

©Башкин В. А., 2017

DOI: 10.18255/1818-1015-2017-4-391-409

УДК 519.7

## О пространственной ограниченности клеточных P-сетей

Башкин В. А.

получена 21 июля 2017

**Аннотация.** Клеточные P-сети — обобщение концепции двухуровневых ресурсных сетей (сетей Петри) на случай бесконечной регулярной системной решетки. Этот формализм представляет собой гибрид сетей Петри и асинхронных клеточных автоматов и предназначен для моделирования мультиагентных систем с динамической пространственной структурой. Пространственная ограниченность — свойство, гарантирующее сохранение конечности “геометрических размеров” (например, площади) активной части системы на протяжении всей её жизни. Определяются три варианта пространственной ограниченности для клеточных P-сетей: локализованность, ограниченность диаметра и ограниченность площади. Исследуются свойства соответствующих алгоритмических проблем, доказываются их неразрешимость в общем случае. Предлагается нетривиальный критерий локализованности одномерной клеточной сети, основанный на новой концепции графа распространения P-автоматов. Описывается алгоритм построения графа распространения, использующий метод насыщения генерирующих путей. Предлагается способ оценки сверху диаметра одномерной клеточной сети с ограниченным графом распространения.

**Ключевые слова:** мультиагентные системы, верификация, сети Петри, клеточные автоматы, P-сети, пространственная ограниченность

**Для цитирования:** Башкин В. А., "О пространственной ограниченности клеточных P-сетей", *Моделирование и анализ информационных систем*, **24:4** (2017), 391–409.

**Об авторах:**

Башкин Владимир Анатольевич, [orcid.org/0000-0002-2534-1026](https://orcid.org/0000-0002-2534-1026), д-р физ.-мат. наук, доцент,  
Ярославский государственный университет им. П.Г. Демидова,  
ул. Советская, 14, г. Ярославль, 150003 Россия, e-mail: [v\\_bashkin@mail.ru](mailto:v_bashkin@mail.ru)

**Благодарности:**

Работа выполнена при финансовой поддержке РФФИ (проект 17-07-00823).

## Введение

Обыкновенные сети Петри представляют собой низкоуровневый формализм с очень простым набором основных элементов: позиция, переход, дуга и фишка. Отсутствуют сколько-нибудь удобные инструменты для высокоуровневых конструкций, таких как модуль и иерархия. Также обыкновенные сети Петри не вполне удобны для моделирования мультиагентных систем с динамической структурой, поскольку в них невозможно изменять в ходе функционирования сети структуру множества переходов. Моделировать возникновение новых агентов и исчезновение старых приходится

изменением разметки вершин-позиций, тем самым разрешая или запрещая срабатывания зависящих от них переходов.

Существует ряд формализмов, построенных на основе обыкновенных сетей Петри, в которых тем или иным способом более явно выделяется понятие агента, а также вносятся конструкции модульности и иерархичности. В частности, в *сетях Петри высокого уровня* (например, в раскрашенных сетях [14], объектных сетях [19] и многих других [11, 15]) вводятся охранные функции на переходах и выражения на дугах, позволяющие усложнить условие срабатывания перехода и производимое им действие. Во вложенных сетях Петри [16] усложняется структура ресурса: он сам может быть сетью Петри (вплоть до рекурсивности).

В работе [8] был определен обладающий двухуровневой структурой визуальный язык сетей автоматов с ресурсами (Р-сетей), позволяющий описывать мультиагентные системы компактно и в то же время достаточно наглядно. Предложены и исследованы новые методы моделирования и анализа систем, представленных в виде сетей с ресурсами (АР-сетей, Р-сетей, клеточных Р-сетей). Показано, что модульные АР-сети и вложенные сети обладают рядом конструктивных композиционных свойств (в частности, существуют специальные виды наследуемости важнейших свойств ограниченности и живости).

В работе [9] было определено и исследовано расширение концепции Р-сети на случай моделей с бесконечной системной сетью. В связи с близостью по концепции к клеточным автоматам такие системы были названы клеточными сетями. В статье [9] была построена иерархия классов одномерных клеточных сетей (цепочек), основанная на ограничении топологии системной сети. Исследована выразительная мощность ряда базовых классов данной иерархии. Доказано, что: сети с полным набором портов эквивалентны машинам Тьюринга; сети без выходных портов эквивалентны конечным автоматам; сети без входных портов бисимулярны сетям Петри без коммуникаций (communication-free PN). Было продемонстрировано, что такие формализмы пригодны для моделирования и верификации распределенных систем агентов на бесконечной решетке (в частности, пространственно распределенных беспроводных сетей).

Ещё один вариант бесконечных сетей Петри был предложен Д. А. Зайцевым в работе [3]. Однако там рассматриваются обыкновенные одноуровневые сети Петри, и структура решетки сформирована на том же уровне, что и структура поведения отдельного узла. В работе [3] показано, что подобные сети могут быть использованы для моделирования и верификации статичных вычислительных решеток.

Одной из проблем, возникающих при анализе потенциально бесконечных динамических “перемещающихся в пространстве” клеточных моделей, является выяснение самой возможности таких бесконечных “перемещений”. Например, для агентов-сенсоров обход всей решетки может быть желательным свойством. Напротив, для агентов-базовых станций желательно сохранение постоянного расстояния до станций-соседей (то есть запрет превышения максимального диаметра сети). Кроме того, пространственная ограниченность позволяет рассчитывать на разрешимость многих интересных семантических свойств моделируемых систем, поскольку, как уже было установлено ранее [9], подобные “слабые” клеточные сети слабее машин Тьюринга.

В данной работе исследуются возможности построения конструктивных критериев пространственной ограниченности для существенных подклассов клеточных сетей Петри. При этом рассматриваются различные варианты пространственной ограниченности: от “конечности множества когда-либо населенных ячеек” до более слабой “ограниченности диаметра населенной области”.

