

С. М. Ачасова*(Новосибирск)***АЛГОРИТМ ПАРАЛЛЕЛЬНЫХ ПОДСТАНОВОК
САМОВОСПРОИЗВОДЯЩЕЙСЯ В ДВУХ НАПРАВЛЕНИЯХ ПЕТЛИ**

Разработан алгоритм параллельных подстановок (АПП), который моделирует процесс самовоспроизведения в клеточном пространстве в двух направлениях (снизу вверх и слева направо) структуры, имеющей форму четырехугольной петли. Предложенная АПП-модель самовоспроизведения существенно более лаконична по сравнению с классической клеточно-автоматной моделью.

Введение. Исследования самовоспроизведения на логическом уровне начал фон Нейман. Он создал клеточный автомат (КА), который являлся универсальным конструктором и универсальным вычислителем: мог вычислить любую вычислимую функцию и сконструировать в клеточном пространстве рядом с собой любой другой КА по заданному описанию [1]. Если была описана сама универсальная машина, то это был случай самовоспроизведения.

Далее Кодд [2] упростил неймановскую универсальную машину, уменьшив число клеточных состояний с 29 до 8. Но это все еще была весьма громоздкая структура с огромным числом правил изменения состояния клетки, и ее логическая организация являлась достаточной для самовоспроизведения.

Дальнейшим важным шагом в развитии истории самовоспроизведения была работа Лангтона [3], создавшего на основе фрагмента универсального конструктора Кодда – периодического излучателя, являвшегося тактирующим устройством, – простую клеточно-автоматную структуру в виде прямоугольной петли, которая, не будучи универсальной, обладала способностью к самовоспроизведению, и ее логическая организация являлась только необходимой для самовоспроизведения. Основная идея Лангтона – хранить самописание динамически, в виде циркулирующей в петле последовательности состояний клеток.

Чтобы исключить случаи тривиального реплицирования первоначальной структуры, т. е. построения копии в результате выполнения некоторой последовательности правил перехода, где «само» не является ключевым понятием, был сформулирован критерий истинного самовоспроизведения, т. е. процесса построения копии, активно управляемого информацией, записанной в самой копирующейся структуре. Этот критерий состоит в двойственном использовании информации, вставленной в клеточно-автоматную

структуру: информация должна служить программой для построения копии и переписываться в дочернюю структуру как неинтерпретируемые данные.

Хотя петля Лангтона – довольно простое устройство, она имеет несколько сотен правил перехода клетки из одного состояния в другое. Последовали упрощения [4, 5], но и упрощенные петли содержали несколько десятков правил перехода. В работе [6] предложено использовать для моделирования самовоспроизводящейся клеточно-автоматной петли Алгоритм параллельных подстановок (АПП) [7, 8]. Это – расширенная парадигма классического КА, которая по сравнению с ним имеет дополнительные возможности, а именно: произвольный шаблон подстановки, функциональные подстановки, одна подстановка может изменять состояния сразу нескольких клеток. Благодаря этим свойствам описание процесса самовоспроизведения в нотации АПП получается более коротким и структурированным по сравнению с описанием на языке классического КА. Представленная в [6] линейная АПП-модель процесса самовоспроизведения петли (построение копий идет слева направо от исходной структуры) имеет почти в 3 раза меньше правил перехода, чем в исходной КА-модели, и обходится 6 состояниями клетки по сравнению с 9 в исходной модели. В данной работе представляется АПП-модель процесса самовоспроизведения петли в двух направлениях (слева направо и снизу вверх), основанная на линейной модели [6].

Алгоритм параллельных подстановок для самовоспроизводящейся в двух направлениях петли. В данной работе АПП функционирует в двумерном клеточном пространстве, которое представляется прямоугольной координатной сеткой с клетками в ее узлах. Клетка именуется парой координат и в каждый момент дискретного времени находится в одном из множества возможных состояний. Переход клетки в новое состояние определяется локальным правилом – подстановкой. Левая часть подстановки определяет условие ее применимости, правая часть задает новые состояния клеткам так называемой базы подстановки. Клетки, состояния которых не изменяются в результате выполнения этой подстановки, образуют ее контекст. Все применимые в некотором такте дискретного времени подстановки выполняются одновременно. Поскольку каждая подстановка может содержать более чем одну базовую клетку, то возможна ситуация, в которой одна и та же клетка оказывается в зоне применимости двух подстановок. Если эта клетка в одной подстановке является базовой, а в другой контекстной, то не возникает проблема с изменением ее состояния. Если же клетка является базовой в обеих подстановках и подстановки изменяют ее состояние по-разному, то возникает противоречие в применимости подстановок, в корректном множестве подстановок противоречия исключены [7, 8].

