

Клеточно-автоматное моделирование физико-химических процессов нано уровня на графических ускорителях*

К.В. Калгин

Институт вычислительной математики и математической геофизики СО РАН

Моделирование каталитических физико-химических процессов на нано уровне кинетическим методом Монте-Карло требует достаточно больших вычислительных ресурсов. Создание “точных” параллельных алгоритмов моделирования таких процессов на графических ускорителях проблематично. В работе исследуются возможности применения блочно-синхронных режимов теории клеточных автоматов для моделирования кинетическим методом Монте-Карло. Предлагаются новые блочно-синхронные режимы, реализуемые на графическом ускорителе с достаточно высокой эффективностью, но имеющие большую точность воспроизведения динамики оригинальных моделей.

Введение

Большое число уже существующих моделей физико-химических процессов нано уровня, в основе которых лежит кинетический метод Монте-Карло (см., например, [1, 2]), в которых обычно используются регулярные решетки и локальные правила изменения характеристик частиц материала, представляются в форме асинхронного клеточного автомата.

Асинхронное клеточно-автоматное (АКА) моделирование реальных процессов требует больших вычислительных мощностей, поскольку для выявления каких-либо свойств моделируемых процессов необходимо оперировать большими количествами частиц (10^{10} - 10^{12}) в течение длительного времени (10^3 - 10^5 итераций). Естественно возникает потребность использовать параллельные вычислители для ускорения вычислений, к этому также располагают основные свойства клеточного автомата – естественная мелкозернистая параллельность и локальность взаимодействий при функционировании. Однако, асинхронные клеточные автоматы не поддаются столь же легкому распараллеливанию, как и обычные, синхронные, клеточные автоматы.

Это связано с тем, что при распараллеливании необходимо соблюдать следующие свойства: корректность и эффективность распараллеливания, а также равноправие в выборе клеток. Первые два свойства типичны для большинства параллельных программ. Последнее свойство возникает из определения асинхронного режима работы клеточного автомата и заключается в следующем: вероятность выбора некоторой клетки на некотором шаге должна быть равна вероятности выбора любой другой клетки на любом другом шаге.

Существующие параллельные алгоритмы можно разделить на два класса: не нарушающие свойства равноправия в выборе клеток [2, 3] (сохраняющие асинхронизм), и нарушающие его [4, 5] (нарушающие асинхронизм). Алгоритмы первого класса можно назвать “точными”, они пригодны для исполнения только на мультипроцессорах и мультикомпьютерах, но не на графических ускорителях.

В работах [4, 5, 6] на трёх моделях показано, что применение алгоритмов, нарушающих асинхронизм, не вносит существенных изменений в моделируемый процесс. В основе этих алгоритмов лежат блочно-синхронные режимы работы клеточного автомата. В данной работе на примере модели ZGB (Ziff-Gulari-Barshad) [7] физико-химического процесса

*Исследование выполнено по Программе фундаментальных исследований Президиума РАН, проект 15-9; поддержано Междисциплинарным Интеграционным проектом СО РАН 47 и РФФИ в рамках научных проектов 11-01-00567а и 12-01-31455;

окисления углекислого газа на каталитической поверхности показывается, что использование блочно-синхронных режимов может существенно изменить моделируемый процесс. Вопрос о том, для каких моделей использование алгоритмов, нарушающих асинхронизм, допустимо, то есть не вносит изменения в моделируемый процесс, остаётся открытым.

Далее в разделе 1 даётся формальное определение клеточного автомата; в разделе 2 даются определения различных режимов, вводится новый расширенный блочно-синхронный режим; в разделе 3 описывается классическая модель ZGB [7] протекания химических реакций на поверхности катализатора; в разделе 4 исследуется статистическое отклонение эволюции при блочно-синхронных режимах от эволюции при асинхронном режиме для модели ZGB, показывается преимущество новых расширенных блочно-синхронных режимов; в разделе 5 описывается реализация и полученное ускорение на современных графических ускорителях.