Основной результат работы — критерий локализованности клеточной сети, основанный на построении так называемого графа распространения. Соответствующий алгоритм использует метод насыщения, близкий к методу построения покрывающего дерева обыкновенной сети Петри (например, [4]). Однако в данном случае мы насыщаем не конечномерный вектор разметки, а так называемые генерирующие пути в графе распространения.

В работе рассматриваются одномерные клеточные Р-сети (системная решетка представляет собой цепочку ячеек), однако все предложенные методы могут быть легко обобщены на более высокие размерности (например, клеточные модели на плоскости и в трехмерном пространстве).

Статья организована следующим образом. В первой главе приводятся основные определения и обозначения, касающиеся мультимножеств, Р-сетей и клеточных Р-сетей. В частности, приводятся основные свойства выразительности для соответствующих классов формальных моделей. Вторая глава посвящена проблемам пространственной ограниченности одномерных клеточных Р-сетей. Вводятся определения, доказывається неразрешимость в общем случае, приводится критерий локализованности на основе графа распространения. Заключение содержит некоторые выводы и направления дальнейших исследований.

## 1. Предварительные сведения

### 1.1. Мультимножества

Пусть  $S$  — конечное множество. *Мультимножеством*  $M$  над множеством  $S$  называется отображение  $M : S \rightarrow Nat$ , где  $Nat$  — множество неотрицательных целых чисел. Обозначим через  $\mathcal{M}(S)$  множество всех конечных мультимножеств над  $S$ .

Операции и отношения теории множеств естественно расширяются на конечные мультимножества. Пусть  $M_1, M_2, M_3 \in \mathcal{M}(S)$ . Полагаем:  $M_1 \subseteq M_2 \Leftrightarrow \forall s \in S : M_1(s) \leq M_2(s)$ ;  $M_1 = M_2 + M_3 \Leftrightarrow \forall s \in S : M_1(s) = M_2(s) + M_3(s)$ ;  $M_1 = M_2 \cup M_3 \Leftrightarrow \forall s \in S : M_1(s) = \max\{M_2(s), M_3(s)\}$ .

### 1.2. Сети автоматов, управляемых ресурсами (Р-сети)

Мы определяем Р-сеть как АР-систему [1, 2], в которой в роли фишек выступают специальные расширенные конечные автоматы — так называемые автоматы, управляемые ресурсами (Р-автоматы). Таким образом, мы расширяем понятие фишки (ресурса) путем добавления ей собственного поведения. В ходе своей работы автоматная фишка потребляет и производит фишки-ресурсы (в том числе автоматные), находящиеся в узлах, соседних тому, в котором она сама находится. Дуги в АР-сети мы помечаем именами портов, указывая тем самым узлы с доступными ресурсами.

Пусть  $\Omega$  — конечное множество *типов*,  $Const$  — конечное множество типизированных индивидуальных объектов, которые мы называем константами. Тип константы  $c \in Const$  обозначается как  $Type(c)$ . Для  $a \in \Omega$  через  $Const(a)$  мы обозначаем множество всех констант типа  $a$ .

Далее, пусть  $\Pi$  — конечное множество типизированных (элементами  $\Omega$ ) *портов* (имён портов). Тип порта  $\pi \in \Pi$  обозначим  $Type(\pi)$ .

Лежащая в основе Р-сети сеть активных ресурсов с дугами, помеченными именами портов, называется *системной сетью*.

**Определение 1.** Системная сеть — это набор  $SN = (V, I, O, \pi)$ , где

- $V$  — конечное множество вершин (ресурсов);
- $I : V \times V \rightarrow Nat$  — множество потребляющих дуг;
- $O : V \times V \rightarrow Nat$  — множество производящих дуг;
- $\pi : (I \cup O) \rightarrow \Pi$  — функция, помечающая дуги именами портов.

Заметим, что фактически набор  $(V, I, O)$  представляет собой структуру сети активных ресурсов [2]: типы объектов (активных ресурсов) и способы их взаимодействия (потребление и производство). Графически вершины сети изображаются кружками, потребляющие дуги — пунктирными стрелками, производящие дуги — непрерывными стрелками. Потребляющая дуга  $(u, v)$  означает, что объекты в вершине  $v$  могут (выступая в качестве активных агентов) потреблять объекты из вершины  $u$  (выступающие в качестве пассивных ресурсов). Производящая дуга  $(u, v)$  означает, что объекты в вершине  $u$  могут (выступая в качестве активных агентов) производить объекты в вершине  $v$  (выступающие в качестве пассивных ресурсов). При этом один и тот же объект может в рамках различных взаимодействий (определяемых графом системной сети) выступать в любой из 4 ролей: потребляющего агента, потребляемого ресурса, производящего агента и производимого ресурса (в том числе сразу в нескольких ролях одновременно).

В качестве объектов выступают модифицированные конечные автоматы, переходы которых дополнительно к смене внутреннего состояния самого объекта преобразуют ресурсы в инцидентных вершинах системной сети (согласно особым приписанным им ресурсным выражениям).

**Определение 2.** Разметкой  $M$  системной сети  $SN$  называется функция

$$M : V \rightarrow \mathcal{M}(Const),$$

сопоставляющая каждому узлу сети мультимножество находящихся в нём объектов.

Размеченной системной сетью называется пара  $(SN, M_0)$ , где  $SN = (V, I, O, \pi)$  — системная сеть, а  $M_0$  — её начальная разметка.

Обозначим через  $Var$  множество типизированных переменных с типами из  $\Omega$ . Пусть  $a \in \Omega$ . Определим язык  $L(a)$  *ресурсных выражений* типа  $a$  следующим образом.

Пусть  $\pi \in \Pi$ . *Входной терм* есть терм вида  $\pi?e$ , где  $e$  — переменная или константа типа  $Type(\pi)$ . Аналогично, *выходной терм* есть терм вида  $\pi!e$  с теми же условиями на  $\pi$  и  $e$ . *Ресурсное выражение* есть терм вида  $\alpha_1; \alpha_2; \dots; \alpha_k$ , где  $\alpha_j$  ( $j = 1, \dots, k$ ) представляет собой входной или выходной терм (типы подвыражений  $\alpha_1, \alpha_2, \dots, \alpha_k$  могут различаться).

