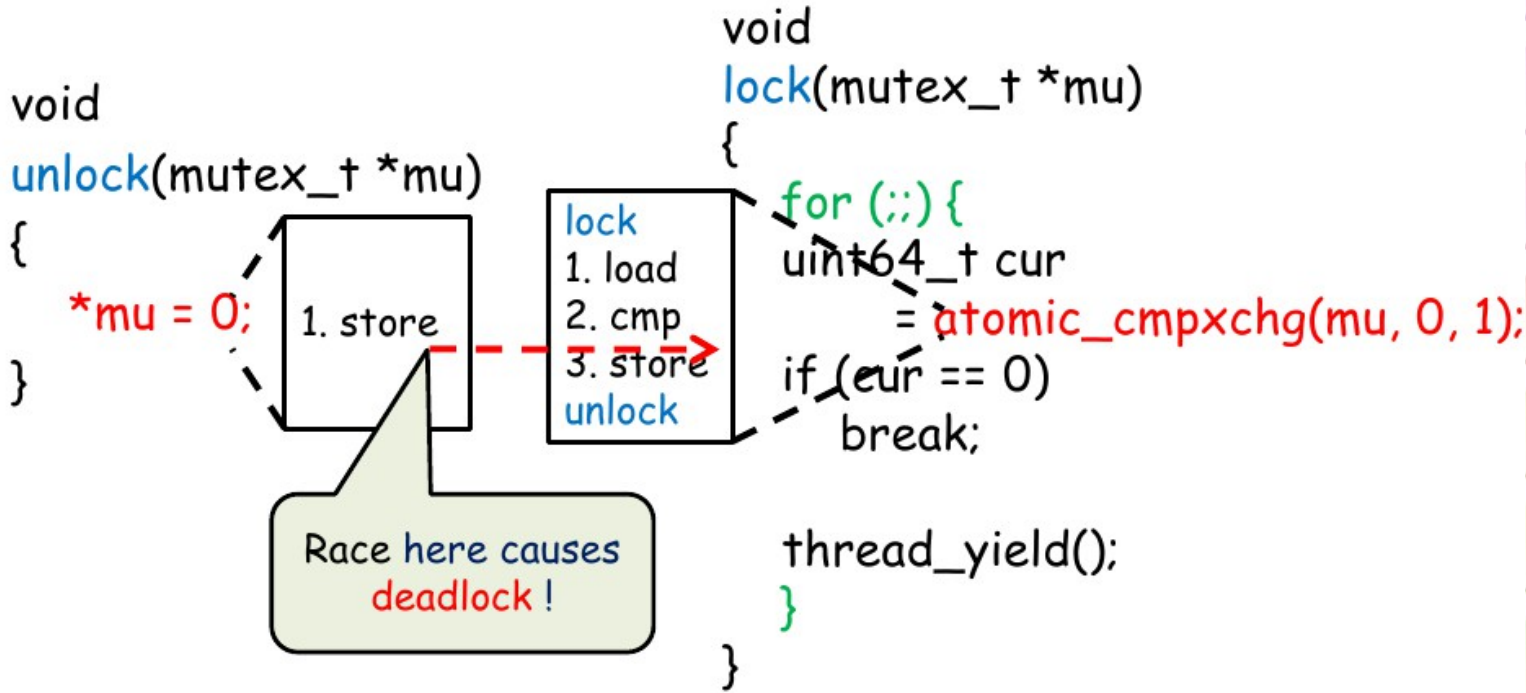
Реальные системы

- Процессоры взаимодействуют между собой посредством чтения/записи общей памяти
  - Порядок чтения/записи на процессоре/между процессорами
  - Атомарные инструкции
  - Гранулярность доступов

Атомарные инструкции

1. Симуляция: через атомарные хозяйские
   - Но архитектуры имеют разное число/типы атомарных инструкций!
2. Использование критической секции в хозяйском коде
   - Не работает в общем случае!

