

УДК 519.245, 519.688

## Дискретная стохастическая модель просачивания жидкости через пористое вещество: особенности параллельной реализации\*

О.Л. Бандман

Институт вычислительной математики и математической геофизики Сибирского отделения Российской академии наук,  
просп. Акад. М.А. Лаврентьева, 6, Новосибирск, 630090

E-mail: bandman@ssd.sccc.ru

**Бандман О.Л.** Дискретная стохастическая модель просачивания жидкости через пористое вещество: особенности параллельной реализации // Сиб. журн. вычисл. математики / РАН. Сиб. отд.-ние. — Новосибирск, 2018. — Т. 21, № 1. — С. 5–21.

Исследуются особенности параллельных реализаций дискретной стохастической модели, имитирующей просачивание жидкости через вещество (почву), имеющее сложную пористую микроструктуру. Моделирование должно показать процесс движения жидкости по извилинам пор и заполнение каверн и колодцев. Дискретная стохастическая модель этого процесса, предложенная ранее, представляет собой стохастический клеточный автомат (СКА), функционирование которого задается набором элементарных локальных операторов, действующих в дискретном клеточном пространстве и имитирующих перемещение (диффузия, конвекция, адсорбция) и преобразования (реакция, фазовое превращение) абстрактных или реальных частиц. Микроуровень представления процесса требует больших размеров клеточных пространств и, следовательно, вычислений на суперкомпьютерах. Главная проблема при этом состоит в том, что получение приемлемой эффективности параллельной реализации возможно только путем внесения детерминированности в алгоритм вычисления, т. е. снижения стохастичности модели. Несмотря на интенсивное изучение и применение стохастических моделей, методы параллельной реализации их на суперкомпьютерах изучены слабо. В статье этот пробел частично заполняется результатами серии вычислительных экспериментов, позволивших оценить достоинства и недостатки возможных способов реализации стохастической дискретной модели процесса просачивания жидкости в пористую среду со сложной морфологией на многопроцессорном кластере.

DOI: 10.15372/SJNM20180101

**Ключевые слова:** дискретное моделирование, стохастический клеточный автомат, стохастичность алгоритма, правила переходов, параллельные вычисления, блочно-синхронный режим, пористый материал, дискретная модель просачивания.

**Bandman O.L.** A discrete stochastic model of water permeation through a porous substance: parallel implementation peculiarities // Siberian J. Num. Math. / Sib. Branch of Russ. Acad. of Sci. — Novosibirsk, 2018. — Vol. 21, № 1. — P. 5–21.

The parallel implementation peculiarities of a discrete stochastic model that simulates water permeation through a porous substance (soil) with a complex morphology are studied. The simulation should reveal the fluid flowing along pore curves and filling wets and cavities. The discrete stochastic model of the process, proposed earlier, is a stochastic cellular automaton (SCA), whose functioning is represented by a set of elementary local operators acting in a cellular space and imitating displacement (diffusion, convection, adsorption) and transformations (reaction, phase transition) of abstract or real particles. The microlevel of the process representation requires the cellular space of a huge size, and hence, the computations should be implemented on supercomputers. With this, the main problem is in the fact that obtaining an acceptable parallelization efficiency is possible only by inserting some determinism into the computation algorithm, i.e., by decreasing the model stochasticity. Although stochastic models are under intensive investigation, the parallel implementation methods for them are poorly studied. This gap is partially covered by the results of computational

---

\*Работа выполнена при поддержке Программы Президиума РАН (грант № 15-6-2016).

experiments, given in this paper, which allow one to assess the advantages and drawbacks of methods for the discrete stochastic mode of water permeation through implementing a porous medium on a multicore cluster.

**Keywords:** *discrete simulation methods, stochastic cellular automata, stochasticity of the algorithm, transition rules, parallel computing, block-synchronous mode of functioning, porous material, permeation model.*

## 1. Введение

Математические модели пространственно распределенных естественных процессов, основанные на дифференциальных уравнениях в частных производных, предполагают, что пространственно-временные функции, описывающие моделируемый процесс, непрерывны и дифференцируемы. Однако в связи с возрастающим интересом в науке и промышленности к новым материалам и технологиям возникает необходимость моделировать явления существенно нелинейные и разрывные, например химические реакции, образование кристаллов, электроразряды в газах, рост и деление клеток живых организмов и др., которые составляют класс процессов типа “реакция–диффузия” (RD-процессов) [1–3]. Такие процессы могут быть диссипативными, содержать фазовые переходы, а также происходить в пространстве со сложной прерывистой геометрией. Для их моделирования предпринимаются попытки исследовать возможности событийно-стохастических методов имитационного моделирования. Такой подход применяется уже несколько десятилетий, причем в разных областях науки основанные на нем методы называются по-разному. Первым был метод моделирования химической реакции окисления моноокси углерода (ZGB-реакция [4]), который привел к развитию направления, называемого химиками “Кинетическим Методом Монте-Карло” (КМК-методом) [5–8]. В материаловедении аналогичный метод используется для моделирования разрушения материалов и называется “методом вероятностных клеточных автоматов” [9–11]. При моделировании биологических явлений самоорганизации [12] и формирования пространственных и поверхностных структур применяется чаще всего термин “асинхронный клеточный автомат” [12, 13] или “стохастический клеточный автомат” [14]. Последние названия наиболее точно соответствуют классификации математических моделей пространственной динамики, а также согласуются с современным способом представления моделей в виде правил (rule based models), который имеет свое воплощение в языке программирования “Wolfram Language” [15] и реализован в широко распространенной системе программирования *Mathematica* [16].

В дискретных стохастических методах математическое описание RD-процессов составляется из набора элементарных локальных операторов (правил перехода) над абстрактными или реальными частицами, наделенными способностью перемещаться (диффузия, конвекция) в дискретном пространстве и претерпевать разного рода превращения в дискретном времени (реакция, фазовый переход). Локальные операторы имеют вид вероятностных подстановок, отображающих локальные изменения состояний пространства моделирования при наличии определенных условий. Пространством моделирования является упорядоченное множество клеток. Клетка характеризуется пространственными координатами и состоянием. Состояниями клеток могут быть целые числа или символы, отображающие наличие в клетках определенного типа частиц в заданный момент времени. Применение подстановок к клеткам происходит стохастически: в каждый дискретный момент времени случайно выбирается подстановка (временная стохастичность) и применяется к случайно выбранной клетке (пространственная стохастичность или асинхронность) [13, 14].

Несмотря на большую востребованность и интенсивное распространение стохастических моделей, методы параллельной реализации на суперкомпьютерах разработаны для них слабо. А, между тем, моделирование реальных процессов имеет смысл при таких размерах клеточных пространств, которые могут быть обработаны только на современных вычислительных системах с параллельной архитектурой. Параллельная реализация СКА-моделей совсем не тривиальна. Проблемы связаны с тем, что пространственная стохастичность требует передачи каждого изменения состояния пограничной клетки немедленно соседнему процессору, что делает неэффективным применение метода декомпозиции пространства (domain decomposition method) при реализации модели на суперкомпьютерных кластерах. Первые попытки построить приемлемые алгоритмы параллельной реализации [17, 18] привели к громоздким и непрактичным реализациям. Предложенный позже метод перехода от асинхронного к блочно-синхронному режиму [19] оказался более эффективным и удобным для моделирования сложных стохастических процессов [14] и успешно использовался в ряде исследований [20, 21]. Однако метод носит эвристический характер и требует дополнительных обоснований, которые могут быть получены путем экспериментальных исследований следующих особенностей стохастической модели [13].