**Определение 3.** Управляемым ресурсами автоматом типа  $a \in \Omega$  (*ресурсным автоматом, Р-автоматом*) называется набор  $A = (S_A, T_A, l_A)$ , где  $S_A$  — конечное множество состояний,  $T_A \subseteq S_A \times S_A$  — отношение перехода и  $l_A : T_A \rightarrow L(a)$  — функция пометки переходов.

Таким образом, Р-автомат — это всего лишь конечный автомат с помеченными переходами и без выделенного начального состояния. В Р-сетях Р-автоматы будут играть роль фишек сетей Петри. В вырожденном случае, когда Р-автомат содержит одно состояние и не содержит переходов, такие фишки представляют собой обычные цветные фишки, которые мы будем называть *элементарными ресурсами*.

**Определение 4.** Пусть  $\Omega = \{a_1, \dots, a_k\}$  — конечное множество типов,  $A = (A_1, \dots, A_k)$  — конечное множество Р-автоматов, такое, что

- 1) для типа  $a_i \in \Omega$  множество  $Const(a_i)$  определяется как множество всех состояний Р-автомата  $A_i$ ;  $Const =_{def} \bigcup_{a \in \Omega} Const(a)$ ;
- 2) каждый  $A_i$  есть Р-автомат типа  $a_i$  над множеством типов  $\Omega$  и множествами  $\Omega$ -типизированных переменных  $Var$ , имён портов  $\Pi$  и констант  $Const$ .

Сеть управляемых ресурсами автоматов (*Р-сеть*)  $RN = (\Omega, SN, (A_1, \dots, A_k))$  состоит из описанного выше конечного множества Р-автоматов  $(A_1, \dots, A_k)$  и системной сети  $SN$  над множеством типов  $\Omega$  и множествами  $\Omega$ -типизированных имён портов  $\Pi$  и констант  $Const$ .

*Разметкой (состоянием) Р-сети* называется разметка лежащей в её основе системной сети.

Допуская некоторую вольность в обозначениях, условимся не различать автомат  $A$  и его имя/тип  $a$ . Далее мы будем писать  $(a|s)$  для обозначения константы, соответствующей состоянию  $s$  автомата  $A$  типа  $a$ . В зависимости от контекста такая константа может называться *агентом, ресурсом* или просто *объектом*.

Определим интерливинговую семантику для Р-сетей. Срабатыванием Р-сети является последовательность срабатываний переходов её агентов. Таким образом, только агенты могут менять состояние системы.

Для срабатывания перехода  $t$  в агенте  $a$  требуются ресурсы, перечисленные во входных подтермах помечающего  $t$  ресурсного выражения. Входной терм  $p?e$  описывает ресурс  $e$ , который должен быть получен через порт  $p$  системной сети (то есть этот ресурс должен находиться в том узле системной сети, из которого ведёт дуга порта  $p$ ). Аналогично, выходные термы определяют ресурсы, производимые агентом (и узлы-получатели для этих ресурсов).

Режимы срабатывания перехода определяются возможными означиваниями переменных.

**Определение 5.** Означиванием (переменных) называется функция

$$\mathbf{bind} : Var \rightarrow Const,$$

такая, что для любой переменной  $\varphi \in Var$  выполняется  $Type(\mathbf{bind}(\varphi)) = Type(\varphi)$ .

Пусть  $M$  — разметка Р-сети  $RN = (\Omega, SN, (A_1, \dots, A_k))$ ,  $a|s$  — Р-автомат в состоянии  $s$ , находящийся в узле  $v$  сети  $RN$  при разметке  $M$ ,  $t$  — переход в  $a|s$ . Пусть  $b$  — означивание переменных. Переход  $t = (s, s')$  с меткой  $l_a(t)$  активен при разметке  $M$ , если существует взаимно однозначное соответствие между входными подтермами в  $l_a(t)$  и объектами (фишками) в  $M$ , такое, что для каждого подтерма  $p?e$  существует объект  $e[b]$  в узле  $v'$ , соединенном с узлом  $v$  входной дугой, помеченной именем порта  $p$ .

Описанное выше соответствие определяет “подразметку”  $\check{M}$  системной разметки, описываемую входными подтермами выражения  $l_a(t)$ . Она включает все объекты, потребляемые срабатыванием перехода  $t$  при означивании  $b$ . Заметим, что для данных  $t$  и  $b$  разметка  $\check{M}$  определяется недетерминированно (может быть несколько вариантов разметок, удовлетворяющих условиям). Через  $\mathbf{in}(t[b])$  мы обозначим множество всех таких разметок.

Аналогично, определим множество  $\mathbf{out}(t[b])$  разметок, сопоставляющих узлам объекты, производимые переходом  $t$  при означивании  $b$  в соответствие с выходными подтермами  $l_a(t)$ .

**Определение 6.** Пусть  $(a|s)$  — агент, находящийся в узле  $v \in V$  при разметке  $M$ ,  $t \in T_a$  — переход, такой, что  $t = (s, s')$  для некоторого  $s'$ .

Переход  $t$  активен с означиванием переменных  $b$ , если существует разметка  $\check{M} \in \mathbf{in}(t[b])$ , такая, что  $\forall u \in V : \check{M}(u) \subseteq M(u)$ .

Активный означенный переход может (недетерминированно) сработать, преобразуя  $M$  в новую разметку  $M'$ , такую, что  $\forall u \in V : M'(u) = M(u) - \check{M}(u) + \hat{M}$ , где  $\hat{M} \in \mathbf{in}(t[b])$  и  $\hat{M} \in \mathbf{out}(t[b])$ .

