

Использование гибридного клеточного автомата для пространственного моделирования злокачественных образований

Ю.Б. Адмиральский

МГУ им. М.В. Ломоносова

Задачи исследования

Тематика исследования является актуальной:

- Применение достаточно подробных моделей на основе уравнений в частных производных для описания клеток опухоли приводит к необходимости решать сложные задачи со свободной границей¹;
- Построение модели, пригодной для расчета больших опухолей, ставит вопросы масштабирования алгоритмов, которые не рассматриваются в большинстве работ по тематике.

При выполнении исследования преследовались следующие цели:

- 1 Построение гибридной модели путем объединения уравнения диффузии (описание распространения кислорода и белков) и классического клеточного автомата (описание клеток опухоли);
- 2 Построение клеточного автомата, реализующего подзадачу деления клеток, анализ реализации.

¹*H. M. Byrne, M. A. J. Chaplain*. Free boundary value problems associated with the growth and development of multicellular spheroids. 1997.

Модель диффузии

Основой для модели диффузии выступит рассматриваемая на области D (прямоугольной в двухмерном случае) краевая задача для уравнения диффузии:

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial t}(\bar{x}, t) = D_0(t)\Delta u(\bar{x}, t) + f(\bar{x}, t), \quad \bar{x} \in D, t \in (t_0, T]; \\ u(\bar{x}, 0) = u_0(\bar{x}), \quad \bar{x} \in \bar{D}; \\ \left. \frac{\partial u}{\partial \bar{n}}(\bar{x}, t) \right|_{\bar{x} \in \partial D} = \psi(\bar{x}, t), \quad t \in [t_0, T]. \end{array} \right. \quad (1)$$

Клеточный автомат

Определение

Клеточным автоматом мы назовём совокупность объектов $(C, S, \mathcal{N}(\bar{r}), \varphi)$, где:

- C — регулярная сетка клеток–автоматов (считаем кубической);
- S — конечное непустое множество состояний;
- $\mathcal{N}(\bar{r})$ — локальное множество соседства:

$$\mathcal{N}(\bar{r}) = \{\bar{r} + \Delta_i \mid i = \overline{1, \dots, M}\};$$

- φ — функция перехода:

$$S_t(\bar{r}) = \varphi(S_{t-1}(\mathcal{N}(\bar{r}))).$$

Диффузия и КА

Область D разбивается на подобласти D_{ij} (для двумерного случая квадраты со стороной порядка размера одной клетки):

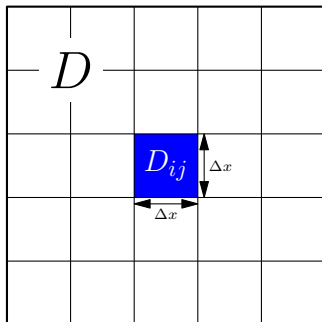


Рис.: Разбиение области D .

Диффузия и КА

Уравнение дискретизируется по пространству путем рассмотрения следующих функций:

$$Q_{ij}(t) = \iint_{D_{ij}} u(\xi, t) d\xi; \quad Q_{ij,0} = \iint_{D_{ij}} u(\xi, t_0) d\xi; \quad F_{ij}(t) = \iint_{D_{ij}} f(\xi, t) d\xi.$$

Для двумерного случая, преобразуя уравнение и дискретизируя его по времени, можно получить явные выражения для Q_{ij}^t вида:

$$Q_{ij}^{k+1} = Q_{ij}^k + D_0 \left(h_{ij}^{x,k} + h_{ij}^{y,k} - h_{(i-1)j}^{x,k} - h_{i(j-1)}^{y,k} \right) \Delta t + F_{ij}^k \Delta t + O(\Delta t^2),$$

где $h_{ij}^{x,t}$ зависит от Q_{i+1j}^t и Q_{ij}^t , а $h_{ij}^{y,t}$ зависит от Q_{ij+1}^t и Q_{ij}^t .

Такую дискретную модель можно применять вместе с КА, сопоставив каждую переменную Q_{ij}^k и F_{ij}^k ячейке клеточного автомата и используя функцию перехода, зависящую от этих переменных.

Клеточный цикл

Одна из основных задач клеточного автомата в данной модели — реализация клеточного цикла. Для описания динамики делений клеток мы используем следующее подмножество состояний клеточного цикла:

- G_0 — специализированное состояние, соответствующее клетке, которая не пытается делиться по каким-либо причинам (в т.ч. потому, что уже недавно произвела деление);
- D — состояние деления, соответствующее клетке, которая проходит стадии подготовки и деления (фазы G_1 , S , G_2 и митоз);
- H — состояние гипоксии, возникающее при нехватке кислорода;
- N — состояние некроза, при долгом отсутствии кислорода и питательных веществ.