1. Достижение приемлемой эффективности параллельной реализации возможно только за счет введения синхронизации, т. е. за счет уменьшения стохастичности модели.
2. Введение синхронизации в вычислительный процесс должно быть согласовано во всех процессорах, участвующих в параллельной реализации.
3. Простота локальных операторов КА приводит к такому соотношению затрат времени на вычисления и межпроцессорные обмены, что делает глубокое распараллеливание неэффективным.

Учет этих особенностей накладывает некоторые ограничения на выбор параметров параллельного алгоритма реализации модели на суперкомпьютерах. В статье делается попытка количественно оценить указанные выше ограничения и подтвердить полученные оценки вычислительными экспериментами. В качестве репрезентативного примера взята стохастическая клеточно-автоматная модель конвекционно-реакционного процесса просачивания жидкости через пористый материал со сложной извилистой структурой пор.

Статья состоит из четырех пунктов. Следующий (второй) пункт содержит формальное представление стохастической клеточно-автоматной модели RD-процесса. Третий пункт посвящен СКА-модели просачивания жидкости в почву и проблемам ее параллельной реализации. В заключении даны анализ результатов и выводы.

## 2. Стохастическая дискретная модель реакционно-диффузионного процесса

### 2.1. Формальное представление стохастического клеточного автомата

Математическое представление дискретной стохастической модели пространственной динамики (далее модели) основано на формализме *алгоритма параллельных подстановок* [22], адаптированном в [23] к описанию КА, моделирующих естественные процессы. В соответствии с [22] модель задается тройкой понятий:

$$\aleph = \langle X, A, \Theta(X) \rangle,$$

где

$X = \{\mathbf{x}_k \mid k = 1, \dots, N\}$  — конечное множество *имен* клеток;

$A$  — *алфавит состояний* клеток — конечное множество символов;

$\Theta(X) : A \times X \rightarrow A \times X$  — *глобальный оператор перехода*.

Время предполагается равномерно дискретным  $t = 0, 1, \dots, t, \dots, T$ . В каждый момент  $t$  модель характеризуется *клеточным массивом*  $\Omega(t) \in A \times X$ ,  $\Omega(t) = \{(u_k, \mathbf{x}_k) : u_k \in A, \mathbf{x}_k \in X\}$ , который является упорядоченным (по именам) множеством пар вида  $(u_k, \mathbf{x}_k)$ , называемых *клетками*, а проекция клеточного массива  $\Omega_A(t) = (u_1, u_2, \dots, u_{|X|})$  на алфавит состояний называется *глобальным состоянием* КА. На множестве  $X$  определено *соседство клетки* с именем  $\mathbf{x}$ :

$$T(\mathbf{x}) = \{\mathbf{x}, \mathbf{x} + \mathbf{a}_1, \dots, \mathbf{x} + \mathbf{a}_{q-1}\}, \quad (1)$$

где  $\mathbf{a}_j$  — вектор смещения  $\mathbf{x}$  по координате  $j$ ,  $q$  — размер соседства. Подмножество клеток с именами из  $T(\mathbf{x})$  образуют *локальную конфигурацию*

$$S(\mathbf{x}) = \{(u_0, \mathbf{x}), (u_1, \mathbf{x} + \mathbf{a}_1), \dots, (u_{q-1}, \mathbf{x} + \mathbf{a}_{q-1})\} \quad (2)$$

с *определяющим соседством*  $T(\mathbf{x})$ .

Глобальный оператор  $\Theta(X)$  является композицией некоторого количества *локальных операторов*  $\theta_j(\mathbf{x})$ ,  $j = 1, \dots, n$ , из конечного множества  $\Theta = \{\theta_1(\mathbf{x}), \dots, \theta_n(\mathbf{x})\}$ . Локальный оператор  $\theta_j(\mathbf{x})$  имеет вид подстановки:

$$\theta_j(\mathbf{x}) : S_j(\mathbf{x}) \rightarrow S'_j(\mathbf{x}). \quad (3)$$

Применение  $\theta_j(\mathbf{x})$  к конкретной клетке с именем  $\mathbf{x}$  сводится к замене состояний  $u_i$  в клетках из  $S_j(\mathbf{x})$  (2) на новые значения  $u'_i$  в клетках из  $S'_j(\mathbf{x})$  (3), причем

$$u'_i = f_k(u_0, \dots, u_{q-1}), \quad q = |S_j(\mathbf{x})|, \quad k = 0, \dots, |S'_j(\mathbf{x})|, \quad |S'_j(\mathbf{x})| \leq |S_j(\mathbf{x})|, \quad (4)$$

где  $f_k(u_0, \dots, u_{q-1})$  — *функция перехода*.

Далее предполагается, что применение подстановки  $\theta_j(\mathbf{x})$  к одной клетке  $\mathbf{x} \in X$  происходит за один временной шаг, обозначаемый  $\tau$ , а выполнение  $\Theta(X)$  состоит из применения всех  $\theta_j \in \Theta$  ко всем клеткам  $\mathbf{x} \in X$ , что составляет глобальный переход  $\Omega(t+1) = \Theta(\Omega(t))$ , называемый *итерацией*. Таким образом, каждая итерация содержит  $|X| |\Theta| = Nn$  шагов, т.е. применений подстановок  $\theta_j(\mathbf{x})$ ,  $j = 1, \dots, |\Theta|$ ,  $\mathbf{x} \in |X|$ , а результатом моделирования является итерационный процесс

$$\Omega^* = \Omega(0), \dots, \Omega(t), \dots, \Omega(T_{\text{fin}}), \quad (5)$$

называемый *эволюцией*.

Вычисление глобального оператора  $\Theta(X)$  должно удовлетворять *условию корректности*, которое требует, чтобы при переходе из одного глобального состояния в другое не произошло “состязаний”. Иными словами, не должна одна и та же клетка одновременно подвергаться воздействию двух попыток изменения ее состояния. Формально это условие выражается в том, что

$$\forall \mathbf{x}, \mathbf{y} \in X \ \& \ \forall k, m \in \{1, \dots, n\} : T'_k(\mathbf{x}) \cap T'_m(\mathbf{y}) = \emptyset, \quad n = |\Theta(\mathbf{x})|. \quad (6)$$

При этом возможно, что  $m = k$ .

Эволюция существенно зависит от способа (стохастический или детерминированный) выбора клеток и подстановок в вычислении  $\Theta(X)$  на каждой итерации. Количественной характеристикой, в значительной степени влияющей на поведение модели, является ее

*стохастичность* ( $\lambda$ ), которая определяется как доля вероятностных элементарных операций в вычислении глобального оператора [26]:

$$\lambda = E_{\text{rand}}/E, \quad (7)$$

где  $E$  и  $E_{\text{rand}}$  — общее количество элементарных операций и количество вероятностных элементарных операций в вычислении  $\Theta(X)$  соответственно. Операция считается вероятностной, если ее применение определяется случайным числом.

Способ выбора имен клеток  $\mathbf{x} \in X$  определяет *пространственную стохастичность*, тогда как способ выбора подстановок — *операционную стохастичность*. Совместно они определяют *стохастичность модели*.

## 2.2. Режимы функционирования

Существует три следующих базовых типа пространственной стохастичности СКА. В [27] они определяются *режимами* функционирования СКА и обозначаются индексами из множества  $\rho = \{\sigma, \alpha, \beta\}$ .