Здесь используются параллельные подстановки двух видов: символьные и функциональные. Шаблоном подстановки является квадрат 2×2 клетки. Символьные подстановки $\theta_1, \dots, \theta_{12}$ (рис. 1) в левой части содержат состояния четырех клеток, в правой – новые состояния базовых клеток и черточки на месте клеток контекста. Левая часть функциональной подстановки $\theta_{13}, \dots, \theta_{21}$ (см. рис. 1) задается одним из шаблонов P_1, \dots, P_4 , в каждом из которых записаны четыре переменные, правая часть определяется одной из функций f_1, \dots, f_6 от этих переменных.

АПП для самовоспроизведения в двух направлениях строится на основе алгоритма линейного реплицирования [6]. Клетки имеют 8 состояний, 6 из них те же, что в [6] (0 – состояние покоя, 1 – элемент вспомогательной внутренней оболочки, 2 – элемент регулярной внутренней оболочки, 3 – сигнал

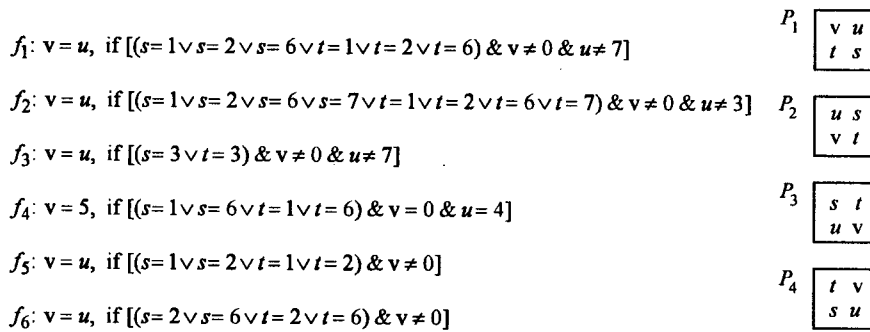
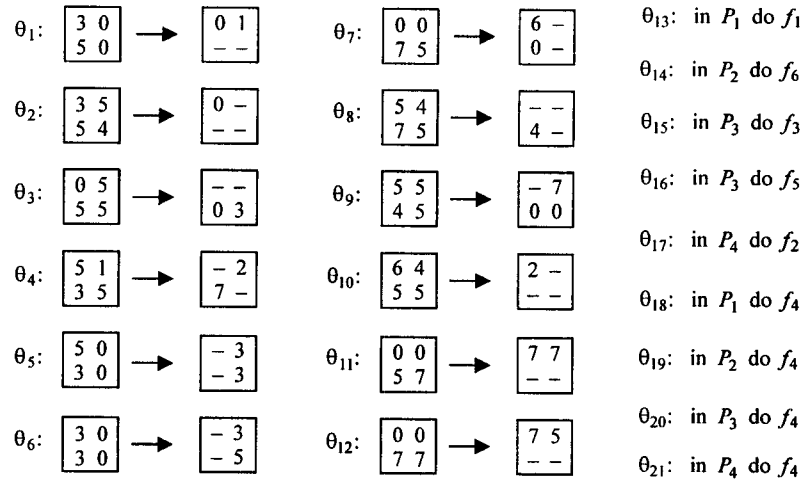


Рис. 1

ветвления и элемент регулярной внешней оболочки, 4 – сигнал роста, 5 – строительная компонента) и 2 дополнительных состояния (6 – элемент вспомогательной внутренней оболочки, которая в отличие от 1, используется при построении копий слева направо, применяется при росте петли снизу вверх, 7 – элемент регулярной внешней оболочки, таким же образом отличающийся от 3, как 6 от 1).

Исходной самовоспроизводящейся структурой является петля, образованная квадратом 3×3 клетки, в центре него имеется внутренняя оболочка, которая при данном размере петли образуется только одной клеткой. Вокруг внутренней оболочки по 8 клеткам против часовой стрелки циркулирует последовательность состояний, представляющая собою самописание петли; в этой последовательности присутствуют два сигнала роста, что соответствует размеру стороны петли. Петля помещается в достаточное для построения определенного количества копий двумерное клеточное пространство, все клетки которого, кроме образующих петлю, находятся в состоянии покоя.

