

УДК 06.055.2 + 061.3 + 002.2 + 070.43

## СОВЕРШЕНСТВОВАНИЕ ПРОГРАММНО-АППАРАТНОЙ БАЗЫ КЛЕТОЧНЫХ АВТОМАТОВ

*Коноплева А.П.**Донецкий национальный технический университет*

*Рассматриваются вопросы реализации клеточных автоматов на базе CUDA и GPU. Предлагаются варианты аппаратной реализации клеточного автомата в виде сопроцессора или сетевого ресурса.*

### Введение

В работах [1-3] были реализованы варианты программ моделирования клеточных автоматов в среде Adobe Flash на языке Action Script и в среде Visual Studio 2008 на языках C++ и C#. Рассматривалось поле клеток размерностью не более 64x64, т.к. при попытке расширения размерности клеточного поля возникала проблема существенного снижения быстродействия выполнения программы из-за значительного возрастания количества вычислительных операций. В связи с этим возникла задача увеличения быстродействие работы программы моделирования клеточных автоматов (КА), в том числе постбинарных (ПКА) на полях большой размерности.

Наиболее эффективными способами для ускорения КА является:

- параллельная реализация на современных графических процессорах;
- аппаратная реализация.

### 1 Особенности параллельной программной реализации

Программная реализация ПКА состоит из 2-х частей: функция генерации первого поколения клеток на поле при помощи координат в формате гиперкод и функция выполнения алгоритма ПКА. В алгоритме генерации первого поколения клеток использовались рекурсивные вызовы функций. К сожалению, одним из ограничений CUDA является невозможность использования рекурсии. Поэтому параллельная реализация данной части ПКА пока остается нерешенной. Эту часть алгоритма целесообразно реализовывать на CPU. Однако хорошо поддается распараллеливанию на CUDA вторая часть – непосредственно алгоритм работы правил ПКА. CUDA использует модель SIMT, когда параллельно каждым процессором выполняется один и тот же поток инструкций над разными элементами данных. В нашем случае в качестве данных выступают ячейки ПКА, каждая из которых может быть в 4 различных состояниях. Поток для ПКА выполняет вычисления в соответствии с алгоритмом реализации локальных правил ПКА.

GPU является вычислительным устройством, сопроцессором (device) для центрального процессора (host), обладающим собственной памятью и обрабатывающим параллельно большое количество потоков. Ядром (kernel) называется функция для GPU, исполняемая потоками.

На рисунке 1 приведён алгоритм реализации ПКА на CUDA. Алгоритм состоит из двух частей: рис. 1 а) – функции, выполняемые на CPU, рис. 1 б) – функции, выполняемые на GPU.

### 2 Результаты моделирования ПКА на CPU и CUDA

Был проведен ряд исследований для полей ПКА размерностью 32\*32 (1024 клетки), 32\*64 (2048 клетки), 64\*64 (4096 клетки), 64\*128 (8192 клетки), 128\*128 (16384 клетки), 128\*256 (32768 клетки), 256\*256 (67840 клетки). Исследования проводились на базе программ, реализующих КА на CPU (CPU1), ПКА на CPU (CPU2), КА на GPU с использованием разделяемой памяти (GPU1) и на КА на GPU без использования разделяемой памяти (GPU2).

Проведенный анализ полученных результатов работы алгоритмов позволяет сделать вывод о том, что уже при размерности поля КА 256\*256 использование CUDA и GPU позволяет уменьшить

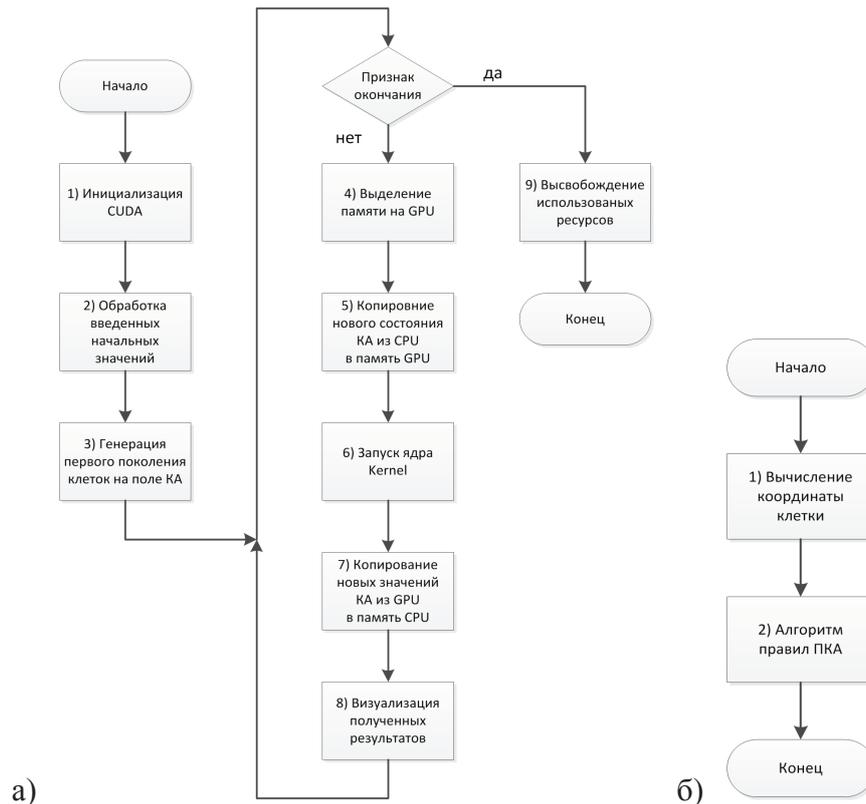


Рисунок 1. Алгоритм реализации КА на CUDA: а) функции, выполняемые на CPU; б) функции, выполняемые на GPU.