1. Определение клеточного автомата

Асинхронная вероятностная КА модель описывается четвёркой

$$КА = \langle X, A, \Theta, \Upsilon \rangle,$$

X – множество координат клеток. Наиболее употребимый вариант множества X – прямоугольная решётка

$$X = \{(i, j) \mid i \in \{0, 1, 2, \dots, N_I - 1\}, j \in \{0, 1, 2, \dots, N_J - 1\}\}. \quad (1)$$

A – алфавит, множество состояний клеток. Клетка x есть пара $x = (\mathbf{x}, a)$, где $\mathbf{x} \in X$ называется координатой клетки, и $a \in A$ – её состоянием. Клеточный массив Ω есть множество клеток, $\Omega = \{(\mathbf{x}, a)\} \subset X \times A$, где ни одна пара клеток не имеет одинаковой координаты и $\{\mathbf{x} \mid (\mathbf{x}, a) \in \Omega\} = X$.

Поскольку между множеством координат клеток и множеством клеток в клеточном массиве существует взаимно однозначное соответствие, далее мы будем отождествлять клетку и её координаты.

Θ – локальный оператор перехода, далее просто *оператор*. Оператор Θ есть вероятностное отображение вида

$$\Theta : A^{|T|} \times A^{|T|} \xrightarrow{p} A^{|T|}, \quad (2)$$

где *шаблон* T есть список *именующих функций* $\phi : X \rightarrow X$, $T = \{\phi_1, \phi_2, \dots, \phi_{|T|}\}$. Шаблон T определяет соседство клетки \mathbf{x} : $T(\mathbf{x}) = \{\phi_1(\mathbf{x}), \dots, \phi_{|T|}(\mathbf{x})\}$. В КА рассматриваемого класса следующие шаблоны наиболее используемы:

$$T_{13}(\mathbf{x}) = \{\mathbf{x} + \mathbf{v}_0, \mathbf{x} + \mathbf{v}_1, \dots, \mathbf{x} + \mathbf{v}_{12}\}, \quad (3)$$

$$T_9(\mathbf{x}) = \{\mathbf{x} + \mathbf{v}_0, \mathbf{x} + \mathbf{v}_1, \dots, \mathbf{x} + \mathbf{v}_8\}, \quad (4)$$

где $V = \{\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_{12}\} = \{(0, 0), (0, 1), (1, 0), (0, -1), (-1, 0), (1, 1), (1, -1), (-1, 1), (-1, -1), (0, 2), (2, 0), (0, -2), (-2, 0)\}$.

Применением оператора Θ к клетке \mathbf{x} называется обновление состояний клеток соседства $T(\mathbf{x})$ состояниями $\Theta(H(\mathbf{x}))$, где $H(\mathbf{x})$ есть список состояний клеток соседства $T(\mathbf{x})$.

Υ – режим работы, который определяет порядок применения оператора к клеткам:

$$\Upsilon \in \{\sigma, \alpha, \beta, \gamma\},$$

где σ означает синхронный режим, α – асинхронный, β – блочно-синхронный, и γ – новый расширенный блочно-синхронный режим. Первые КА модели [8] формулировались в синхронном виде, то есть использовался синхронный режим. Основным режим

работы рассматриваемого в данной работе класса КА моделей – асинхронный. Блочный синхронный режим используется в качестве вспомогательных в параллельных реализациях, в том числе на графических ускорителях. Далее подробно определяются режимы работы и эволюция клеточного автомата.

2. Эволюция клеточного автомата

В ходе моделирования происходит изменение клеточного массива Ω . Процесс моделирования разбивается на итерации. *Эволюция клеточного автомата* есть последовательность $\Omega^* = \{\Omega_0, \Omega_1, \Omega_2, \dots\}$, где Ω_0 – начальный клеточный массив, а Ω_t – клеточный массив после исполнения t -ой итерации.

Режим работы Υ задаёт порядок и количество применений оператора Θ к клеткам. Далее рассмотрим три режима наиболее часто используемых в КА моделях физико-химических процессов: синхронный (σ), асинхронный (α) и блочно-синхронный (β и γ).

Определение. В синхронном режиме σ итерация есть синхронное, одновременное применение оператора Θ ко всем клеткам клеточного массива.