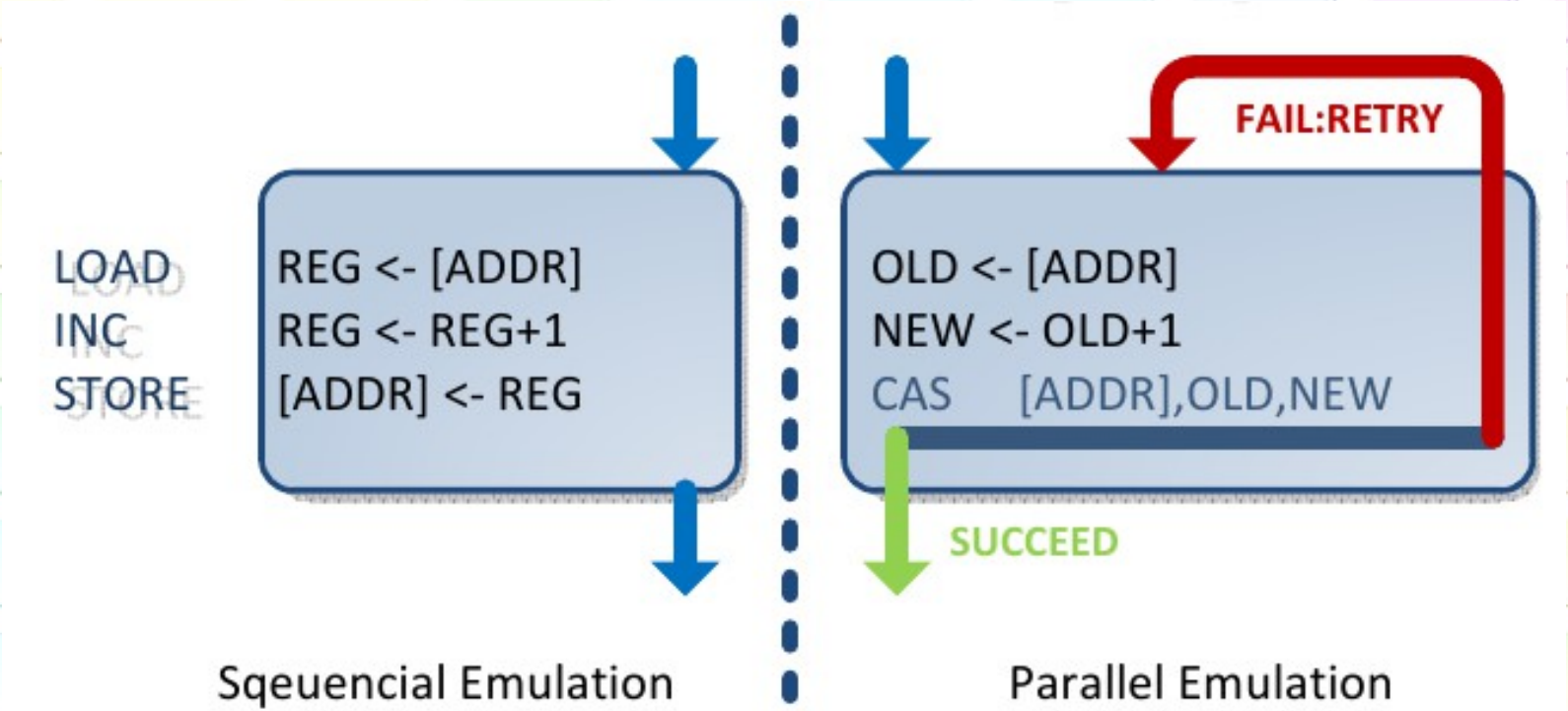
Почему это не работает



Придётся защищать локом **все** доступы в память!

Источник: Wang et al.