Клеточный цикл

Состояния и переходы клеточного цикла:

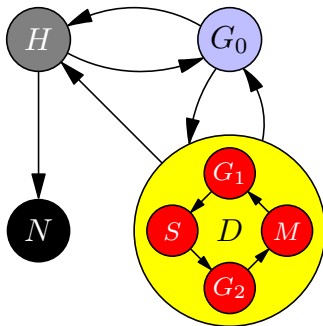


Рис.: Схема клеточного цикла.

При расчете переходов учитывается количество итераций, в течение которых клетка сохраняла свое состояние.

Другие состояния КА

Вторая задача клеточного автомата — обозначать границу опухоли. Для этого рассматриваем дополнительные состояния:

- F — ячейка является свободной и может быть использована для размещения клетки–потомка для какой–либо из делящихся клеток по–соседству (пустое пространство или внеклеточный матрикс);
- C — ячейка, обозначающая капилляр.

В рамках рассмотрения задачи деления, для удобства, выделим из клеточного цикла три дополнительных состояния:

- D^* — клетка на заключительной стадии деления, (в рамках модели — состояние, соответствующее клетке, для которой должна быть выполнена попытка деления на текущей итерации);
- D^+ — успешно разделившаяся клетка;
- G_0^* — состояние, в которое переходит клетка–потомок сразу после деления (в рамках модели - состояние в которое выполняется переход из F при успешном делении соседней клетки).

Другие состояния КА

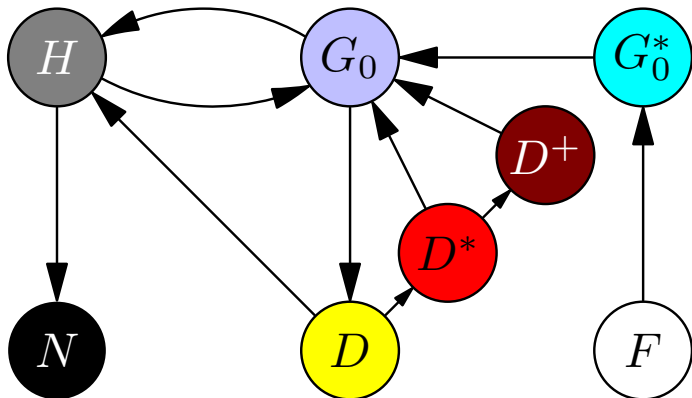


Рис.: Схема клеточного цикла с дополнительными состояниями.

Пример моделирования

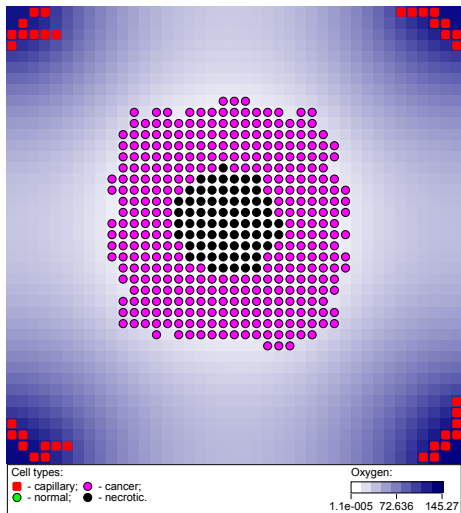


Рис.: Модель небольшой растущей опухоли с образовавшимся некротическим ядром.

Параллельные вычисления

ГКА (как и обычный клеточный автомат) хорошо подходит для параллельных вычислений с распределенной памятью (на отдельных, связанных в сеть, вычислительных узлах).

При среднем размере клетки опухоли в 50 мкм, для расчета опухоли занимающей объем куба со стороной 5 см, потребуется клеточный автомат с 10^9 ячейками.

При совершенствовании модели важно сохранять возможность параллельных вычислений с распределенной памятью.

Задача деления

Рассмотрим подзадачу деления клетки, в рамках которой упростим схему переходов:

- Ячейка в состоянии G_0 переходит в состояние G_0 ;
- Ячейка в состоянии D переходит в состояние D ;
- Ячейка в состоянии F может перейти как в состояние F (если нет деления соседней клетки в данную свободную ячейку), так и в G_0^* (при успешном делении соседней клетки);
- Ячейка в состоянии D^* переходит в состояние D^+ (при успешном делении в соседнюю свободную ячейку) или в состояние G_0 (иначе).

Такое упрощение применимо при рассмотрении отдельной итерации клеточного автомата.