время выполнения алгоритма КА в 3,5 раза. При увеличении размерности поля, это значение будет увеличено в разы.

Однако, наиболее эффективным по быстродействию вариантом КА может быть его аппаратная реализация.

### 3 Аппаратная реализация постбинарных клеточных автоматов

Структурные схемы аппаратной реализации КА впервые предложены в работах Тоффолли и Марголуса [4]. В работе [3] рассмотрена возможность синтеза ПКА.

Поскольку КА является параллельной архитектурой, вне зависимости от количества ячеек время переключения всего устройства за один шаг будет в общем случае равно 1 такту работы устройства.

Нетривиальной задачей является запись начальных условий КА. Этот этап работы может быть организован несколькими способами:

- параллельно;
- последовательно;
- поэтапно.

Параллельная запись – наиболее выигрышна по времени, однако с возрастанием количества ячеек одновременная запись в устройство становится трудно реализуемой аппаратно.

Время последовательной, побитной, записи в устройство напрямую зависит от количества ячеек автомата.

Поэтапная запись может быть организована различными способами: по байту, по 2 байта, по 32 бита, по 64 бита.

В таблице 1 приведены расчеты времени выполнения возможных реализаций записи начальных условий для различных размерностей полей.

Аналогичным образом в КА может быть организовано считывание результатов из КА на каждом шаге для визуализации этих значений на экране.

Таблица 1. Зависимость времени выполнения записи в КА от размерности поля и типа организации записи

Тип записи/ Размерность	Парал- лельно, мкс	Последо- вательно, мкс	Поэтапно, мкс			
			1 байт	2 байта	32 бита	64 бита
32*32	1	1024	128	64	32	16
32*64	1	2048	256	128	64	32
64*64	1	4096	512	256	128	64
64*128	1	8192	1024	512	256	128
128*128	1	16384	2048	1024	512	256
128*256	1	32768	4096	2048	1024	512
256*256	1	65536	8192	4096	2048	1024

Для структурной схемы ПКА в сравнении с КА характерно увеличение разрядности в 2 раза (рис. 2). Это обусловлено необходимостью кодирования 4 состояний ПКА.

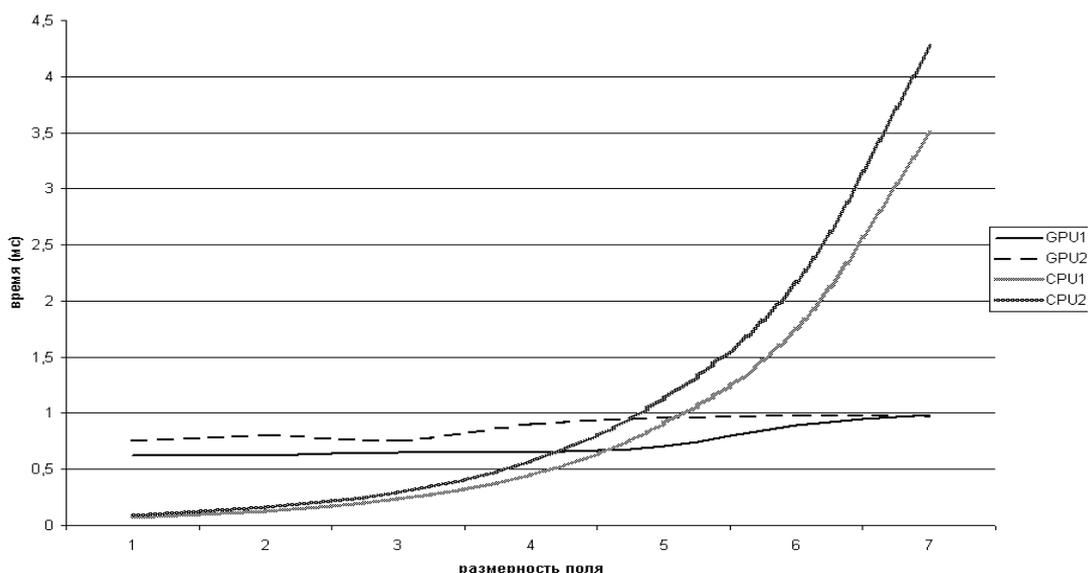


Рисунок 2. Быстродействие КА на CPU (CPU1), ПКА на CPU (CPU2), GPU без разделяемой памяти, GPU с разделяемой памятью при различных размерностях: 1 – 32\*32, 2 – 32\*64, 3 – 64\*64, 4 – 64\*128, 5 – 128\*128, 6 – 128\*256, 7 – 256\*256

Актуальной является также задача ускорения записи начальных данных и считывание результатов для визуализации получаемых значений.