Решение — транзакции

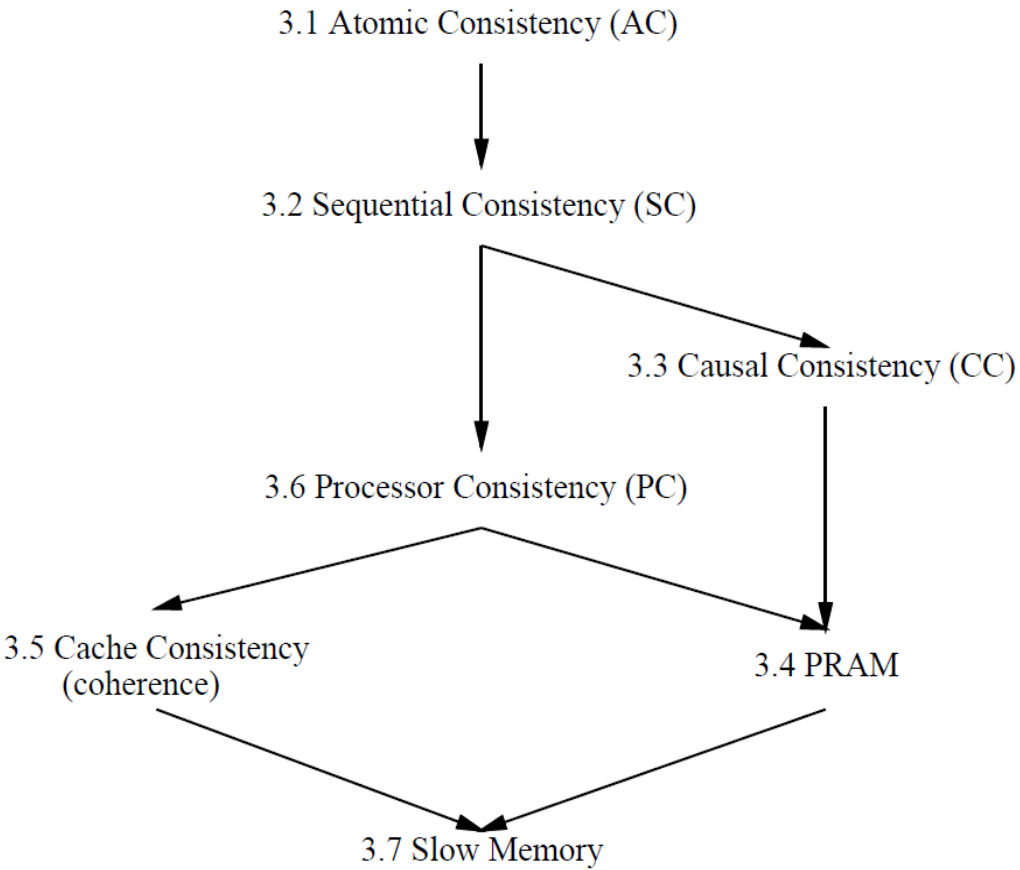


Источник: Wang et al.

Порядок load/store/instruction

- Порядок чтений/записи, который наблюдают потоки:
  - Для исполняющихся инструкций
  - Для локальных чтений/записей данных
  - Для удалённых чтений/записи данных

Модели консистентности



Источник: David Mosberger, **Memory Consistency Models**

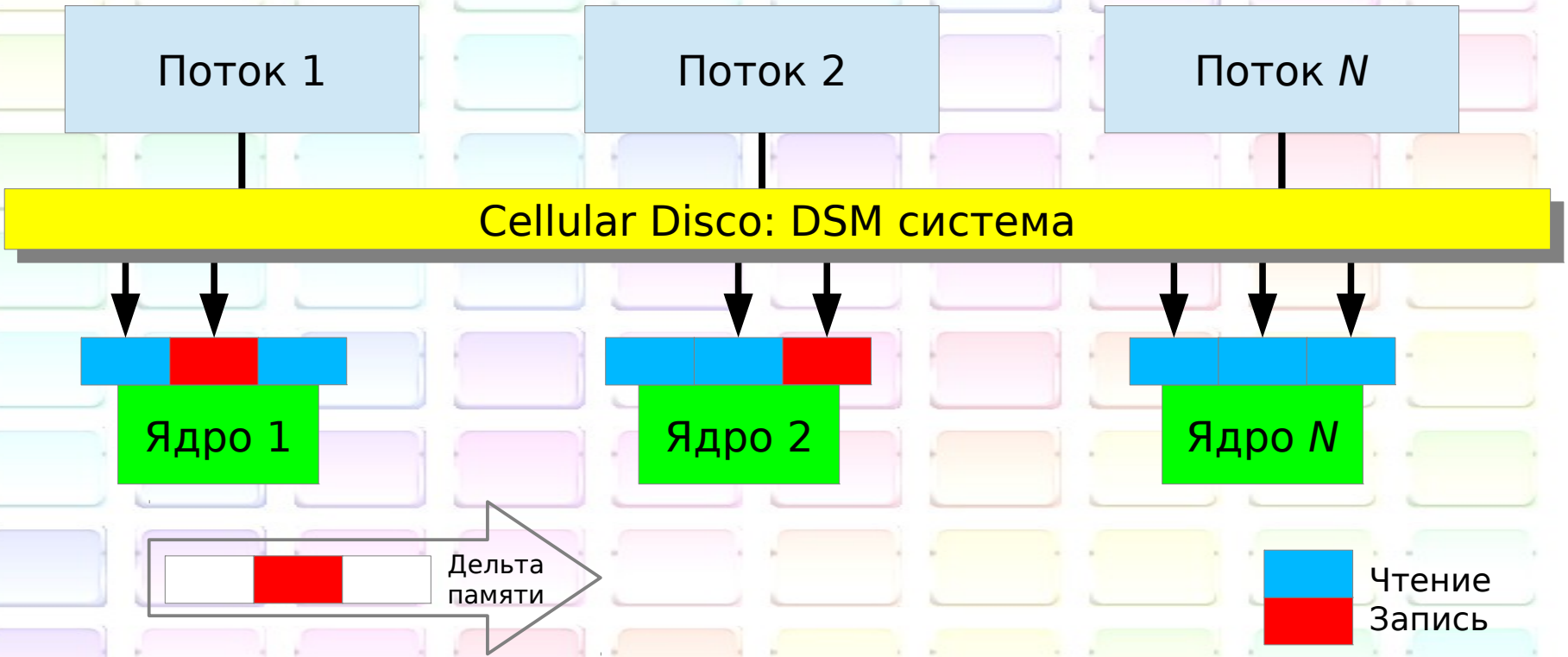
Архитектуры ЦПУ и их модели консистентности