Определение. В асинхронном режиме α итерация состоит из $|X|$ последовательных шагов, на каждом шаге оператор Θ применяется к случайно выбранной клетке. На каждом шаге клетка выбирается независимо от номера итерации, номера шага, клеточного массива и предыдущих выбранных клеток. Разбиение процесса моделирования в асинхронном режиме на итерации, которые состоят из $|X|$ шагов, является условным, и вводится по аналогии с синхронным режимом.

Определяемый далее блочно-синхронный режим содержит как синхронную компоненту, что позволяет эффективно исполнять его на параллельных вычислителях, так и асинхронную, что позволяет в некоторой степени имитировать работу асинхронного режима. Именно поэтому в работах [4, 5] он вводится исключительно как вспомогательный, для эффективного и простого исполнения АКА моделей на параллельных вычислителях. В работах [4, 5, 6] на нескольких АКА моделях физико-химических процессов показано, что эволюции, полученные при асинхронном и блочно-синхронном режимах, не отличимы по анализируемым характеристикам. Полученные в этих статьях результаты не могут служить доказательством какой бы то ни было эквивалентности эволюции КА при асинхронном и блочно-синхронном режимах. Кроме того автором были найдены “контрпримеры” показывающие, что для некоторых КА моделей использование блочно-синхронного режима вместо асинхронного приводит к существенным изменениям в эволюции КА.

Определение. [4, 5] В блочно-синхронном режиме β итерация состоит из w стадий. На каждой стадии оператор Θ синхронно применяется ко всем клеткам случайно выбранного подмножества S_i , $S_i \subset X$, из семейства $S = \{S_1, S_2, \dots, S_n\}$, причем $\forall k |S_k| = \frac{|X|}{w}$. Для обеспечения свойств корректности [4] синхронного применения, равноправности выбора клеток и эффективного параллельного исполнения [10] на семейство S накладываются следующие условия:

$$\bigcup_i S_i = X, \quad (5)$$

$$\forall i \neq j S_i \cap S_j = \emptyset, \quad (6)$$

а также условия с участием дополнительного шаблона T , покрывающего шаблон T_Θ оператора Θ , $T_\Theta \subseteq T$:

$$\forall S_i \bigcup_{\mathbf{x} \in S_i} T(\mathbf{x}) = X, \quad (7)$$

$$\forall S_i \forall \mathbf{x}_1, \mathbf{x}_2 \in S_i : T(\mathbf{x}_1) \cap T(\mathbf{x}_2) = \emptyset. \quad (8)$$

Увеличивая размер шаблона T мы приближаемся к асинхронному режиму: при условии, что $\forall \mathbf{x} \in X T(\mathbf{x}) = X$, блочно-синхронный и асинхронный режимы эквивалентны, $КА_\beta = КА_\alpha$.

Определение. В расширенном блочно-синхронном режиме γ итерация определяется так же, как и в блочно-синхронном режиме за следующим исключением. Условия (5) и (6), которые говорят, что S есть *одно* из возможных разбиений X , заменяются на следующее: S есть объединение *всех* разбиений X , для которых выполняются условия (7) и (8).

Таким образом, расширенный блочно-синхронный режим обладает свойствами корректности синхронного применения, за которое отвечает условие (8), а также эффективного параллельного исполнения, поскольку множество координат клеток покрывается шаблоном T плотно (7). Кроме того, увеличивается размер семейства S , что делает расширенный блочно-синхронный режим более похожим на асинхронный, по сравнению с блочно-синхронным. Тем не менее, условие равноправия при выборе клеток, как и в блочно-синхронном режиме, нарушается, что может привести к статистическим смещениям в моделируемом процессе. Исследование вносимых статистических смещений исследуется далее.

Перечислим элементы семейств S^9 и S^{13} (см. также Рис. 1 и 2) построенных с помощью шаблонов T_9 и T_{13} , соответственно:

$$S_{i,r_0,\dots,r_{N_Y/3}}^{9,1} = \{(x, y) : y \equiv i \pmod{3} \ x \equiv r_{\lfloor y/3 \rfloor} \pmod{3}\}, \quad (9)$$

$$S_{i,r_0,\dots,r_{N_X/3}}^{9,2} = \{(x, y) : x \equiv i \pmod{3} \ y \equiv r_{\lfloor x/3 \rfloor} \pmod{3}\}, \quad (10)$$