*Синхронный режим*,  $\rho = \sigma$ , предполагает следующий алгоритм применения  $\theta(\mathbf{x})$  ко всем  $\mathbf{x} \in X$  на каждой итерации:

1.  $\theta(\mathbf{x})$  (3) применяется ко всем  $\mathbf{x} \in X$  (выбираемым в любом порядке), при этом новые состояния клеток записываются во вспомогательный массив  $\Omega'(t)$ ;
2.  $\Omega'(t)$  является исходным для следующей итерации, т. е.  $\Omega(t+1) = \Omega'(t)$ .

В синхронном режиме функционируют классические КА [28, 29]. Их особенностью является тот факт, что  $|T'(\mathbf{x})| = 1$ , т. е. применение подстановки меняет состояние только одной клетки, и, следовательно, условие корректности (6) удовлетворяется автоматически. Это ограничивает возможности синтеза синхронных КА-моделей, зато упрощает их параллельные реализации, позволяя выполнять межпроцессорные обмены единожды за итерацию. Поскольку все пространственные операции детерминированные,  $E_{\text{rand}}^\sigma = 0$ .

*Асинхронный режим*,  $\rho = \alpha$ , предписывает следующий порядок вычислений глобального оператора:

1. Выбирается клетка  $\mathbf{x} \in X$  с вероятностью  $p = |X|^{-1}$ ;
2.  $\theta(\mathbf{x})$  применяется к выбранной клетке, состояния клеток из  $S'(\mathbf{x})$  немедленно заменяются на соответствующие новые, полученные по (4);
3. Итерация завершается за  $|X|$  повторений пунктов 1 и 2 для каждой  $\theta \in \Theta$ .

При асинхронном режиме условие (6) не накладывает ограничений на количество одновременно обновляемых клеток, что расширяет возможности локальных операторов, но затрудняет реализацию вычислений на параллельно работающих процессорах. Для асинхронных режимов итерацией принято условно считать выполнение  $|X|$  шагов так же, как это принято в кинетическом методе Монте-Карло [7, 8]. На самом деле никакого разграничения вычислительного процесса между итерациями не существует. При асинхронном режиме  $\forall \theta \in \Theta : E_{\text{rand}}^\alpha = N$ .

*Упорядоченный асинхронный режим*,  $\rho = \omega$ , является вырожденным случаем асинхронного режима, когда клетки  $\mathbf{x} \in X$  выбираются в заданном (детерминированном) порядке, каждый раз обновление состояний клеток из  $T'(\mathbf{x})$  происходит немедленно. Все операции детерминированные,  $E_{\text{rand}}^\omega = 0$ .

*Блочно-синхронный режим*,  $\rho = \beta$ , является промежуточным между синхронным и асинхронным режимами [19]. В нем допускаются подстановки, изменяющие более одной клетки одновременно. Однако для соблюдения условия (6) одновременные (синхронные) обновления состояний выполняются на подмножествах  $\Pi_l \in X$ , в которых клетки достаточно далеко отстоят друг от друга. При декартовой структуре трехмерного ( $D = 3$ ) клеточного массива эти подмножества определяются следующим образом.

1. Выбирается компактный блок клеток, называемый  $\beta$ -блоком,  $B_\beta(i, j, k) = \{(i + a_i, j + a_j, k + a_k) : a_i, a_j, a_k = -R_\beta, 1, \dots, 0, \dots, R_\beta\}$ , где  $R_\beta \ll I$  — радиус блока. Размер блока должен удовлетворять условию

$$B_\beta(i, j, k) \supseteq T''(i, j, k), \quad T''(i, j, k) = \bigcup_{l=1}^n T'_l(i, j, k), \quad n = |\Theta|, \quad (8)$$

$T''(i, j, k)$  — обобщенное определяющее соседство СКА, а  $T'_l$  — определяющее соседство локальной конфигурации  $S'_l(i, j, k)$  (2) в подстановке  $\theta_l(i, j, k) \in \Theta$ ,  $q' = |S'_l(i, j, k)|$ . Линейный размер и количество клеток  $\beta$ -блока равны

$$b = 2R_\beta + 1, \quad m = b^D$$

соответственно, где  $D$  — размерность пространства моделирования.

2. Линейные размеры  $I, J, K$  клеточного массива  $X$  корректируются так, чтобы  $I_b = I/b$ ,  $J_b = J/b$ ,  $K_b = K/b$  были целыми числами, после чего выделяется подмножество  $X_0 \subset X$ :

$$X_0 = \{(i, j, k) \in X : \\ i = 0, b, 2b, \dots, I-1; j = 0, b, 2b, \dots, J-1; k = 0, b, 2b, \dots, K-1\}. \quad (9)$$

3. Строится порожаемое  $\beta$ -блоком разбиение  $\Pi = \{\Pi_1, \dots, \Pi_m\}$  множества  $X$  такое, что

$$\Pi_l \cap \Pi_h = \emptyset, \quad \bigcup_{l=1}^m \Pi_l = X \quad \forall h, l \in \{1, \dots, m\}. \quad (10)$$

Каждое подмножество  $\Pi_l \in \Pi$  содержит клетки, полученные сдвигом координат клеток  $(i, j, k) \in X_0$  на значения соответствующих координат клетки  $(a_i, a_j, a_k)_l \in B_\beta(R_\beta, R_\beta, R_\beta)$  (8):

$$\Pi_l = \{(i + a_i, j + a_j, k + a_k) : \\ \forall (i, j, k) \in X_0, \forall (a_i, a_j, a_k)_l \in B(R_\beta, R_\beta, R_\beta)\}, \quad l = 0, \dots, m. \quad (11)$$

4. Итерация состоит из  $m$  выбираемых в случайном порядке *стадий*, на каждой  $l$ -й стадии все  $\theta \in \Theta$  применяются ко всем  $(i, j, k) \in \Pi_l$  синхронно. Для обеспечения корректности (6) при параллельной реализации этот случайный порядок выбора стадий должен быть один и тот же во всех процессорах. Количество вероятностных операций для каждой  $\theta \in \Theta$ :  $E_{\text{rand}}^\beta = m$ .

*Операционная стохастичность* определяет порядок выбора подстановок из  $\{\theta_1, \dots, \theta_n\}$  при выполнении глобального оператора. Наиболее популярными являются

два вида операционной стохастичности (хотя известны и промежуточные [21]), обозначаемые в [13] индексами из  $\mu = \{\delta, \gamma\}$ .

*Детерминированный выбор*,  $\mu = \delta$ , когда к каждой клетке  $\mathbf{x} \in X$  применяются все подстановки  $\theta_i \in \Theta$  в заданном порядке,  $E_{\text{rand}}^\delta = n$ .

*Случайный выбор*,  $\mu = \gamma$ , когда к каждой  $\mathbf{x} \in X$  применяется одна случайно выбираемая подстановка  $\theta_i \in \Theta$  с вероятностью

$$p_i = \frac{k_i}{\sum_{i=1}^n k_i}, \quad (12)$$

где  $k_i$ ,  $i = 1, \dots, n$ , — скорости выполнения событий, соответствующих подстановкам  $\theta_1, \dots, \theta_n$ ,  $E_{\text{rand}}^\gamma = n$ .

Для стохастических моделей далее используется расширенное понятие режима функционирования модели, определяемое парой характеристик из  $\rho \times \mu$ . СКА-модель с глобальным оператором  $\Theta_{\rho,\mu}$  обозначается  $\aleph_{\rho,\mu}$ . Стохастические характеристики для всех случаев сведены в таблице. В сложных СКА-моделях, где применяются методы композиции глобальных операторов [25], каждый из них может работать в своем режиме.