- Инструкции-барьеры
  - IA-32: lfence, mfence, sfence, cpuid
  - IA-64: ld/st.acq, ld/st.rel, mf

	Loads Reordered After Loads?	Loads Reordered After Stores?	Stores Reordered After Stores?	Stores Reordered After Loads?	Atomic Instructions Reordered With Loads?	Atomic Instructions Reordered With Stores?	Dependent Loads Reordered?	Incoherent Instruction Cache/Pipeline?
Alpha	Y	Y	Y	Y	Y	Y	Y	Y
AMD64	Y			Y				
IA64	Y	Y	Y	Y	Y	Y		Y
(PA–RISC)	Y	Y	Y	Y				
PA–RISC CPUs								
POWER	Y	Y	Y	Y	Y	Y		Y
SPARC RMO	Y	Y	Y	Y	Y	Y		Y
(SPARC PSO)			Y	Y		Y		Y
SPARC TSO				Y				Y
x86	Y	Y		Y				Y
(x86 OOStore)	Y	Y	Y	Y				Y
zSeries				Y				Y

Источник :McKenney Paul E. **Memory ordering in modern microprocessorsë**

Корректность симулируемой программы => оптимизации симуляции



Гранулярность доступов

- Размер страниц/кэш линий
- (Не)выровненные доступы в память и атомарность

Опять lock holder preemption

- Возможность вытеснения планировщиком гостевого процесса, держащего блокироку => остальные процессу будут простаивать
  - Модификация гостя (hint'ы)
  - Детектирование моментов входа в критические секции (**`pause`**) и принятия решений о перепланировке процессов

Реализация в современных продуктах

- ~~Многие пытались, немногие преуспели~~
- Нет универсального решения

Simics

- Доменная схема синхронизации
  - Минимальная единица параллельного исполнения — материнская плата
- Процессоры внутри исполняются последовательно
- Детерминистичная система

BigSim

- Система поддержки разработки для IBM BlueGene
- Программная модель Charm++ (AMPI)
- Оптимистическая схема

Graphite

- Симулятор уровня приложений IA-32 (Debian Lenny)
- Основан на Pin
- Консервативная схема с пересылкой меток времени
- Distributed shared memory
- Недетерминистичный

Ещё:

- Parallel Embra (x.: SGI Origin)
- PQEMU (г.:ARM11)
- Sulima (г.: Sparc)
- ArcSim (г.: ARCompact)

Что можно ещё параллелить в симуляторе

- Модели процессоров
  - Отдельный поток для ДТ
    - Но: аккуратно управлять кэш(ами) трансляций
    - Разделяемый кэш трансляций
- Модели периферии — один поток на устройство
  - Полезно для задач с интенсивным IO

Итог

- Симуляция атомарных инструкций
- Симуляция модели консистентности
- Возможность lock holder preemption

Рекомендуемая литература (1/2)

I.Böhm, B.Franke, N.Topham. **Cycle-Accurate Performance Modelling in an Ultra-Fast Just-In-Time Dynamic Binary Translation Instruction Set Simulator**

http://groups.inf.ed.ac.uk/pasta/papers/SAMOS10_JIT_DBT_ISS.pdf

Wang et al. **COREMU: A Scalable and Portable Parallel Full-system Emulator**

http://sourceforge.net/p/coremu/home/Home/

http://software.intel.com/file/30148 (ugh!)

PQEMU: A Parallel System Emulator Based on QEMU / JiunHung Ding, Po-Chun Chang, Wei-Chung Hsu, Yeh-Ching Chung // Proceedings of the 2011 IEEE 17th International Conference on Parallel and Distributed Systems. — Washington, DC, USA: IEEE Computer Society, 2011. — Pp. 276–283.

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6121288

Рекомендуемая литература (2/2)

Edouard Bugnion et al. **Disco: Running commodity operating systems on scalable multiprocessors**

www.cis.upenn.edu/~cis700-6/04f/papers/bugnion-disco.pdf

Jason E. Miller, Harshad Kasture, George Kurian, Charles Gruenwald III, Nathan Beckmann, Christopher Celio, Jonathan Eastep, Anant Agarwal. **Graphite: A Distributed Parallel Simulator for Multicores**

http://groups.csail.mit.edu/carbon/docs/graphite_hpca2010_preprint.pdf

David Mosberger. **Memory Consistency Models**

http://www.eecg.utoronto.ca/~bilas/ece1755/papers/arizona.ps

McKenney Paul E**. Memory ordering in modern microprocessors, Part I**
http://www.linuxjournal.com/article/8211

На следующей лекции:

- Потактовая симуляция

  Моделирование систем, в которых состояние **многих** агентов меняется **каждый** такт

Спасибо за внимание!

Все материалы курса выкладываются на сайте лаборатории:
http://iscalare.mipt.ru/material/course_materials/