$$S^9 = \left(\bigcup_{i,r_0,\dots,r_{N_X/3}} S_{i,r_0,\dots,r_{N_X/3}}^{9,1} \right) \cup \left(\bigcup_{i,r_0,\dots,r_{N_Y/3}} S_{i,r_0,\dots,r_{N_Y/3}}^{9,2} \right), \quad (11)$$

$$S_i^{13,1} = \{(x, y) : x + 5y \equiv i \pmod{13}\}, \quad (12)$$

$$S_i^{13,2} = \{(x, y) : x + 8y \equiv i \pmod{13}\}, \quad (13)$$

$$S^{13} = \bigcup_i (S_i^{13,1} \cup S_i^{13,2}). \quad (14)$$

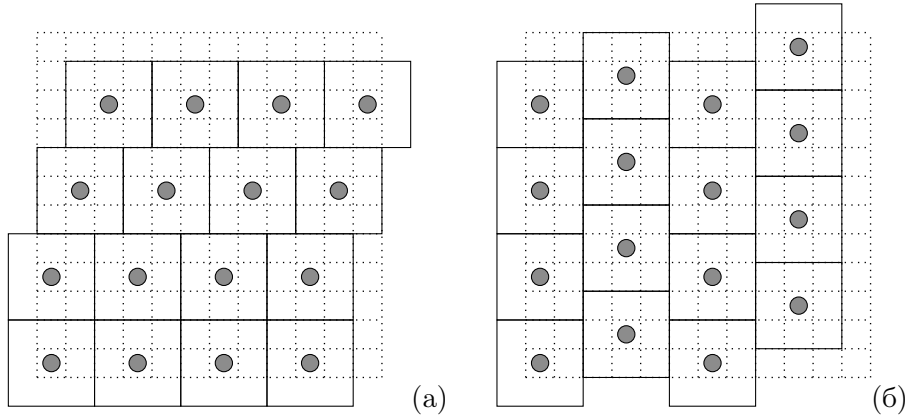


Рис. 1. Серыми кружками выделены клетки, координаты которых принадлежат множествам $S_{0,0,1,2,\dots}^{9,1}$ (а) и $S_{0,1,0,2,\dots}^{9,2}$ (б).

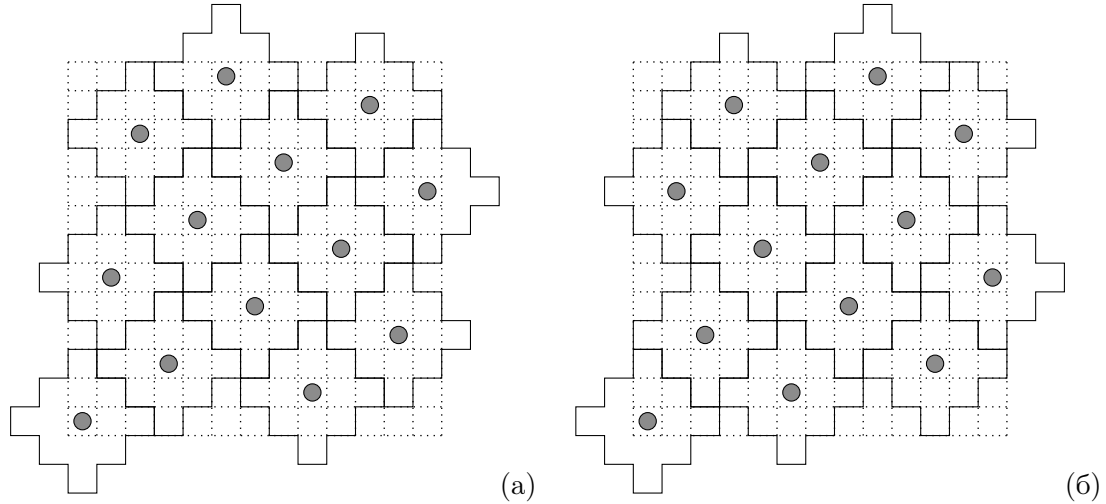


Рис. 2. Серыми кружками выделены клетки, координаты которых принадлежат множествам $S_0^{13,1}$ (а) и $S_0^{13,2}$ (б).