**Таблица.** Стохастические характеристики глобальных операторов  $\Theta_{\rho,\mu}$

Характеристики	$\rho, \mu$							
	$\alpha, \gamma$	$\alpha, \delta$	$\beta, \gamma$	$\beta, \delta$	$\omega, \gamma$	$\omega, \delta$	$\sigma, \gamma$	$\sigma, \delta$
$E_{\text{rand}}$	$Nn$	$N$	$mn$	$m$	$n$	0	$n$	0
$\lambda$	1	$1/n$	$m/N$	$m/Nn$	$1/N$	0	$1/N$	0

В [14] стохастическим назван КА с вероятностным выбором как пространственного, так и операционного режимов, т. е.  $(\alpha, \gamma)$ -режим. Однако разнообразие свойств моделируемых процессов, а также необходимость эффективной параллельной реализации требуют расширения понятия стохастической модели путем включения в него всех режимов из таблицы, включая вырожденный синхронный  $(\sigma, \delta)$ -режим.

### 3. Моделирование процесса просачивания

#### 3.1. Стохастическая модель просачивания

Клеточно-автоматная модель просачивания жидкости в почву была предложена в [30], где подробно описаны ее моделирующие возможности. Здесь приводится упрощенная версия, чтобы сфокусировать внимание на проблемах параллельной реализации.

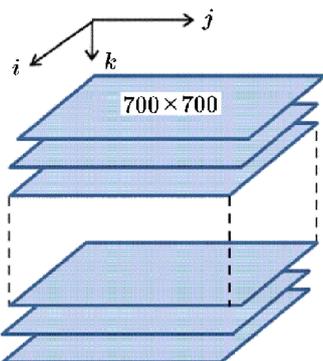
Образец почвы, сквозь которую просачивается жидкость, задан оцифрованной томограммой, представленной набором из 1480 файлов, каждый из которых имитирует один горизонтальный слой размером  $700 \times 700$  байт трехмерного булева массива  $X = \{(i, j, k) : i, j = 0, \dots, 699; k = 0, \dots, 1479\}$  (рисунок 1), что соответствует образцу почвы размером  $2 \times 2 \times 15$  мм<sup>3</sup> (рис. 2). Если двоичное значение байта  $u(i, j, k) = 0$ , то клетка  $(i, j, k)$  имитирует пустое пространство поры, если  $u(i, j, k) = 1$ , то клетка  $(i, j, k)$  имитирует твердую породу.

СКА  $\aleph = \langle A, X, \Theta(X) \rangle$ , моделирующий просачивание жидкости, имеет символичный алфавит состояний  $A = \{s, w, a\}$ , где  $s$  соответствует твердой породе,  $w$  — воде,  $a$  — воздуху в пространстве поры. Дискретное пространство  $X = \{(i, j, k) : i = 0, \dots, I - 1; j = 0, \dots, J - 1; k = 0, \dots, K - 1\}$ , где  $I = J = 700$ ,  $K = 1000$  (высота массива уменьшена,

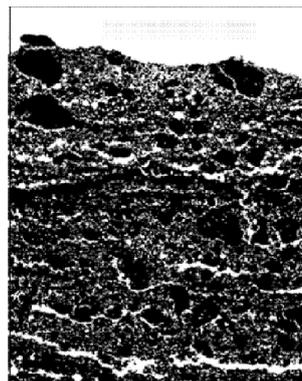
поскольку предварительные вычисления показали, что жидкость ниже не просачивается). Глобальный оператор

$$\Theta(X) = \Theta_C(\Theta_D(X)), \quad (13)$$

где  $\Theta_D(X)$  моделирует диффузию,  $\Theta_C(X)$  — конвекцию.



**Рис. 1.** Структура файлового представления образца почвы



**Рис. 2.** Образец почвы в разрезе при  $j = 400$

Оператор диффузии  $\Theta_D(X)$  имитирует разлив жидкости, т. е. выравнивание свободной поверхности жидкости в кавернах (порах), а также на верхней поверхности жидкости, используя подстановку [30]:

$$\theta_D(i, j, k) : \{(w, (i, j, k)), (a, \phi_l(i, j, k - 1))\} \xrightarrow{p_D} \{(a, (i, j, k)), \phi_l(w, (i, j, k - 1))\}, \quad (14)$$

$$l = 1, \dots, c.$$

Подстановка  $\theta_D(i, j, k)$  выполняет обмен состояниями между клеткой  $(i, j, k)$  и  $l$ -м соседом клетки  $l(i, j, k - 1)$ ,  $l = 1, \dots, c$ , с именем  $\phi_l(i, j, k - 1)$ , у которого  $u_l \neq 0$  (не твердая порода),  $c \leq 4$ . Вероятность  $p_D = 1/c$ ,  $\theta_D(i, j, k)$  применяется последовательно ко всем  $k$ -м плоскостям от  $k = 0$  до  $k = K - 1$  в упорядоченном асинхронном режиме. В каждой  $k$ -й плоскости  $\theta_D(i, j, k)$  применяется ко всем  $(i, j, k)$  синхронно (рис. 3).

Поскольку диффузия — процесс более сложный и медленный, чем конвекция, то для согласования их скоростей глобальный оператор диффузии  $\Theta_D(X)$  должен содержать  $n$  циклов применения  $\theta_D(i, j, k)$  ко всем  $(i, j, k) \in X$  в течение одной итерации, т. е.

$$\Theta_D(X) = (\theta_D(X))^n. \quad (15)$$

Величина  $n$  зависит от свойств вещества и конкретных условий. Ее можно подобрать в процессе тестового моделирования, определив путем визуального наблюдения на мониторе или включив в программу моделирования проверку, при каком значении  $n$  свободная поверхность жидкости остается гладкой (рис. 4). Для увеличения производительности и эффективности параллельной реализации оператор диффузии выполняется в блочно-синхронном режиме  $(\beta, \gamma)$ .

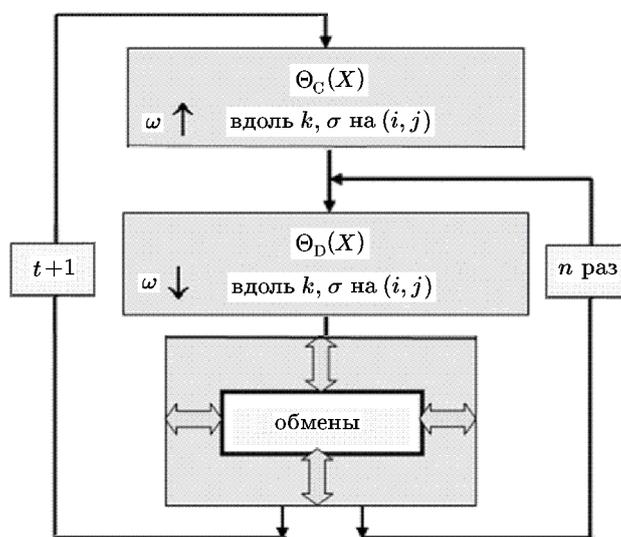


Рис. 3. Схема алгоритма процесса просачивания

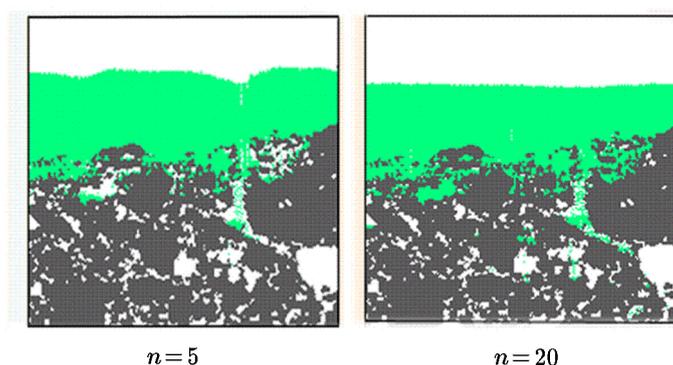


Рис. 4. Определение количества циклов диффузии на каждой итерации, обеспечивающей гладкий уровень поверхности жидкости