Петля, копии которой выстраиваются в линию, имеет одну конструирующую руку в своем правом нижнем углу [6]. В нашем случае копии должны расти в двух направлениях, поэтому петля имеет две конструирующие руки (рис. 2, позиция 0). Одна справа внизу образуется двумя строительными компонентами (состояние 5) и двумя элементами регулярной внешней обо-

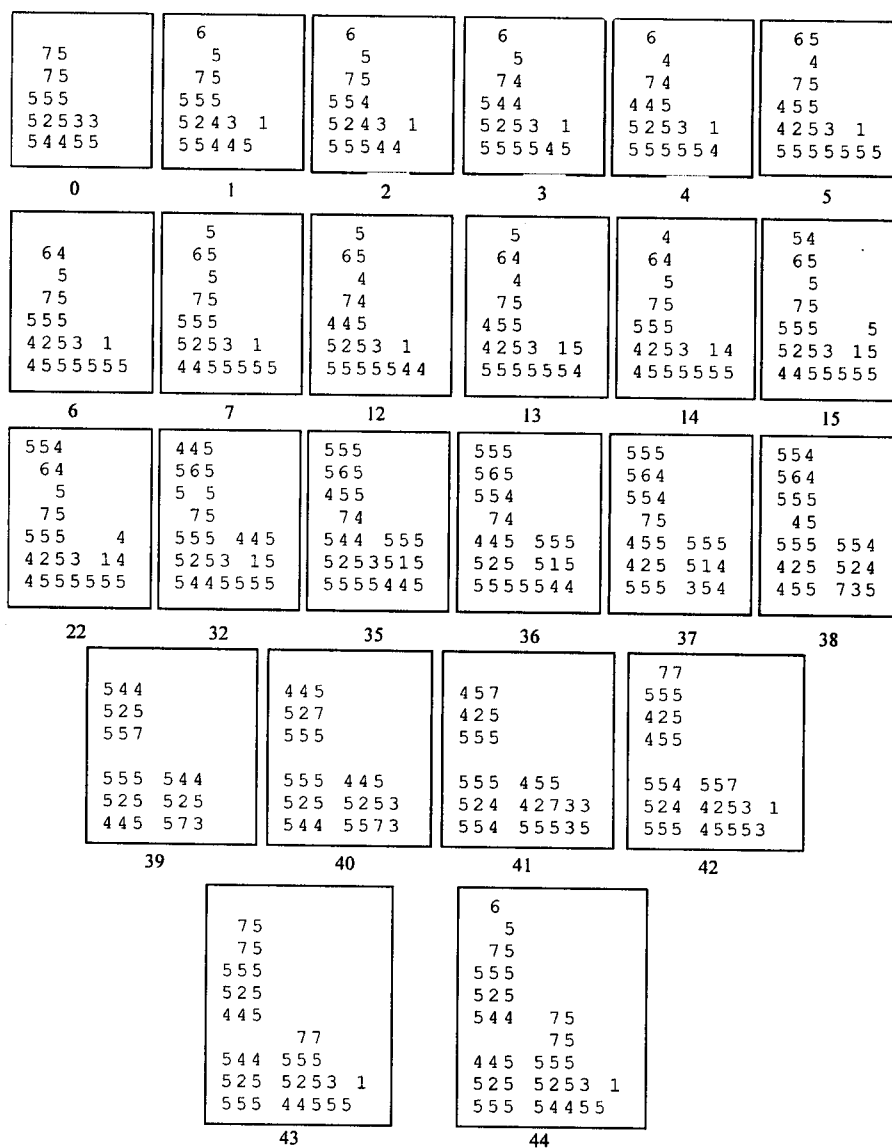


Рис. 2

лочки (состояние 3), другая в правом верхнем углу петли также образуется двумя строительными компонентами и двумя элементами регулярной внешней оболочки (состояние 7). Заметим, что в двух направлениях копируются петли только самого нижнего ряда, в остальных рядах петли растут снизу вверх. Для этих рядов горизонтальное направление копирования не является необходимым, так как другие новые петли добираются сюда первыми. Это значит, что фронт роста петель движется по диагонали квадрата, как показано на рис. 3.