Сети управляемых ресурсами автоматов позволяют моделировать различные аспекты ресурсно-зависимых систем. Рассмотрим простейший классический пример — химическую реакцию (Рис. 1). Единственный активный агент данной системы — автомат  $R$  — определяет сам алгоритм реакции (“две молекулы водорода и одна молекула кислорода превращаются в две молекулы воды”). Системная сеть описывает физическую среду: три резервуара и трубу, в которой происходит реакция. Порты *get* и *put* связывают эти два уровня (или два аспекта) модели. С системной точки зрения порты — это физические “отверстия”, с точки зрения логики агента — это имена аргументов (параметров).

Несмотря на новые возможности моделирования, Р-сети обладают теми же возможностями анализа, что и сети Петри [18]:

**Теорема 1.** [8] Класс систем, моделируемых Р-сетями, совпадает с классом систем, моделируемых сетями Петри.

Сети ресурсных автоматов моделируют явное и неявное взаимодействие агентов без введения каких-либо промежуточных слоёв и/или протоколов. Это позволяет

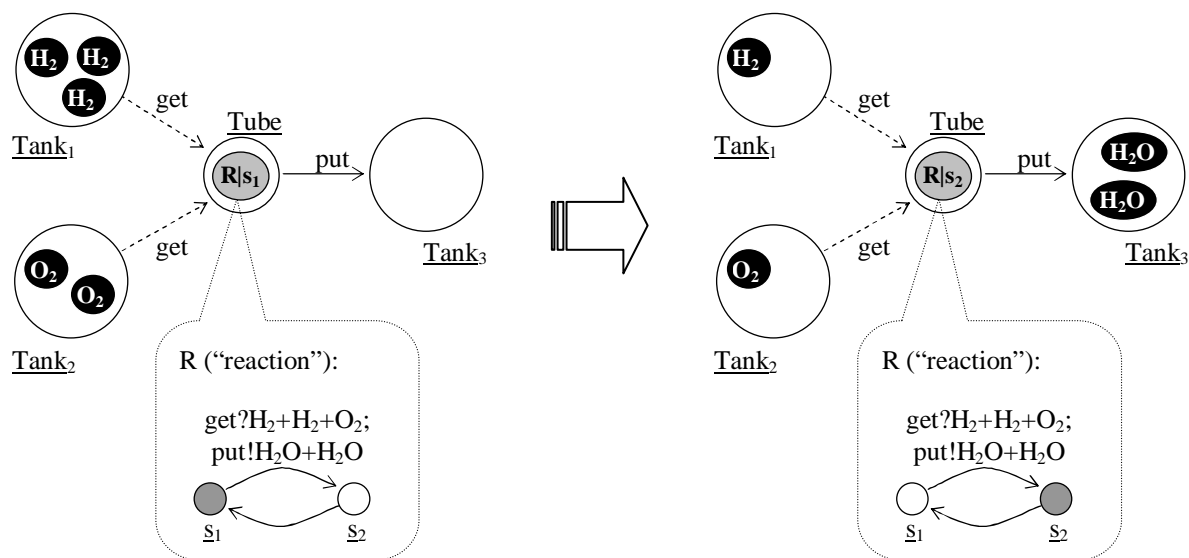


Рис. 1. Пример Р-сети: химическая реакция

Fig. 1. RDA-net example: chemical reaction

достаточно адекватно отражать специфику предметной области при помощи единообразного и компактного синтаксиса. Агенты могут быть перемещены, созданы, скопированы и уничтожены другими агентами (внешняя мобильность). Они также могут выполнять все эти действия (над собой) по собственной инициативе (внутренняя мобильность).

Заметим, что выбор в качестве фишек автоматов, а не сетей Петри, несколько не снижает выразительности формализма по сравнению с обыкновенными и высокоуровневыми сетями Петри (Теорема 1). При этом синтаксис языка получается максимально простым, что позволяет достаточно легко строить различные гибридные и высокоуровневые расширения [10].

### 1.3. Клеточные Р-сети

Одной из интересных сторон Р-сетей является возможность удобного моделирования пространственной динамики – агенты могут перемещать друг друга между узлами системной сети (или перемещаться по ней самостоятельно). Естественен переход к бесконечному случаю:

**Определение 7.** [9] *Клеточная Р-сеть – конечная совокупность ресурсных автоматов, расположенных в узлах бесконечной регулярной системной сети (решётки).*

Клеточные Р-сети (Celluar Resource Driven Automata, CRDA) – модель, подобная клеточным автоматам, где в отдельной ячейке может находиться не автомат (система с конечным числом состояний), а Р-сеть (сеть Петри).

Основные **отличия** клеточных Р-сетей от обыкновенных Р-сетей (классических сетей Петри):

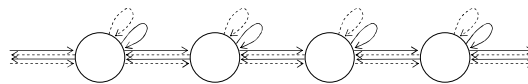
- неограниченное число отдельных сетей (Р-автоматов);
- гибкое взаимодействие между ними, возможность создавать и уничтожать автоматы.

Основные **отличия** клеточных Р-сетей от классических **синхронных клеточных автоматов**:

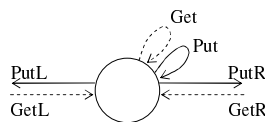
- асинхронность срабатываний ячеек, недетерминированность выбора активной ячейки;
- отдельная ячейка — не автомат, а сеть Петри, то есть система с неограниченным числом состояний.

Заметим, что клеточные автоматы универсально мощны даже в одномерном случае [12], где решетка представляет собой цепочку клеток. Поэтому мы также будем рассматривать только одномерные сети.

Рассматриваемая (бесконечная регулярная одномерная) системная сеть изображена на Рис. 2. Активные фишки (Р-автоматы), находящиеся в клетке, могут потреблять/производить ресурсы из трёх клеток — левой соседней, правой соседней и текущей. Ограничений на фишки нет.