Конвекционный оператор  $\Theta_C(X)$  имитирует движение абстрактных частиц вдоль направления силы гравитации. На каждом шаге подстановка  $\theta_C(i, j, k)$  применяется в асинхронном упорядоченном режиме ко всем слоям массива, начиная от  $k = K - 1$  до  $k = 0$ , в каждом  $k$ -м слое синхронно ко всем клеткам  $(i, j, k)$ :

$$\theta_C(i, j, k) : \{(a, (i, j, k)), (w, (i, j, k - 1))\} \xrightarrow{p_C} \{(w, (i, j, k)), (a, (i, j, k - 1))\}. \quad (16)$$

Вероятность  $p_C$  принимается равной или близкой к единице, так как это самое быстрое действие.

Основной особенностью предлагаемой трехмерной КА-модели просачивания является то, что его конвекционная и диффузионная составляющие разделены в пространстве таким образом, что конвекционная составляющая в нем — одномерная (вдоль вертикальной оси  $k$ ), а диффузионная — двумерная (в горизонтальных плоскостях  $k = \text{const}$ ).

### 3.2. Особенности параллельной реализации стохастической модели просачивания

Для реализации СКА-модели просачивания на суперкомпьютере используется широко распространенный в вычислительной математике метод декомпозиции области вычислений (domain decomposition method), который состоит в разделении клеточного массива на  $P$  частей,  $\Omega = \Omega_0 \cup \dots \cup \Omega_{P-1}$ , далее называемых *доменами*, и размещении их на  $P$  параллельно работающих процессорах. Разделение на домены производилось вдоль вертикальной оси  $k$  двояким образом: а) на пластины (одномерная декомпозиция) и б) на колонны (двумерная декомпозиция) (рис. 5).

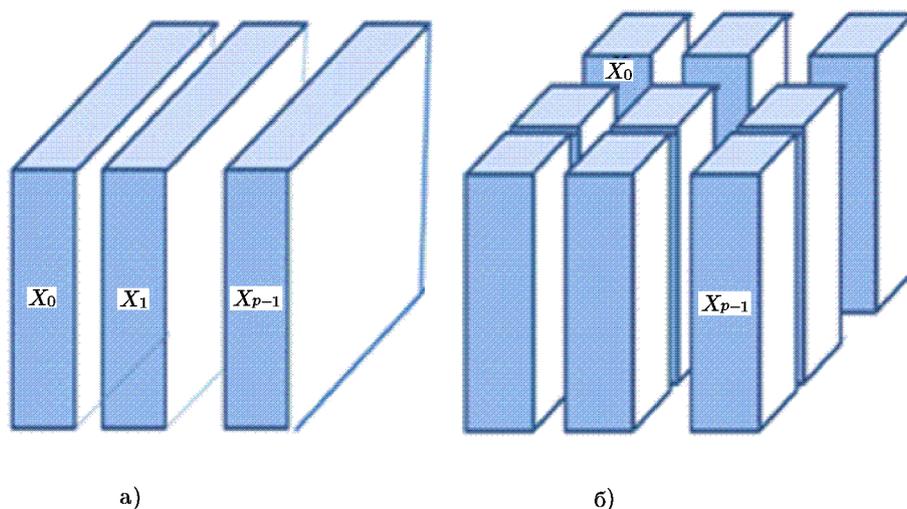


Рис. 5. Два способа разделения клеточного массива на домены

При такой декомпозиции обмены пограничными данными требуются только при исполнении оператора диффузии, т. е. между гранями, параллельными оси  $k$  (рис. 5).

Для того, чтобы обеспечить межпроцессорный обмен, домены дополняются периферийными “теневыми” слоями, содержащими  $V$  клеток, где

$$V_{\text{dom}} = 2R_{\beta}K_{\text{dom}}(I_{\text{dom}} + J_{\text{dom}}), \quad (17)$$

$R_{\beta}$  — радиус обобщенного соседства  $T''$  (8),  $I_{\text{dom}}$ ,  $J_{\text{dom}}$ ,  $K_{\text{dom}}$  — линейные размеры домена.

Задача эффективной параллельной реализации состоит в построении вычисления, оптимального с точки зрения минимизации затрат времени и аппаратуры. В большинстве исследований по параллельным вычислениям в качестве количественной оценки параллельной реализации используют *эффективность распараллеливания*

$$\eta = \frac{\text{Time}_1}{\text{Time}_P} = \frac{\text{Time}_1}{\text{Time}_1 + l_{\nu} \text{Time}_{\text{ex}}}, \quad (18)$$

где  $\text{Time}_1$  — время вычисления одной итерации на одном домене,  $\text{Time}_P$  — время вычисления одной итерации параллельно на  $P$  доменах,  $\text{Time}_{\text{ex}}$  — время, необходимое для выполнения одного обмена данными между  $P$  процессами, а  $l_{\nu}$  — количество обменов данными за одну итерацию.

Из (18) следует, что эффективность связана с параметрами модели посредством величины  $l_\nu$ , которая для режима  $(\beta, \gamma)$  равна (см. таблицу):

$$l_{\beta, \gamma} = |B_\beta| |\Theta| = mn = \lambda N n. \quad (19)$$

Для рассматриваемой задачи при минимальном размере  $\beta$ -блока  $m = 9$  и 20-ти кратном ( $n = 20$ ) повторении  $\Theta_D$  количество обменов на каждой итерации  $l_{\beta, \alpha} = 180$ , при  $m = 25$  эта величина становится  $l_{\beta, \alpha} = 500$ .

Подставив (17) и (19) в (18), получим следующее соотношение между эффективностью и стохастичностью:

$$\eta_{\beta, \gamma} = \frac{1}{1 + (\lambda/N) \sqrt[P]{P/N} (\tau_{\text{ex}}/n\tau_{\text{op}})}, \quad (20)$$

где  $\tau_{\text{ex}}$  и  $\tau_{\text{op}}$  — время межпроцессорного обмена байтом и время выполнения локального оператора соответственно.

Из (20) непосредственно следуют три вывода.

- Эффективность параллельной реализации тем больше, чем меньше стохастичность СКА-модели, которая зависит от выбора размера блока при блочно-синхронном режиме.
- Эффективность слабо уменьшается с увеличением числа параллельных процессов  $P$ .
- Чем сложнее вычисление локальных операторов (меньше отношение  $\tau_{\text{ex}}/n\tau_{\text{op}}$ ), тем больше эффективность.

Первый вывод приводит к необходимости выбирать минимальное значение размера  $\beta$ -блока  $m$  при  $\beta$ -режимах. Вопрос, насколько значение  $m$  влияет на “качество” моделирования, т. е. точность отображения моделируемого явления, пока открыт. Известны только некоторые результаты стохастического моделирования на суперкомпьютерах [19, 20, 31], которые не выявили различий между результатами моделирования при асинхронном и блочно-синхронном режимах.