Предлагается аппаратный метод записи начальных значений КА в виде каскада регистров. Графическое описание работы каскада записи начальных значений приведено на рис. 3, а.

На рис. 3, б – структурная схема каскада записи начальных значений (КЗНЗ) для поля размерностью 4\*4.

При такой организации записи начальных значений весь каскад и, следовательно, все поле ПКА будет заполняться за 1 такт, поскольку все регистры работают параллельно.

Аналогичным образом может быть организован каскад считывания результатов.

В целом КА и ПКА могут быть организован как сопроцессор к обычному компьютеру. В этом случае с компьютера пользователем будут вводиться начальные данные и визуализироваться результаты работы КА на экран.

КА и ПКА могут быть также организованы как сетевой ресурс.

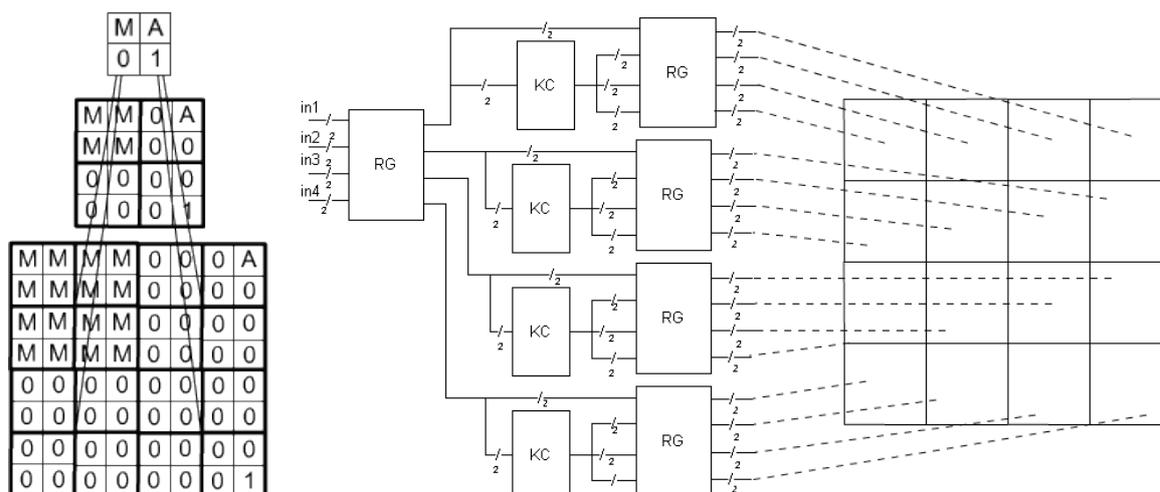


Рисунок 3. Каскад записи начальных значений ПКА:  
а) принцип работы КЗНЗ; б) структурная схема КЗНЗ

### Выводы

В статье рассмотрены варианты параллельной программной реализации КА на базе CUDA. В ходе анализа полученных результатов для КА размерностью  $32 \times 32$ ,  $32 \times 64$ ,  $64 \times 64$ ,  $64 \times 128$ ,  $128 \times 128$ ,  $128 \times 256$ ,  $256 \times 256$ , были сделаны выводы о том, что уже при размерности поля КА  $256 \times 256$  использование CUDA и GPU позволяет уменьшить время выполнения алгоритма КА в 3,5 раза.

Однако наиболее эффективным способом ускорения КА является его аппаратная реализация. В этом случае вся схема КА будет срабатывать за 1 такт работы устройства. В статье рассмотрена возможность ввода начальных значений КА при помощи каскада записи начальных значений.

### Литература

- [1] Аноприенко А.Я., Коноплева А.П., Василенко А.Ю. Оценка производительности при моделировании постбинарных клеточных автоматов и способы ее повышения // Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника (ИКВТ-2009) выпуск 147: - Донецк: ДонНТУ, 2009. С. 96-104.
- [2] Аноприенко А. Я., Коноплева А. П. Моделирование постбинарных клеточных автоматов // Збірник наукових праць. Спеціальний випуск. Матеріали міжнародної наукової конференції «Моделювання 2010» (12-14 травня 2010 р.). Том 2., - Київ: Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. НАН України. 2010. С. 162-170
- [3] Коноплева А.П. Способы формального описания классических и постбинарных клеточных автоматов // Наукові праці Донецького національного технічного університету. Випуск 9 (149). Серія «Проблеми моделювання та автоматизації проектування». – Донецьк: ДонНТУ – 2011. С. 184-194.
- [4] Тоффоли Т., Марголус Н. Машины клеточных автоматов. [Электронный ресурс]. – Издательство «Мир», Москва, 1991. Режим доступа – <http://www.kodges.ru/15136-mashiny-k-letochnykh-avtomatov.html>.