Новый АПП имеет 21 подстановку в отличие от 18 в [6], из них 12 символьных и 9 функциональных (см. рис. 1). Подстановки $\theta_1, \dots, \theta_6$ перенесены

из [6], $\theta_7, \dots, \theta_{12}$ отличаются от первых шести поворотом на 90° . Построение копии начинается с того, что в центр новой пока еще воображаемой петли ставится элемент вспомогательной внутренней оболочки, это – состояние 1 справа от оригинала, что выполняется подстановкой θ_1 , и состояние 6 сверху, за что отвечает θ_7 (см. рис. 2, позиция 1). Далее информация, циркулирующая в петле, удваивается и копия посылается по конструирующей руке, чтобы по окончании процесса копирования быть обернутой вокруг новой оболочки (см. рис. 2, позиции 1–3). Копирование происходит путем вытягивания конструирующей руки, которая через две новые клетки делает поворот на 90° влево, после трех поворотов петля замыкается, и конструирующая рука в виде мостика между оригиналом и копией разрушается. Подстановки θ_2 и θ_3 и соответственно θ_8 и θ_9 отвечают за разрушение горизонтальных и вертикальных мостиков между старой и новой петлями и за введение сигналов (состояния 3 и 7 для горизонтальной и вертикальной копий соответственно) для рождения новых конструирующих рук: одной в горизонтальном, а другой в вертикальном направлениях. Подстановки θ_4 и θ_{10} заменяют вспомогательные состояния 1 и 6 регулярным 2 в законченных петлях. Подстановки θ_5 и θ_6 строят горизонтальную конструирующую руку у новой петли нижнего ряда, θ_{11} и θ_{12} – вертикальную конструирующую руку у всех новых петель.

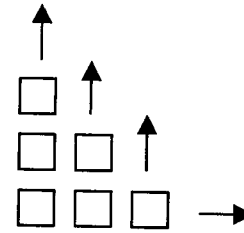


Рис. 3

Подстановки $\theta_{13}, \theta_{14}, \theta_{16}, \theta_{17}$ отвечают за движение информации в петле, θ_{15} отвечает за движение информации по горизонтальному мостику, за движение данных по вертикальному мостику вместе с движением по правому ребру петли отвечает θ_{17} . И наконец, подстановки $\theta_{18}, \dots, \theta_{21}$ осуществляют рост новой петли. Процесс самовоспроизведения петли по описанному алгоритму параллельных подстановок показан на рис. 2, состояние покоя обозначено пустой клеткой. Алгоритм параллельных подстановок для самовоспроизведения в двух направлениях содержит 9 укрупненных функциональных подстановок в отличие от 12 в работе [6], которые выполняют всю работу 12 подстановок.

Заключение. Построение лаконичных алгоритмов самовоспроизведения не является самоцелью. Дело в том, что появление простой самовоспроизводящейся структуры – петли Лангтона – инициировало исследования по созданию «полезного ипликатора», который вместе с построением копий выполняет некоторую вычислительную программу. Например, сделаны попытки использования петли для решения задач комбинаторной оптимизации [9] и выполнения арифметических операций [10]. Использование самовоспроизводящейся петли создает новые возможности для построения клеточных алгоритмов и архитектур. Очевидно, что собственно построение самовоспроизводящихся структур является одной из важных тем в разработке приложений, и поскольку такие структуры наделяются дополнительными функциями, полезно программу самовоспроизведения уметь представлять в компактной форме.

СПИСОК ЛИТЕРАТУРЫ

1. Von Neumann J. Theory of Self-replication Automata /Ed. A. W. Burks. Urbana: University of Illinois Press, 1966.

2. **Codd E. F.** Cellular Automata. New York: Academic Press, 1968.
3. **Langton C. G.** Self-replication in cellular automata // Physica D. 1984. **10**. P. 135.
4. **Byl J.** Self-reproduction in small cellular automata // Physica D. 1989. **34**. P. 295.
5. **Reggia J. A., Armentrout S. I., Chou H.-H., Peng Y.** Simple systems that exhibit self-directed replication // Science. 1993. **259**. P. 1282.
6. **Ачасова С. М.** Моделирование самовоспроизведения в клеточном пространстве на основе алгоритма параллельных подстановок // Автометрия. 2002. **38**, № 4. С. 87.
7. **Ачасова С. М., Бандман О. Л.** Корректность параллельных вычислительных процессов. Новосибирск: Наука, 1990.
8. **Achasova S. M., Bandman O. L., Markova V. P., Piskunov S. V.** Parallel Substitution Algorithm. Theory and Application. Singapore: World Scientific, 1994.
9. **Chou H.-H., Reggia J. A.** Problem solving during artificial selection of self-replicating loops // Physica D. 1998. **115**. P. 293.
10. **Petraglio E., Henry J.-M., Tempesti G.** Arithmetic operations on self-replicating cellular automata // Lecture Notes in Artificial Intelligence. 1999. **1674**. P. 447.

*Институт вычислительной математики
и математической геофизики СО РАН,
E-mail: achasova@ssd.ssc.ru*

*Поступила в редакцию
9 апреля 2003 г.*