Таким образом, полностью определено семейство S расширенного блочно-синхронного КА для наиболее часто используемых шаблонов T_N , где $N \in \{9, 13\}$. Далее также используются блочно-синхронные режимы с семействами S , построенными с помощью шаблонов T_{25} (квадрат 5×5) и T_{49} (квадрат 7×7). Семейства S^{25} и S^{49} устроены аналогично семейству S^9 , построенного с помощью шаблона T_9 (квадрат 3×3).

3. Модель ZGB

Для проведения вычислительных экспериментов используется известная АКА модель ZGB (Ziff-Gulari-Barshad) [7] физико-химического процесса окисления углекислого газа на каталитической поверхности. Множество координат есть прямоугольная решётка (1). Алфавит $A = \{\emptyset, CO, O\}$, где состояние \emptyset означает пустой активный центр катализатора, а состояния CO и O – нахождение соответствующих молекул в активном центре катализатора.

Шаблон T_Θ оператора Θ равен $T_\Theta = T_{13}$. Во время применения оператора Θ в начале случайным образом выбирается элементарный процесс с вероятностью, соответствующей его скорости протекания: процесс адсорбции CO , адсорбции O_2 и реакции $CO + O = CO_2$. Затем выполняются действия по изменению состояний клеток из соседства T_{13} в соответствии с выбранным элементарным процессом. Эти действия подробно описаны в работе [7]. Стоит заметить лишь то, что большую часть кода реализующего оператор перехода составляют условные переходы, присваивания и целочисленные операции.

В данной работе в качестве дополнительного элементарного процесса вводится диффузия для определения её влияния на степень отличия эволюций при различных режимах работы. Диффузия вводится таким же способом, как это делалось в более поздних работах по моделированию поверхностных каталитических реакций методом Монте-Карло (см., например, [1]).

4. Различие в поведении асинхронного и блочно-синхронных режимов

В работах [4, 5, 6] на трёх физико-химических моделях показывается, что изменение режима с асинхронного α на блочно-синхронный β не вносит существенных изменений в общую картину эволюции КА. Как это будет показано далее, не для всех моделей изменение режима не изменяет эволюцию КА.

Для анализа степени подобия эволюций клеточных автоматов $КА_\alpha$, $КА_{\beta_N}$ и $КА_{\gamma_N}$ как стохастического объекта далее используется следующий подход: по эволюциям Ω_α^* , $\Omega_{\beta_N}^*$ и $\Omega_{\gamma_N}^*$ вычисляются характеристики $f(\Omega_\alpha^*)$, $f(\Omega_{\beta_N}^*)$ и $f(\Omega_{\gamma_N}^*)$, которые являются случайными величинами, после чего сравниваются их математические ожидания. Далее математическое ожидание характеристики для различных режимов будем обозначать через $m_\Upsilon(f)$, где $\Upsilon \in \{\alpha, \beta_N, \gamma_N\}$. Численную оценку величины m_Υ далее будем обозначать через m'_Υ . Для численной оценки m_Υ генерируется такое количество эволюций Ω_Υ^* , что $m_\Upsilon \in (m'_\Upsilon - 0.001, m'_\Upsilon + 0.001)$ с вероятностью больше 0.99.

Наиболее часто употребляема следующая характеристика

$$f_{t,a}(\Omega^*) = \frac{|\{(\mathbf{x}, a) \in \Omega_t\}|}{N_I \cdot N_J}, \quad (15)$$

то есть $f_{t,a}(\Omega^*)$ равно доле клеток с состоянием a в клеточном массиве после исполнения t итераций. С помощью этой характеристики зачастую очень трудно отличить эволюции с различными режимами.