а) общий вид решетки



б) порты отдельной ячейки

Рис. 2. Решетка одномерной клеточной Р-сети (1-dim CRDA)

Fig. 2. System grid of 1-dim CRDA

**Теорема 2.** [9] *Класс систем, моделируемых одномерными клеточными Р-сетями (1-dim CRDA), совпадает с классом систем, моделируемых машинами Тьюринга.*

## 2. Пространственная ограниченность

### 2.1. Неразрешимость

В доказательстве Теоремы 2 в работе [9] использовалось преобразование произвольной машины Тьюринга, представленной в виде двухсчётчикового автомата  $M$



(машины Минского), в одномерную клеточную сеть  $N(M)$  с не более чем двумя автоматами в каждой ячейке в любой момент времени. Значения первого и второго счетчиков моделировались количеством “живых” клеток, соответственно слева и справа от центральной ячейки. Конечно, “длина” живой части решетки в такой системе может расти неограниченно.

Очевидно, что такая пространственная неограниченность является необходимым условием универсальности. Если её нет, то система представляет собой обычную, а не клеточную Р-сеть, которая является всего лишь сетью Петри (Теорема 1).

Пространственная ограниченность является нетривиальным семантическим свойством, которое требует для своей проверки анализа поведения системы (в общем случае бесконечного).

Сформулируем некоторые возможные варианты определения пространственной ограниченности:

**Определение 8.** (*Пространственная ограниченность*)

- (1) Сеть локализована, если ни один из автоматов никогда не появляется за пределами некоторого конечного множества ячеек  $D$ .
- (2) Сеть имеет ограниченный диаметр, если в каждый момент времени наибольшее расстояние между “населенными” ячейками (число разделяющих их ячеек) не превосходит некоторого натурального  $n$ .
- (3) Сеть имеет ограниченную площадь, если в каждый момент времени число “населенных” ячеек не превосходит некоторого натурального  $n$ .

Непосредственно из определений следует:

**Утверждение 1.** “Локализованность”  $\Rightarrow$  “Ограниченный диаметр”  $\Rightarrow$  “Ограниченная площадь”.

В то же время легко придумать контрпримеры для  $(2) \Rightarrow (1)$  и  $(3) \Rightarrow (2)$ .

По построению сети  $N(M)$  в доказательстве Теоремы 2 (см. [9]) мы имеем:

**Утверждение 2.** Если некоторая 2-счетчиковая машина Минского  $M$  ограничена (оба её счетчика ограничены), то соответствующая 1-dim CRDA  $N(M)$  локализована.

Тогда из неразрешимости ограниченности для 2-счетчиковых машин Минского [5] имеем:

**Следствие 1.** Локализованность неразрешима для 1-dim CRDA.

Поскольку свойства ограниченности диаметра и площади ещё слабее (Утверждение 1), имеем:

**Следствие 2.** Проблемы “ограниченности диаметра” и “ограниченности площади” неразрешимы для 1-dim CRDA.

Очевидно, это означает, что пространственная ограниченность (во всех трёх вариантах) неразрешима и при более высоких размерностях системной решетки в случае её нетривиальной топологии. Однако, как будет показано далее, использование свойств монотонности сетей Петри позволяет построить достаточно широкие критерии пространственной ограниченности.

## 2.2. Обозначения для одномерной решетки

Для удобства работы с одномерными клеточными сетями введём ряд новых терминов и обозначений.

Во-первых, будем считать, что все ячейки пронумерованы слева направо целыми числами. За условную точку отсчета примем нулевую ячейку.

Во-вторых, в отличие от [8, 9] мы будем использовать ресурсные выражения без переменных (только константы). Поскольку в каждом типе число констант конечно, это не снижает выразительной мощности модели. В то же время подобное упрощение синтаксиса позволяет более компактно описывать и анализировать бесконечную цепочку ячеек.

**Определение 9.** Пусть  $a$  — некоторый  $P$ -автомат,  $s$  — его состояние,  $M$  — системная разметка. Обозначим через  $M[i](a|s)$  количество автоматов  $a|s$  в ячейке одномерной системной сети (цепочки ячеек) с номером  $i$  при разметке  $M$  (ячейки пронумерованы слева направо последовательными значениями из  $\mathbb{Z}$ ).

Тогда состояние ячейки с номером  $i$  при разметке  $M$  представляет собой мультимножество над  $Const$ :

$$M[i] =_{def} \sum_{a \in \Omega, s \in S_a} M[i](a|s),$$

или, что то же самое,  $M[i] = \sum_{c \in Const} M[i](c)$ .

Разметка всей системной сети-цепочки представляет собой бесконечномерный вектор вида  $M = (\dots, M[i], M[i+1], \dots)$ , где  $M[i] \in \mathcal{M}(Const)$ ,  $i \in \mathbb{Z}$ .

Счетное множество всех возможных разметок обозначим как  $\mathcal{M}^\infty(Const)$ .

**Определение 10.** Пусть  $M, M' \in \mathcal{M}^\infty(Const)$ . Тогда  $\forall i \in \mathbb{Z}, c \in Const$ :

$$(M + M')[i](c) =_{def} M[i](c) + M'[i](c),$$

$$(M \cup M')[i](c) =_{def} \max\{M[i](c), M'[i](c)\},$$

$$(M \cap M')[i](c) =_{def} \min\{M[i](c), M'[i](c)\}.$$

Скажем, что  $M \subseteq M'$ , если  $\exists k \in \mathbb{Z} : \forall i \in \mathbb{Z}, c \in Const$  выполняется  $M[i](c) \leq M'[i+k](c)$ .

Таким образом, при сравнении мы допускаем сдвиг нумерации ячеек влево или вправо на произвольную величину. Это естественно, поскольку все связи локальны, а ячейки имеют идентичные свойства, не зависящие от номера.

Набор доступных портов один и тот же для каждой ячейки:

$$\Pi = \{Get, Put, GetL, PutL, GetR, PutR\}.$$

Следовательно, для любого перехода  $t \in T_a$  его ресурсное выражение  $l(t)$  распадается на шесть подвыражений:  $l_{Get}(t), l_{Put}(t), l_{GetL}(t), l_{PutL}(t), l_{GetR}(t)$  и  $l_{PutR}(t)$  соответственно. Заметим, что каждое из этих выражений является мультимножеством констант.