Простой алгоритм деления

Рассмотрим простой алгоритм, реализующий деление клетки²:

- 1 Последовательно для каждой ячейки, находящейся в состоянии D^* (клетка, готовая к делению), ищем соседнюю ячейку в состоянии F (свободное место). Поиск завершается либо нахождением свободного места, либо выводом о том, что свободного места в пределах множества соседства нет;
- 2 Если свободное место найдено, тогда ячейка переходит в состояние D^+ , а найденная свободная ячейка переходит в состояние G_0^* ;
- 3 Если свободного места не найдено, тогда ячейка переходит в состояние G_0 (контактное торможение).

Возникает сложность — результат выполнения шага 1 алгоритма зависит от результата выполнения шага 2 для всех ранее рассмотренных ячеек в состоянии D^* . В таком виде алгоритм не реализуется клеточным автоматом, поскольку невозможно указать функцию перехода, реализующую шаг 2, и явно зависящую только от соседних ячеек найденной на шаге 2 ячейки.

²A. R. Kansal et al. Simulated Brain Tumor Growth Dynamics Using a Three-Dimensional Cellular Automaton. 2000.

Простой алгоритм деления

При параллельном исполнении шага 2, более одного параллельно работающего вычислителя может выбрать одну и ту же ячейку в состоянии F , и перевести все ячейки, для которых выполнялся поиск, в состояние D^+ (даже если есть другие свободные ячейки, которые были бы найдены, если бы вычислитель продолжил поиск).

Эта сложность приводит к необходимости построения алгоритма, подходящего для параллельного исполнения, который может быть реализован функцией перехода КА.

Задача деления — основной алгоритм

Автором был предложен и применен **алгоритм коллективного перебора направлений**:

- 1 Зафиксируем одно из направлений (и соответствующую ему соседнюю ячейку КА). Для каждой ячейки в состоянии D^* проверяем, находится ли фиксированная ячейка в состоянии F . Если да, то переводим рассматриваемую ячейку в состояние D^+ . Иначе, оставляем ячейку в состоянии D (преобразование F_2);
- 2 Для каждой ячейки в состоянии F проверяем соседнюю ячейку в противоположном направлении, и если она находится в состоянии D^* , то переводим рассматриваемую ячейку в состояние G_0^* . Иначе, оставляем ячейку в состоянии F (преобразование F_1);
- 3 Синхронизируем данные о результатах выполнения предыдущих шагов между вычислителями;
- 4 Повторяем шаги 1, 2 и 3 последовательно для каждого направления.

Шаги 1 и 2 можно выполнять параллельно, поскольку для каждой ячейки в состоянии F может рассматриваться деление не более одной соседней клетки. Количество направлений (а значит итераций цикла 1-4) не зависит от размеров расчетной области, а зависит только от её размерности и того, какие ячейки считаются соседними.

Задача деления — основной алгоритм

Задача деления разбилась на последовательные подзадачи деления в фиксированном направлении (субитерации). Преобразование для каждой субитерации можно реализовать с помощью клеточного автомата как композицию преобразований F_2 и F_1 :

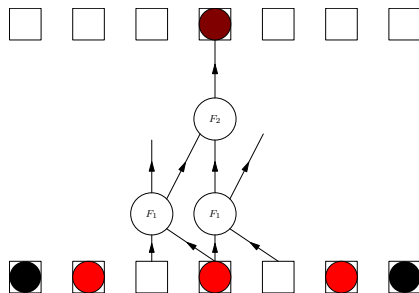


Рис.: Схема функции перехода для КА.

Видно, что результат преобразования зависит только от двух соседних ячеек.

Добавление случайности

Для избежания появления «предпочтительных направлений» требуется добавить случайность в перебор направлений для деления. При этом требуется:

- воспроизводимости численных экспериментов (используя ГПСЧ и сохраняя его начальное состояние);
- независимость генерируемой последовательности от распределения ячеек ГКА по вычислителям и от количества этих вычислителей.

Простейший вариант — случайным образом коллективно перебирать направления, используя общий (для всех ячеек ГКА) ГПСЧ;

- (+) требуется всего один ГПСЧ, который будет (с использованием общего начального состояния) рассчитываться каждым вычислителем независимо от остальных;
- (+) линейная зависимость вычислительной сложности одной итерации от размеров области соседства ГКА;
- (-) не обеспечивает решения проблемы «предпочтительных направлений» при большом количестве делящихся на данной итерации клеток.

Добавление случайности

Более сложный вариант — выбирать своё случайное направление для каждой из делящихся клеток, осуществляя на каждой субитерации коллективного перебора направлений только клеток с совпадающим собственным направлением деления.