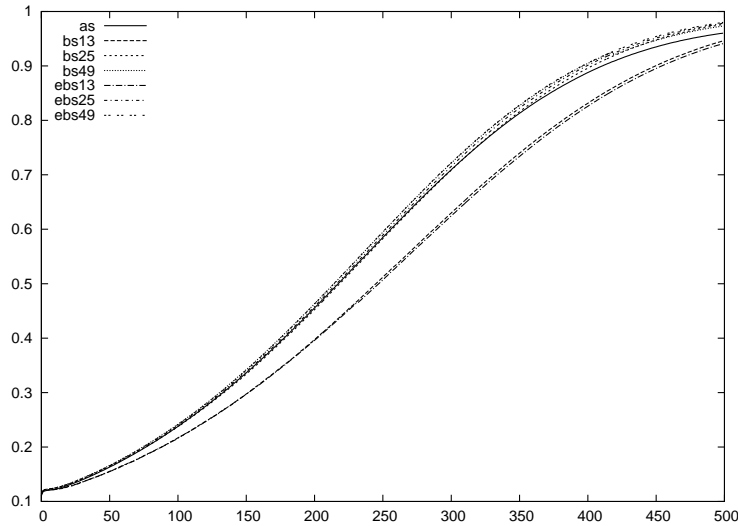


Рис. 3. Значение характеристики $f_{t,CO}$ для модели ZGB без диффузии для первых 500 итераций, $t < 500$. Графики as, bs13, bs25, bs49, ebs13, ebs25 и ebs49 соответствуют значению характеристики $f_{t,CO}$ при режимах $\alpha, \beta_{13}, \beta_{25}, \beta_{49}, \gamma_{13}, \gamma_{25}$ и γ_{49} .

На Рис. 3 представлено значение характеристики $f_{t,CO}$ для модели ZGB без диффузии. Видно, что часть графики, соответствующие режимам β_{13} и β_{25} , существенно отклоняются от графика, соответствующего асинхронному режиму. При введении в модель диффузии, отличить графики невозможно.

В данной работе предлагается использовать следующую более информативную характеристику эволюции клеточного автомата

$$f_{t,\Delta\mathbf{x}}(\Omega^*) = \frac{|\{(\mathbf{x}, a) \in \Omega_t : (\mathbf{x} + \Delta\mathbf{x}, a) \in \Omega_t\}|}{N_I \cdot N_J}, \quad (16)$$

то есть $f_{t,\Delta\mathbf{x}}(\Omega^*)$ равно доле пар клеток с координатами \mathbf{x} и $\mathbf{x} + \Delta\mathbf{x}$ в одинаковых состояниях в клеточном массиве после исполнения t итераций. Предлагаемая характеристика $f_{t,\Delta\mathbf{x}}$ содержит в себе информацию не только о количестве, но и о пространственном расположении состояний, в отличие от часто используемой характеристики $f_{t,a}$.

В Табл. 1 представлены результаты численной оценки математического ожидания среднеквадратичных отклонений

$$g_{t,\Upsilon} = \sqrt{\frac{1}{26 \cdot 51} \sum_{\Delta \mathbf{x}} (m_{\alpha}(f_{t,\Delta \mathbf{x}}) - m_{\Upsilon}(f_{t,\Delta \mathbf{x}}))^2} \quad (17)$$

при $k_{diff} \in \{0, 1\}$. Для численной оценки $g_{t,\Upsilon}$, обозначаемой как $g'_{t,\Upsilon}$, генерируется такое количество эволюций Ω_{α}^* и Ω_{Υ}^* , что $g_{t,\Upsilon} \in (g'_{t,\Upsilon} - 0.001, g'_{t,\Upsilon} + 0.001)$ с вероятностью больше 0.99.

	$\Upsilon = \beta_{13}$	$\Upsilon = \beta_{25}$	$\Upsilon = \beta_{49}$	$\Upsilon = \gamma_{13}$	$\Upsilon = \gamma_{25}$	$\Upsilon = \gamma_{49}$
$k_{diff} = 0$	0.0067	0.0039	0.0026	0.0054	0.001	0.0008
$k_{diff} = 1$	0.0028	0.0021	0.0015	0.0019	0.0006	0.0004

Таблица 1. Численные оценки математического ожидания среднеквадратичных отклонений $g_{t,\Upsilon}$ при $t = 100$ и варьирующихся Υ и коэффициенте k_{diff} . Значение $k_{diff} = 0$ соответствует оригинальной модели ZGB, значение $k_{diff} = 1$ – модели ZGB с введённой диффузией (скорость протекания процесса диффузии в этом случае равна сумме скоростей остальных элементарных процессов).