Также заметим, что в силу однозначности именования портов множества  $\mathbf{in}(t)$  и  $\mathbf{out}(t)$  всегда содержат ровно по одному элементу. Обозначим эти элементы как  $\bullet t$  и  $t^\bullet$  и назовём пред- и постусловием перехода  $t$  соответственно. Имеем  $\bullet t = (l_{Get}(t), l_{GetL}(t), l_{GetR}(t))$ ,  $t^\bullet = (l_{Put}(t), l_{PutL}(t), l_{PutR}(t))$ .

**Определение 11.** Пусть автомат  $a|s$  находится в ячейке с номером  $i$  при разметке  $M$  (т.е.  $M[i](a|s) > 0$ ). Пусть также  $t \in T_a$  — переход автомата  $a$ , такой, что  $a|s \xrightarrow{t} a|s'$  для некоторого  $s' \in S_a$ . Переход  $t$  активен, если выполняется

- $l_{GetL}(t) \subseteq M[i - 1]$ ;
- $l_{Get}(t) \subseteq M[i]$ ;
- $l_{GetR}(t) \subseteq M[i + 1]$ .

Активный переход  $t$  может сработать, порождая новую разметку  $M'$  :

- $M'[i - 1] = M[i - 1] - l_{GetL}(t) + l_{PutL}(t)$ ;
  - $M'[i]$  :
    - $M'[i](c) = M[i](c) - l_{Get}(t)(c) + l_{Put}(t)(c)$  для всех  $c \neq a|s$ ;
    - при  $M[i](a|s) = l_{Get}(t)(a|s)$  положим  $M'[i](a|s) = M[i](a|s) - l_{Get}(t)(a|s) + l_{Put}(t)(a|s)$ ;
    - при  $M[i](a|s) > l_{Get}(t)(a|s)$  недетерминированно выберем один из двух вариантов:  $M'[i](a|s) = M[i](a|s) - l_{Get}(t)(a|s) + l_{Put}(t)(a|s)$  или  $M'[i](a|s) = M[i](a|s) - (a|s) + (a|s') - l_{Get}(t)(a|s) + l_{Put}(t)(a|s)$ ;
  - $M'[i + 1] = M[i + 1] - l_{GetR}(t) + l_{PutR}(t)$ ;
  - $M'[j] = M[j]$  при  $j \notin \{i - 1, i, i + 1\}$
- Такое срабатывание обозначим как  $M \xrightarrow[i]{t} M'$ .

**Лемма 1.** Пусть  $M, M' \in \mathcal{M}^\infty(Const)$ ,  $t \in T_a$  для некоторого  $a \in \Omega$ . Тогда

$$M \xrightarrow[i]{t} M' \wedge t^\bullet \subseteq \bullet t \Rightarrow M' \subseteq M.$$

*Доказательство.* Непосредственной подстановкой в определения активности и срабатывания перехода.  $\square$

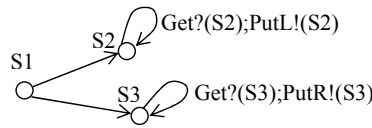
**Лемма 2.** Пусть  $M, M' \in \mathcal{M}^\infty(Const)$ ,  $t \in T_a$  для некоторого  $a \in \Omega$ . Тогда

$$M \xrightarrow[i]{t} M' \wedge |t^\bullet| \leq |\bullet t| \Rightarrow |M'| \leq |M|.$$

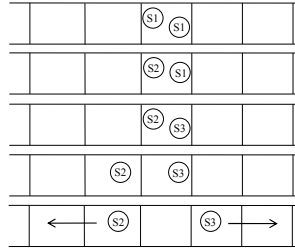
*Доказательство.* Аналогично. Заметим также, что с точки зрения количества автоматов не важно, в каком направлении они перемещаются при срабатывании.  $\square$

**Следствие 3.** 1. Если для каждого перехода  $t$  каждого P-автомата  $a$  выполнено  $\bullet t \subseteq t^\bullet$ , то сеть локализована.

2. Если для каждого перехода  $t$  каждого P-автомата  $a$  выполнено  $|\bullet t| \leq |t^\bullet|$ , то сеть имеет ограниченную площадь.



а) Диаграмма переходов автомата



б) “Расползание” автоматов

Рис. 3. Сеть неограниченного диаметра

Fig. 3. Net with unbounded diameter

Сформулированные выше свойства тривиальны: если переходы не увеличивают разметку соседних ячеек, то и разметка всей сети не увеличивается; а если переходы не увеличивают общее количество автоматов, то и занимаемая площадь не может вырасти.

Для свойства ограниченности диаметра таких простых ограничений уже недостаточно: даже “неувеличивающая” сеть может неограниченно разрастаться (Рис. 3).

Следующая теорема устанавливает, что для клеточных сетей существует аналог свойства монотонности сетей Петри (см., например, [4]):

**Теорема 3.** Пусть  $M, M', K \in \mathcal{M}^\infty(\text{Const}), t \in T_a$  для некоторого  $a \in \Omega$ . Тогда

$$M \xrightarrow{i} M' \Rightarrow M + K \xrightarrow{i} M' + K.$$

*Доказательство.* Заметим, что при рассмотрении конечного числа срабатываний (в частности, одного) клеточная сеть ведёт себя как обычная Р-сеть с конечным числом ячеек системной сети (т.е. как обычная сеть Петри). Далее доказательство аналогично доказательству соответствующей теоремы в [4].  $\square$

Однако, в отличие от обыкновенных сетей Петри, клеточные Р-сети не являются вполне структурированными системами переходов [13], так как отношение  $\subseteq$  на множестве  $\mathcal{M}^\infty(\text{Const})$  (определение 10) не является правильным квазипорядком.

### 2.3. Граф распространения

Рассмотрим способ описания взаимного перемещения Р-автоматов на одномерной решетке при отсутствии сдерживания со стороны *Get*-выражений (то есть в предположении наличия неограниченных ресурсов в каждой ячейке).