- (+) обеспечивает требуемую «равномерность» деления клеток по направлениям;
- (-) требует большое количество различных ГПСЧ (равное числу ячеек ГКА);
- (-) возможны «промахи» когда клетка, в результате деления других клеток, не может поместить потомка в соответствии с случайно выбранным ранее собственным направлением, но может в другом (которое проверялось в одной из предыдущих субитераций). Решение этой проблемы приводит к квадратичной зависимости вычислительной сложности одной итерации от размеров области соседства ГКА;

Добавление случайности

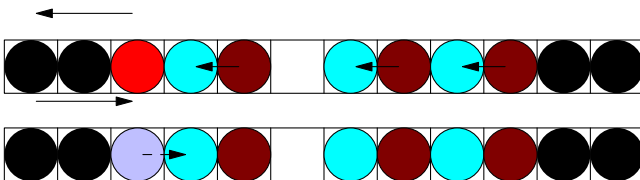
Вместо использования отдельного ГПСЧ для каждой из ячеек, предлагается использовать отдельные ГПСЧ с малым вектором состояния (например, LCG или ICG) для каждой строки и столбца. Для каждой ячейки ГПСЧ получается составным от соответствующих ячейке ГПСЧ строки и столбца — путём применения обратного преобразования Смирнова.

Оптимальное деление

Рассмотрим решение, которое находит алгоритм коллективного перебора направлений для следующей конфигурации:



Ход работы алгоритма выглядит следующим образом:



Оптимальное (с точки зрения количества разделившихся клеток) выглядит следующим образом:



Оптимальное деление

Возникает вопрос - **можно ли добиться оптимального (максимизирующего количество разделившихся клеток) решения для произвольной начальной конфигурации делящихся и свободных клеток за одну итерацию КА?**

Для данной задачи автором доказано, что не существует клеточного автомата, который для произвольной одномерной начальной конфигурации размера $n > 1$ находит решение задачи оптимального деления за одну итерацию и использует информацию менее чем о $\frac{n-1}{2}$ соседях.

Оптимальное деление

Возникает вопрос - **можно ли добиться оптимального (максимизирующего количество разделившихся клеток) решения для произвольной начальной конфигурации делящихся и свободных клеток за одну итерацию КА?**

Для данной задачи автором доказано, что не существует клеточного автомата, который для произвольной одномерной начальной конфигурации размера $n > 1$ находит решение задачи оптимального деления за одну итерацию и использует информацию менее чем о $\frac{n-1}{2}$ соседях.

Для решения задачи используем, то что мы ищем соответствие между ячейками в состоянии D^* и F . Для заданной начальной конфигурации ячеек можно построить граф, вершины которого будут соответствовать ячейкам КА в состоянии D^* и F . Вершины, соответствующие двум ячейкам КА в состояниях D^* и F , соединены ребром, если ячейки являются соседними.

Оптимальное деление

Поскольку при таком построении графа никакие две вершины, соответствующие ячейкам в одинаковом состоянии, не соединены ребром, граф является двудольным. Оптимальному делению, в таком случае, будет соответствовать любое наибольшее паросочетание в этом графе.

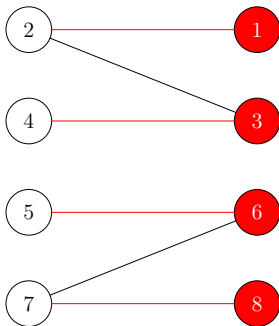
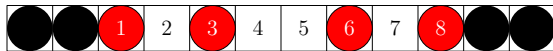


Рис.: Конфигурация и соответствующий граф.

Оптимальное деление

Известные параллельные алгоритмы поиска наибольшего паросочетания в двудольном графе для вычислительных систем с распределённой памятью имеют проблемы с масштабированием в общем случае³.

³*A. Azad, A. Buluç.* Distributed-Memory Algorithms for Maximum Cardinality Matching in Bipartite Graphs. 2016.

Список литературы

- [1] *K. M. Zapolski, Y. B. Admiralskiy, A. S. Bratus.* Hybrid Cellular Automaton Method for Homogeneous Tumour Growth Modelling. // *Russian Journal of Numerical Analysis and Mathematical Modelling*, vol. 29, no. 5, pp. 319–329, 2014.
- [2] *H. M. Byrne, M. A. J. Chaplain.* Free boundary value problems associated with the growth and development of multicellular spheroids. // *European Journal of Applied Mathematics*, vol. 8, no. 6, pp. 639–658, 1997.
- [3] *A. R. Kansal et al.* Simulated Brain Tumor Growth Dynamics Using a Three-Dimensional Cellular Automaton. // *Journal of Theoretical Biology*, vol. 203, no. 4, pp. 367–382, 2000.
- [4] *A. Azad, A. Buluç.* Distributed-Memory Algorithms for Maximum Cardinality Matching in Bipartite Graphs. // *IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 32–42, 2016.