Перечислим основные результаты, которые можно наблюдать на представленном Рис. 3 и в Табл. 1:

1. Эволюции $\Omega_{\beta_{13}}^*$ и $\Omega_{\gamma_{13}}^*$ существенно отличаются от Ω_{α}^* по обеим характеристикам $f_{t,a}$ (15) и $f_{t,\Delta \mathbf{x}}$ (16) при $k_{diff} = 0$;
2. Эволюции Ω_{α}^* , $\Omega_{\beta_{25}}^*$, $\Omega_{\beta_{49}}^*$, $\Omega_{\gamma_{25}}^*$ и $\Omega_{\gamma_{49}}^*$ слабо различимы с точки зрения первой характеристики $f_{t,a}$ (15) при $k_{diff} = 0$;
3. Эволюции Ω_{α}^* , $\Omega_{\beta_{13}}^*$, $\Omega_{\beta_{25}}^*$, $\Omega_{\beta_{49}}^*$, $\Omega_{\gamma_{13}}^*$, $\Omega_{\gamma_{25}}^*$ и $\Omega_{\gamma_{49}}^*$ практически не различимы с точки зрения первой характеристики $f_{t,a}$ (15) при $k_{diff} = k_{adsCO} + k_{adsO_2}$;
4. Эволюции $\Omega_{\gamma_{25}}^*$ и $\Omega_{\gamma_{49}}^*$ практически неотличимы от эволюции Ω_{α}^* по всем представленным характеристикам, включая $g_{t,\Upsilon}$ (17), при всех значениях k_{diff} .

Перечисленные результаты, наблюдаемые автором и на многих других моделях, обобщаются в следующие два очень важных утверждения:

1. КА модели с диффузией существенно лучше поддаются моделированию блочно-синхронными режимами;
2. Предложенные автором расширенные блочно-синхронные режимы расширяют возможности эффективного параллельного исполнения КА моделей.

5. Реализация на CUDA

CUDA [9] расшифровывается как Compute Unified Device Architecture (унифицированная архитектура вычислительных устройств), описывает архитектуру аппаратной части и программного обеспечения. CUDA создана компанией Nvidia в 2007 году, позволяет разрабатывать программы общего назначения для графических ускорителей. Для тестирования использовались графические ускорители Nvidia GTX 280 и Nvidia GTX 680 (Kepler), их характеристики представлены в Таб. 2.

Стадии итерации блочно-синхронных режимов на графическом ускорителе выполняются последовательно, поскольку информационно зависимы между собой. Применение оператора Θ к клеткам выбранного на текущей стадии подмножества S_i может быть выполнено

	GTX 280	GTX 680
Тактовая частота (ГГц)	1.3	1.1
Мультипроцессоров (шт.)	30	8
Потоковых процессоров (шт.)	240	1536
Глобальная память (Мбайт)	768	4096
Разделяемая память (Кбайт)	16	48
Текстурный кэш (байт)	8192	8192
Кэш (Кбайт)	–	48
Регистры (шт.)	16384	32768
Планируемое число потоков на мультипроцессоре	1024	1536

Таблица 2. Параметры используемых графических ускорителей.

параллельно, согласно условию непересечения окрестностей различных клеток, входящих в S_i (8).

Параллельный алгоритм. Программа, исполняемая на центральном процессоре, используя генератор псевдо случайных чисел выбирает S_i из семейства S , после чего запускает функцию на графическом ускорителе, исполняющую одну стадию. Номер подмножества i передаётся на графический ускоритель через аргумент запускаемой функции. Число порождаемых потоков на графическом ускорителе равно числу клеток, к которым применяется оператор Θ на текущей стадии, $|S_i|$. По номеру подмножества i и своему номеру каждый поток в соответствии с формулами (9-14) вычисляет координаты клетки, к которой этот поток будет применять оператор Θ . После применения оператора к клетке поток завершает свою работу. После завершения всех порождённых потоков, с центрального процессора запускается следующая стадия.

Согласно рекомендациям разработчиков CUDA [9], для достижения высокой производительности необходимо порождать потоков в несколько раз больше, чем имеется ядер на графическом ускорителе. Данное условие будет выполняться на современных графических ускорителях для размеров клеточной области больше 700×700 . Именно для таких размеров и требуется распараллеливание.