Далее представлен алгоритм, строящий так называемый *граф распространения* системы P-автоматов. Этот граф представляет собой (не всегда конечную) комбинацию диаграмм переходов всех автоматов из  $\Omega$  во всех ячейках сети, до которых они могут “распространиться” при старте процесса из начальной ячейки. При этом различные диаграммы, находящиеся в одной и той же ячейке или в двух соседних, могут быть связаны дополнительными “порождающими дугами”. Неформально, если автомат  $a$  в состоянии  $s$  способен сработать, генерируя (через какой-то из Put-портов) новый автомат  $a'$  в состоянии  $s'$ , то в графе распространения добавляется “генерирующая дуга” из  $a|s$  в  $a'|s'$ . При этом если  $a'|s'$  был сгенерирован через  $PutL$ , то дуга идёт в соседнюю левую ячейку, а если через  $Put$  или  $PutR$ , то, соответственно, в текущую или соседнюю правую. При этом Get-выражения не учитываются (и, следовательно, никак не сдерживают развитие процесса распространения автоматов), то есть получается аппроксимация сверху поведения любого сообщества клеток-автоматов с типами из  $\Omega$ .

**Алгоритм 1.** (*граф распространения*)

$PropagationGraph(\Omega)$

**Вход:** множество типов  $\Omega = \{a_1, \dots, a_k\}$ .

**Выход:** граф распространения  $\mathcal{G}_{prop}(\Omega)$ .

1. Положим  $G$  — пустой граф, обозначим все состояния всех P-автоматов при знаке “новое”.
2. Пока в автоматах есть “новые состояния”, повторим следующее: выберем “новое” состояние  $a|s$ , обозначим его признаком “старое”, достроим граф распространения  $G$  при помощи вызова процедуры  $Propagation(G, a|s, 0, \emptyset, \emptyset)$ .
3. Вернём  $G$ .

**Алгоритм 2.** (*достройка графа распространения*)

$Propagation(G, a|s, Shift, t, v)$

**Вход:** Ориентированный граф  $G$ ; состояние  $a|s$  ( $a \in \Omega, s \in S_a$ ); сдвиг  $Shift \in \{-1, 0, +1\}$ ; переход  $t$  ( $t \in T_b$  для некоторого  $b \in \Omega$ ); вершина  $v$  графа  $G$  (помеченная тройкой  $i|b|q$ , такой, что  $b|q \xrightarrow{t} \dots, i \in \mathbb{Z}$ , причем  $(a|s) \in l_{PutL}(t)$  при  $Shift = -1$ ,  $(a|s) \in l_{Put}(t)$  при  $Shift = 0$  и  $(a|s) \in l_{PutR}(t)$  при  $Shift = +1$ ).

**Выход:** Ориентированный граф  $G$ .

1. Добавим (в случае отсутствия) в граф  $G$  вершину с меткой  $i + Shift|a|s$  и “генерирующую” дугу от  $i|b|q$  к  $i + Shift|a|s$  с двойной меткой  $(Shift, t)$  (меткой сдвига  $Shift$  и меткой перехода  $t$ ).
2. Рассмотрим все вершины графа  $G$  с метками вида  $i + Shift|a|x$  для всех  $x \in S_a$ . Пока в диаграмме переходов автомата  $a$  найдётся состояние  $a|y$ , достижимое от  $a|x$  посредством перехода  $z$  (т.е.  $a|x \xrightarrow{z} a|y$ ), для которого в графе  $G$  ещё нет вершины с меткой  $i + Shift|a|y$  или дуги от  $i + Shift|a|x$  к  $i + Shift|a|y$  с одинарной меткой перехода  $z$ , добавим в  $G$  такую вершину  $u$ /или дугу.

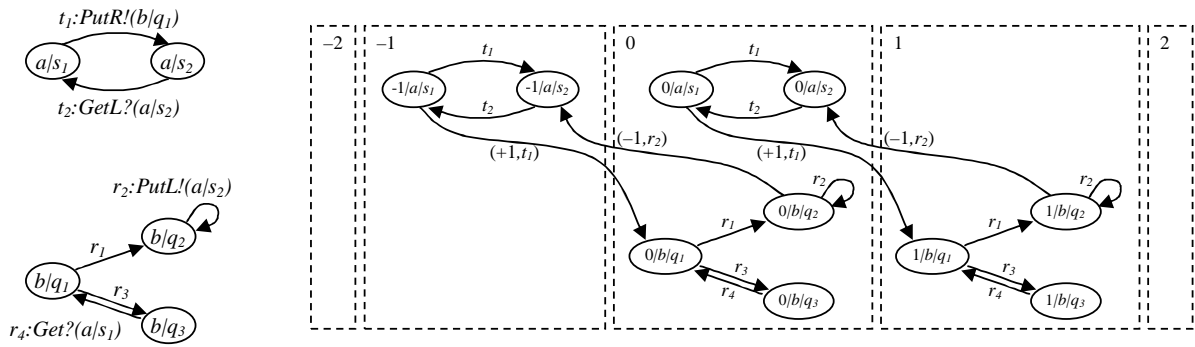


Рис. 4. Множество P-автоматов  $\Omega = \{a, b\}$  и его граф распространения

Fig. 4. A set of RDAs  $\Omega = \{a, b\}$  and its propagation graph

- За. Для каждой вершины  $v'$  с меткой вида  $j|a|x$ , добавленной в граф  $G$  на предыдущем шаге, перехода  $t' \in T_a$  и константы  $a'|s'$ , таких что  $a|x \xrightarrow{t'} a|y$  для некоторого  $y$  и  $(a'|s') \in l_{PutL}(t')$ , выполним  $Propagation(G, a'|s', -1, t', v')$ .
- Зб. Для каждой вершины  $v'$  с меткой вида  $j|a|x$ , добавленной в граф  $G$  на предыдущем шаге, перехода  $t' \in T_a$  и константы  $a'|s'$ , таких что  $a|x \xrightarrow{t'} a|y$  для некоторого  $y$  и  $(a'|s') \in l_{Put}(t')$ , выполним  $Propagation(G, a'|s', 0, t', v')$ .
- Зв. Для каждой вершины  $v'$  с меткой вида  $j|a|x$ , добавленной в граф  $G$  на предыдущем шаге, перехода  $t' \in T_a$  и константы  $a'|s'$ , таких что  $a|x \xrightarrow{t'} a|y$  для некоторого  $y$  и  $(a'|s') \in l_{PutR}(t')$ , выполним  $Propagation(G, a'|s', +1, t', v')$ .