Второй вывод стимулирует выбор доменов больших размеров и с одинаковыми размерами “теневого” слоев (домены в виде кубов и квадратов, при которых отношение  $V/P$  при равных  $N$  меньше).

И, наконец, третий вывод призывает отдавать предпочтение кластерам с быстрыми межпроцессорными коммуникациями.

Кроме того, требование корректности (6) коллективных коммуникаций (MPI-Sendrecv [32]) накладывает дополнительные ограничения на организацию межпроцессных обменов. Это значит, что  $\beta$  — стадии  $\Pi_l \in \Pi$ , хотя и выбираются случайно, в любой момент должна быть одной той же во всех доменах. Для этого на каждой итерации необходимо в одном из доменов сформировать случайную последовательность из  $m$  натуральных чисел и транслировать ее всем процессорам.

### 3.3. Результаты моделирования

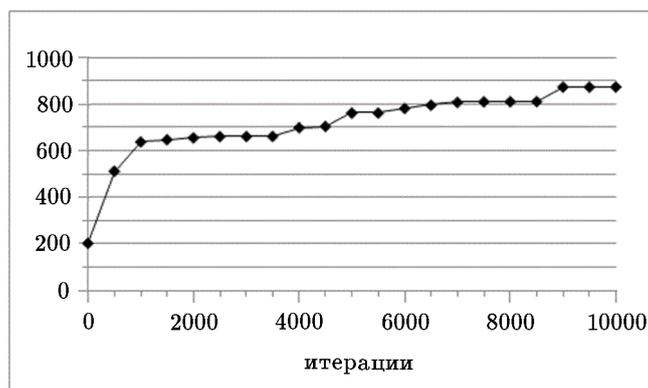
Экспериментальное моделирование процесса просачивания производилось в два этапа. На первом этапе просачивание моделировалось в последовательном режиме для фрагмента клеточного массива размером  $200 \times 200 \times 500$  на компьютере Intel Core-i7 (2,66 ГГц). На этом этапе были отлажены программные модули операторов конвекции и

диффузии, а также определено необходимое количество повторений оператора диффузии для обеспечения полной гладкости поверхности жидкости.

Параллельная версия исследовалась на кластерах NKS-30T Intel Xeon 5540 (ССКЦ ИВМиМГ СО РАН, Новосибирск) и MVS-10p Intel Xeon Phi 7120D ОСКЦ (Москва). Были приняты следующие начальные условия: в почве все поры пусты, на поверхность почвы налито некоторое количество жидкости  $16 \cdot 10^6$  частиц (60 грамм). Граничные условия вдоль осей  $i$  и  $j$  — периодические. Программирование выполнялось на языке Си<sup>++</sup> с использованием библиотеки MPI [32].

Исследовались следующие параметры и свойства параллельных реализаций СКА-модели.

1. *Глубина и скорость просачивания жидкости в почву.* Зависимость глубины проникания от числа итераций (модельного времени) имеет вид кривой (рис. 6). На эту зависимость не влияет выбор параметра алгоритма моделирования СКА (режима, количества параллельных процессов и топологии их соединений). Из рис. 6 видно, что скорость просачивания меняется с глубиной, что свидетельствует о неоднородности пористой структуры. В частности видно, что на глубинах  $k \approx 828$  и  $k \approx 660$  существуют пустоты.



**Рис. 6.** Зависимость глубины проникания жидкости от времени при реализации модели в режиме  $(\beta, \gamma)$  на  $P = 4 \times 10 = 40$  процессорах

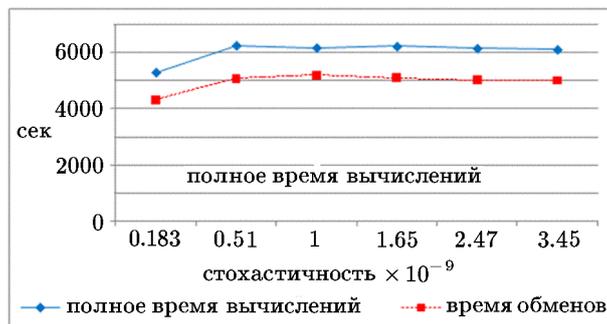
2. *Эффективность распараллеливания в зависимости от стохастичности алгоритма.* На рис. 7 показаны полученные в результате моделирования значения полного времени вычисления и времени, теряемого на межпроцессных обменах, для разных значений стохастичности модели. Моделирование проводилось параллельно на 10 ядрах кластера MVS-10p при одномерной декомпозиции (рис. 4 а)).

Линии на рис. 7 показывают, что эффективность

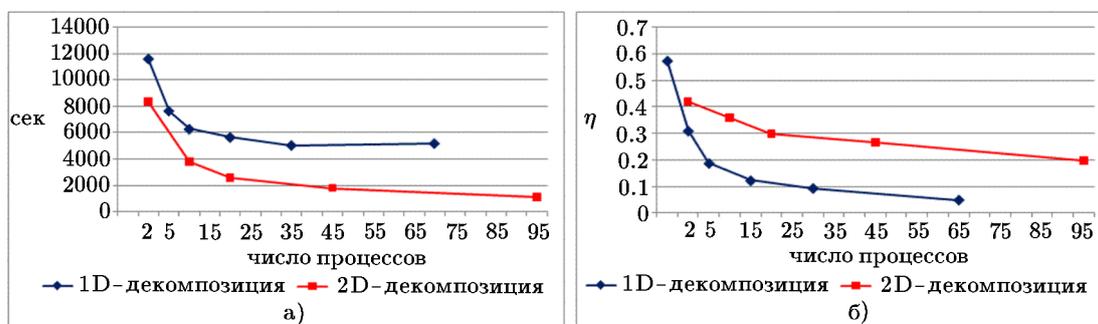
$$\eta = \frac{\text{Time}_P - \text{Time}_{\text{ex}}}{\text{Time}_P}$$

практически не зависит от выбора размера  $\beta$ -блока и не превышает 0.2 при размерах  $\beta$ -блока более минимального ( $m > 3^2$ ).

3. *Ускорение вычислений в зависимости от числа процессоров и топологии коммуникации ядер.* Моделирование производилось на кластере MVS-10p при двух вариантах декомпозиции пространства моделирования. Из показанных на рис. 8 результатов моделирования видно, насколько эффективнее вычисления при двумерной декомпозиции. Кроме того, видно, что распараллеливание на  $P \geq 10$  процессов для рассматриваемого СКА нецелесообразно.



**Рис. 7.** Зависимость времени вычислений от стохастичности СКА-модели  $\aleph_{\beta,\gamma}$  на  $P = 10$  процессорах при одномерной декомпозиции пространства моделирования. Значения стохастичности соответствуют следующим размерам  $\beta$ -блоков СКА:  $|B_\beta| = \{3^2, 5^2, 7^2, 9^2, 11^2, 13^2\}$



**Рис. 8.** Зависимости а) времени выполнения  $T = 1000$  итераций и б) эффективности параллельной реализации от количества параллельных процессоров при одномерной и двумерной декомпозиции области моделирования

### 4. Заключение

Параллельная реализация стохастической модели просачивания позволила получить конкретные зависимости стохастичности и эффективности от параметров параллельной реализации и оценить их влияние на основные характеристики модели, что необходимо для обоснованного выбора параметров модели. Результатом проведенных исследований можно считать следующие положения:

- 1) для эффективной параллельной реализации пространственно стохастического СКА необходимо преобразование его к блочно-синхронному режиму ( $\beta$ -режиму);
- 2) стохастичность СКА зависит от выбора размера  $\beta$ -блока, увеличение которого слабо влияет на ускорение вычисления;
- 3) условия корректности параллельных вычислений требует согласованности стадий  $\beta$ -режима во всех процессорах.