В Таб. 3 представлены результаты тестирования реализации описанного алгоритма на современных графических ускорителях при шаблоне $T = T_{13}$. Использование других шаблонов T_{25} и T_{49} влияет на время исполнения незначительно.

	1000 × 1000	2000 × 2000	8000 × 8000
Ускорение GTX 280/Core i7	×25	×31	×35
Ускорение GTX 680/Core i7	×48	×60	×65

Таблица 3. Результаты тестирования при размерах клеточной области в 1000×1000 , 2000×2000 и 8000×8000 клеток на графических ускорителях GTX 280 и GTX 680. Показано ускорение по отношению ко времени работы последовательной программы на одном ядре процессора Core i7.

По результатам приведённым в Табл. 3 и работе [11] можно сказать, что один рядовой графический ускоритель GTX 280 на тестируемой задаче по производительности заменяет мощную 32-ядерную вычислительную систему над общей памятью (4 процессора Intel Xeon X7560 2.2Ghz, была доступна автору в рамках проекта Intel Manucore Testing Lab), а новый графический ускоритель GTX 680 заменит мощную 64-ядерную вычислительную систему (8 процессоров Intel Xeon X7560 2.2 Ghz, один узел в кластере Новосибирского Государственного Университета [12]).

Заключение

В работе показана принципиальная возможность использования графических ускорителей для моделирования физико-химических процессов на нано уровне кинетическими методами Монте-Карло. Предложены новые расширенные блочно-синхронные режимы, обеспечивающие существенно меньшее статистическое отклонение от “эталонного” асинхронного режима и имеющие ту же, что и блочно-синхронный режим, эффективность параллельного исполнения.

Работу следует продолжить в направлении (а) теоретического обоснования возможности использования расширенных блочно-синхронных режимов и (б) накопления статистики для других физико-химических моделей в основе которых лежит кинетический метод Монте-Карло.

Литература

1. Elokhin, V.I., Latkin, E.I., Matveev, A.V., and Gorodetskii, V.V. Application of Statistical Lattice Models to the Analysis of Oscillatory and Autowave Processes on the Reaction of Carbon Monoxide Oxidation over Platinum and Palladium Surfaces.
2. Overeinder B. J., Sloot P. M. A. Extensions to TimeWarp Parallel Simulation for Spatial Decomposed Applications // Proceedings of the Fourth United Kingdom Simulation Society Conference (UKSim 99) / Ed. by D. Al-Dabass, R. Cheng. Cambridge, UK: 1999. P. 67-73.
3. Lubachevsky B. Efficient parallel simulation of asynchronous cellular arrays // Complex Systems. 1987. Vol. 1. P. 1099-1123.
4. O.L. Bandman. Parallel Simulation of Asynchronous Cellular Automata Evolution. ACRI 2006, LNCS 4173, pp.41-47, 2006.
5. S. V. Nedeя, J. J. Lukkien, P. A. J. Hilbers, A. P. J. Jansen. Methods for Parallel Simulations of Surface Reactions. Proceedings of the 17th International Symposium on Parallel and Distributed Processing, 2003.
6. Anastasia Sharifulina, Vladimir Elokhin: Simulation of Heterogeneous Catalytic Reaction by Asynchronous Cellular Automata on Multicomputer. PaCT 2011: 204-209
7. Ziff R.M., Gulari E., Barshad Y. "Kinetic phase transitions in an irreversible surface-reaction model". Phys Rev Lett 56 (24): 2553-56, 1986.
8. von Neumann J. Theory of self reproducing automata. - University of Illinois Urbana. USA. 1966
9. NVIDIA CUDA Programming Guide. http://www.nvidia.com/object/cuda_get.html
10. K. V. Kalgin. Comparative Study of Parallel Algorithms for Asynchronous Cellular Automata Simulation on Different Computer Architectures. ACRI-2010, LNCS-6350, pp. 399-408, 2010.
11. К. В. Калгин. Параллельная реализация асинхронных клеточных автоматов на 32-ядерной вычислительной системе. Сиб. журн. вычисл. матем., 15:1 (2012), 55-65.
12. Вычислительный кластер Новосибирского Государственного Университета, <http://www.nusc.ru>.