### Литература

1. **Воссара N.** Reaction–Difusion Complex Systems. — Berlin: Springer, 2004.
2. **Ванаг В.К.** Диссипативные структуры в реакционно–диффузионных системах. Эксперимент и теория. — Ижевск: ИКИ, 2008.

3. **Adamatzky A.** Reaction–Diffusion Automata: Phenomenology, Localisations, Computation. — Heidelberg: Springer, 2013.
4. **Ziff R.M., Gulari E., and Barshad Y.** Kinetic phase transitions in an irreversible surface-reaction model // *Phys. Rev. Lett.* — 1986. — Vol. 56, № 24. — P. 2553–2556.
5. **Chatterjee A., Vlachos D.G.** An overview of spatial microscopic and accelerated kinetic Monte-Carlo methods // *J. Comp. Aided Matter Des.* — 2007. — Vol. 14. — P. 253–308.
6. *Advances in Chemical Physics: Monte Carlo methods in chemical physics, Volume 105 / I. Prigogine, S.A. Rice.* — Willey & Sons Inc., 2009.
7. **Matveev A.V., Latkin E.I., Elokhin V.I., and Gorodetskii V.V.** Turbulent and stripes wave patterns caused by limited  $\text{CO}_{ads}$  diffusion during CO oxidation over Pd(110) surface: kinetic Monte Carlo studies // *Chem. Eng. J.* — 2005. — Vol. 107. — P. 181–189.
8. **Елохин В.И., Матвеев А.В., Городецкий В.В.** Автоколебания и химические волны в реакции окисления СО на Pt и Pd: кинетические модели Монте-Карло // *Кинетика и катализ.* — 2009. — Т. 50, № 1. — С. 45–53.
9. **Ванаг В.К.** Исследование пространственно-распределенных динамических систем методами вероятностного клеточного автомата // *Успехи физических наук.* — 1999. — Т. 169, № 5. — С. 481–505.
10. **Алексеев Д.В., Казунина Г.А., Чередниченко А.В.** Клеточно-автоматное моделирование процесса разрушения хрупких материалов // *Прикладная дискретная математика.* — 2015. — Т. 28, № 2. — С. 103–117.
11. **Nurminen L., Kuronen A., and Kaski K.** Kinetic Monte Carlo simulation of nucleation on patterned substrates // *Phys. Rev. B.* — 2000. — Vol. 63. — P. 035407-1–035407-7.
12. **Витвицкий А.А.** Клеточные автоматы с динамической структурой для моделирования роста биологических тканей // *Сиб. журн. вычисл. математики / РАН. Сиб. отд-ние.* — Новосибирск, 2014. — Т. 17, № 4. — С. 315–327. — Перевод: Vitvitsky A.A. Cellular automata with a dynamic structure for simulate the growth of biological tissues // *Numerical Analysis and Applications.* — 2014. — Vol. 7, № 4. — P. 263–273.
13. **Бандман О.Л.** Клеточно-автоматные модели естественных процессов и их реализация на современных компьютерах // *Прикладная дискретная математика.* — 2017. — № 35. — С. 102–121.
14. **Бандман О.Л., Киреева А.Е.** Стохастическое клеточно-автоматное моделирование колебаний и автоволн в реакционно-диффузионных системах // *Сиб. журн. вычисл. математики / РАН. Сиб. отд-ние.* — Новосибирск, 2015. — Т. 18, № 3. — С. 255–274. — Перевод: Bandman O.L., Kireeva A.E. Stochastic cellular automata simulation of oscillations and autowaves in reaction-diffusion systems // *Numerical Analysis and Applications.* — 2015. — Vol. 8, № 3. — P. 208–222.
15. **Wolfram S.** An Elementary Introduction to the Wolfram Language. — USA : Wolfram Media Inc., 2017.
16. Wolfram MATHEMATICA. — <https://www.wolfram.com/mathematica/>.
17. **Lubachevsky B.D.** Efficient parallel simulations of asynchronous cellular arrays // *Complex Systems.* — 1987. — Vol. 1, № 6. — P. 1099–1123.
18. **Overeinder B.J., Sloot P.M.A.** Application of time warp to parallel simulations with asynchronous cellular automata // *Proc. European Simulation Symposium.* — The Netherlands, 1993. — С. 397–402.
19. **Bandman O.** Parallel Simulation of Asynchronous Cellular Automata Evolution // *Cellular Automata. Proc. ACRI 2006.* — Springer, 2006. — P. 41–48. — (Lecture Notes in Computer Science; LNCS 4173).

20. **Kireeva A.** Parallel Implementation of Totalistic Cellular Automata Model of Stable Patterns Formation // Parallel Computing Technologies. Proc. PaCT 2013 / V. Malyshkin. — Springer, 2013. — P. 330–343. — (Lecture Notes in Computer Science; LNCS 7979).
21. **Bandman O.** Implementation of large-scale cellular automata models on multi-core computers and clusters // Proc. HPCS 2013. — Helsinki, 2013. — P. 304–310.
22. **Achasova S., Bandman O., Markova V., and Piskunov S.** Parallel Substitution Algorithm. Theory and Application. — Singapore: World Scientific, 1994.
23. **Бандман О.Л.** Клеточно-автоматные модели пространственной динамики // Системная информатика: сб. научн. тр. — Новосибирск: Изд-во СО РАН, 2006. — Вып 10. — С. 59–111.
24. **Ачасова С.М., Бандман О.Л.** Корректность параллельных процессов. — Новосибирск: Наука, 1990.
25. **Bandman O.** Parallel composition of asynchronous cellular automata simulating reaction-diffusion processes // Proc. ACRI 2010. — Springer, 2010. — P. 395–398. — (Lecture Notes in Computer Science; LNCS 6350).
26. **Bandman O.** Parallelization efficiency versus stochasticity in simulation reaction-diffusion by cellular automata // J. of Supercomputing. — 2016. — Vol. 73, iss. 2. — P. 687–699.
27. **Бандман О.Л.** Режимы функционирования асинхронных клеточных автоматов, моделирующих нелинейную пространственную динамику // Прикладная дискретная математика. — 2015. — Т. 27, № 1. — С. 105–119.
28. **Von Neumann J.** Theory of Self-Reproducing Automata / Arthur W. Burks. — London: Univ. of Illinois Press, 1966.
29. **Wolfram S.** Cellular Automata and Complexity: Collected Papers. — USA: Westview Press, 2009.
30. **Bandman O.** 3D Cellular Automata Model of Fluid Permeation through Porous Material // Parallel Computing Technologies. Proc. PaCT 2013 / V. Malyshkin. — Springer, 2013. — P. 278–290. — (Lecture Notes in Computer Science; LNCS 7979).
31. **Nedea S.V., Lukkien J., Jansen A., and Hilbers P.** Methods for Parallel Simulations of Surface Reactions. — URL: <http://arXiv.org> > physics > arXiv:physics/0209017.
32. **Оленев Н.Н.** Основы параллельного программирования в системе MPI. — М.: Изд-во ВЦ РАН, 2005.

*Поступила в редакцию 10 августа 2017 г.,  
в окончательном варианте 30 августа 2017 г.*

## Литература в транслитерации

1. **Воссара N.** Reaction–Diffusion Complex Systems. — Berlin: Springer, 2004.
2. **Vanag V.К.** Dissipativnye struktury v reaktsionno–diffuzionnyh sistemah. Eksperiment i teoriya. — Izhevsk: IKI, 2008.
3. **Adamatzky A.** Reaction–Diffusion Automata: Phenomenology, Localisations, Computation. — Heidelberg: Springer, 2013.
4. **Ziff R.M., Gulari E., and Barshad Y.** Kinetic phase transitions in an irreversible surface-reaction model // Phys. Rev. Lett. — 1986. — Vol. 56, № 24. — P. 2553–2556.
5. **Chatterjee A., Vlachos D.G.** An overview of spatial microscopic and accelerated kinetic Monte-Carlo methods // J. Comp. Aided Matter Des. — 2007. — Vol. 14. — P. 253–308.
6. Advances in Chemical Physics: Monte Carlo methods in chemical physics, Volume 105 / I. Prigogine, S.A. Rice. — Willey & Sons Inc., 2009.

7. **Matveev A.V., Latkin E.I., Elokhin V.I., and Gorodetskii V.V.** Turbulent and stripes wave patterns caused by limited  $\text{CO}_{ads}$  diffusion during CO oxidation over Pd(110) surface: kinetic Monte Carlo studies // *Chem. Eng. J.* — 2005. — Vol. 107. — P. 181–189.
8. **Elohin V.I., Matveev A.V., Gorodetskiy V.V.** Avtokolebaniya i himicheskie volny v reaktsii okisleniya CO na Pt i Pd: kineticheskie modeli Monte-Karlo // *Kinetika i kataliz.* — 2009. — Т. 50, № 1. — S. 45–53.
9. **Vanag V.K.** Issledovanie prostranstvenno-raspredeennykh dinamicheskikh sistem metodami veroyatnostnogo kletochnogo avtomata // *Uspekhi fizicheskikh nauk.* — 1999. — Т. 169, № 5. — S. 481–505.
10. **Alekseev D.V., Kazunina G.A., Cherednichenko A.V.** Kletochno-avtomatnoe modelirovanie protsessa razrusheniya hrupkikh materialov // *Prikladnaya diskretnaya matematika.* — 2015. — Т. 28, № 2. — S. 103–117.
11. **Nurminen L., Kuronen A., and Kaski K.** Kinetic Monte Carlo simulation of nucleation on patterned substrates // *Phys. Rev. B.* — 2000. — Vol. 63. — P. 035407-1–035407-7.
12. **Vitvitskiy A.A.** Kletochnye avtomaty s dinamicheskoy strukturoy dlya modelirovaniya rosta biologicheskikh tkaney // *Sib. zhurn. vychisl. matematiki / RAN. Sib. otd-nie.* — Novosibirsk, 2014. — Т. 17, № 4. — S. 315–327. — Перевод: Vitvitskiy A.A. Cellular automata with a dynamic structure for simulate the growth of biological tissues // *Numerical Analysis and Applications.* — 2014. — Vol. 7, № 4. — P. 263–273.
13. **Bandman O.L.** Kletochno-avtomatnye modeli estestvennykh protsessov i ih realizatsiya na sovremennykh komp'yuterah // *Prikladnaya diskretnaya matematika.* — 2017. — № 35. — S. 102–121.
14. **Bandman O.L., Kireeva A.E.** Stokhasticheskoe kletochno-avtomatnoe modelirovanie kolebaniy i avtovoln v reaktsionno-diffuzionnykh sistemah // *Sib. zhurn. vychisl. matematiki / RAN. Sib. otd-nie.* — Novosibirsk, 2015. — Т. 18, № 3. — S. 255–274. — Перевод: Bandman O.L., Kireeva A.E. Stochastic cellular automata simulation of oscillations and autowaves in reaction-diffusion systems // *Numerical Analysis and Applications.* — 2015. — Vol. 8, № 3. — P. 208–222.
15. **Wolfram S.** An Elementary Introduction to the Wolfram Language. — USA : Wolfram Media Inc., 2017.
16. Wolfram MATHEMATICA. — <https://www.wolfram.com/mathematica/>.
17. **Lubachevsky B.D.** Efficient parallel simulations of asynchronous cellular arrays // *Complex Systems.* — 1987. — Vol. 1, № 6. — P. 1099–1123.
18. **Overeinder B.J., Sloot P.M.A.** Application of time warp to parallel simulations with asynchronous cellular automata // *Proc. European Simulation Symposium.* — The Netherlands, 1993. — C. 397–402.
19. **Bandman O.** Parallel Simulation of Asynchronous Cellular Automata Evolution // *Cellular Automata. Proc. ACRI 2006.* — Springer, 2006. — P. 41–48. — (Lecture Notes in Computer Science; LNCS 4173).
20. **Kireeva A.** Parallel Implementation of Totalistic Cellular Automata Model of Stable Patterns Formation // *Parallel Computing Technologies. Proc. PaCT 2013 / V. Malyshev.* — Springer, 2013. — P. 330–343. — (Lecture Notes in Computer Science; LNCS 7979).
21. **Bandman O.** Implementation of large-scale cellular automata models on multi-core computers and clusters // *Proc. HPCS 2013.* — Helsinki, 2013. — P. 304–310.
22. **Achasova S., Bandman O., Markova V., and Piskunov S.** Parallel Substitution Algorithm. Theory and Application. — Singapore: World Scientific, 1994.
23. **Bandman O.L.** Kletochno-avtomatnye modeli prostranstvennoy dinamiki // *Sistemnaya informatika: sb. nauchn. tr.* — Novosibirsk: Izd-vo SO RAN, 2006. — Вып. 10. — S. 59–111.

24. **Achasova S.M., Bandman O.L.** Korrektnost' parallel'nyh protsessov. — Novosibirsk: Nauka, 1990.
25. **Bandman O.** Parallel composition of asynchronous cellular automata simulating reaction-diffusion processes // Proc. ACRI 2010. — Springer, 2010. — P. 395–398. — (Lecture Notes in Computer Science; LNCS 6350).
26. **Bandman O.** Parallelization efficiency versus stochasticity in simulation reaction-diffusion by cellular automata // J. of Supercomputing. — 2016. — Vol. 73, iss. 2. — P. 687–699.
27. **Bandman O.L.** Rezhimy funktsionirovaniya asinhronnyh kletochnyh avtomatov, modeliruyushchih nelineynuyu prostranstvennuyu dinamiku // Prikladnaya diskretnaya matematika. — 2015. — T. 27, № 1. — S. 105–119.
28. **Von Neumann J.** Theory of Self-Reproducing Automata / Arthur W. Burks. — London: Univ. of Illinois Press, 1966.
29. **Wolfram S.** Cellular Automata and Complexity: Collected Papers. — USA: Westview Press, 2009.
30. **Bandman O.** 3D Cellular Automata Model of Fluid Permeation through Porous Material // Parallel Computing Technologies. Proc. PaCT 2013 / V. Malyshkin. — Springer, 2013. — P. 278–290. — (Lecture Notes in Computer Science; LNCS 7979).
31. **Nedea S.V., Lukkien J., Jansen A., and Hilbers P.** Methods for Parallel Simulations of Surface Reactions. — URL: <http://arXiv.org> > physics > arXiv:physics/0209017.
32. **Olenev N.N.** Osnovy parallel'nogo programirovaniya v sisteme MPI. — M.: Izd-vo VTS RAN, 2005.