Пример графа распространения приведён на Рис. 4. Здесь рассмотрены автоматы двух типов ( $a$  и  $b$ ), способные создавать друг друга. Правая часть рисунка представляет граф  $\mathcal{G}_{prop}(\{a, b\})$ , для наглядности разбитый на “ячейки”. Видно, что заполнены только ячейки с номерами -1, 0 и 1. Дуги  $(+1, t_1)$  и  $(-1, r_2)$  соответствуют событиям генерации новых автоматов в соседних ячейках (правой и левой соответственно) при срабатываниях  $t_1$  и  $r_2$ .

Довольно очевидно, что любой конечный набор P-автоматов типов  $a$  и  $b$ , изображенных на рисунке 4, не сможет “расползтись” по решетке системной P-сети более чем на одну ячейку влево и одну ячейку вправо относительно своих стартовых позиций. Это следует из того, что никакая последовательность переходов в соответствующем графе распространения не уходит далеко от той ячейки, в которой она “стартовала.” В данном случае ячейки графа  $\mathcal{G}_{prop}(\Omega)$  можно рассматривать как абстрактные аналоги ячеек решетки системной сети (использующие не абсолютную нумерацию — номера ячеек, а относительную — сдвиги номеров ячеек относительно исходной).

**Утверждение 3.** Число компонент связности графа  $\mathcal{G}_{prop}(\Omega)$  не превосходит  $|\Omega|$ .

*Доказательство.* Заметим, что по построению графа каждая вершина достижима из некоторой вершины вида  $(0|a|s)$  (т.е. вершины, находящейся в нулевой ячейке). А

в нулевой ячейке находятся в точности все вершины всех диаграмм переходов всех автоматов. Кроме того, в ней также имеются все внутренние дуги диаграмм переходов (с одинарными метками), а также может присутствовать некоторое количество генерирующих дуг с двойными метками.  $\square$

Единственный способ получения неограниченного распространения автоматов влево и/или вправо — бесконечные пути в соответствующем графе распространения.

**Определение 12.** *Бесконечный ациклический путь в графе распространения назовём генерирующим.*

*Генерирующий путь, проходящий через все ячейки с индексами, не превышающими некоторого  $i \in \mathbb{Z}$ , назовём генерирующим влево.*

*Генерирующий путь, проходящий через все ячейки с индексами, превышающими некоторое  $i \in \mathbb{Z}$ , назовём генерирующим вправо.*

**Утверждение 4.** 1. *Любой генерирующий путь является либо генерирующим влево, либо генерирующим вправо.*

2. *Никакой путь не может быть одновременно генерирующим влево и генерирующим вправо.*

3. *В ациклическом пути  $\sigma$  найдутся две одинаковые генерирующие дуги с метками  $(-1, t)$ , такими, что вторая из них находится в графе левее первой (т.е. стартует в ячейке с меньшим индексом)  $\Leftrightarrow$  путь  $\sigma$  является генерирующим влево.*

4. *В ациклическом пути  $\sigma$  найдутся две одинаковые генерирующие дуги с метками  $(+1, t)$ , такими, что вторая из них находится в графе правее первой (т.е. стартует в ячейке с большим)  $\Leftrightarrow$  путь  $\sigma$  является генерирующим вправо.*

*Доказательство.* (1) Заметим, что в каждой ячейке графа распространения содержится конечное множество вершин и дуг. Следовательно, любой бесконечный ациклический путь должен проходить через бесконечное множество ячеек. Кроме того, все дуги, не являющиеся внутренними дугами ячейки, связывают между собой исключительно соседние ячейки. Следовательно, путь, проходящий через ячейки с индексами  $i$  и  $i + 2$ , проходит также и через ячейку с индексом  $i + 1$ .

(2) Предположим противное: такой путь  $\sigma$  существует. Но в этом случае он должен бесконечно много раз проходить через центральную ячейку. Тогда в силу конечности числа вершин в ячейке мы получим нарушение свойства ациклическости.

(3) ( $\Rightarrow$ ) Предположим противное: в графе имеется такой не генерирующий влево ациклический путь:

$$\begin{aligned} \sigma = \dots \rightarrow (i|a|s) \xrightarrow{(-1,t)} (i-1|a'|s') \rightarrow \dots \\ \rightarrow (j|a|s) \xrightarrow{(-1,t)} (j-1|a'|s') \rightarrow \dots, \text{ где } i > j. \end{aligned}$$

Однако в таком случае по построению графа эта дуга должна повториться ещё раз ещё левее:  $(j+(j-i)|a|s) \xrightarrow{(+1,t)} (j+1+(j-i)|a'|s')$ . Более того, она будет повторяться

бесконечно много раз на границах ячеек с шагом  $(j - i)$ . Следовательно, путь бесконечен и проходит все ячейки слева от  $i$ -ой, то есть является генерирующим влево — противоречие.

(3) ( $\Leftarrow$ ) В силу бесконечности числа пересечений справа налево границ ячеек и конечности множеств переходов автоматов хотя бы один из них должен повторяться.

(4) Симметрично предыдущему случаю.  $\square$

**Теорема 4.** *Граф  $\mathcal{G}_{prop}(\Omega)$  конечен тогда и только тогда, когда в нём нет ни одного генерирующего влево или вправо пути.*

*Доказательство.* ( $\Leftarrow$ ) В силу первой части Утверждения 4 в графе нет ни одного бесконечного ациклического пути. В силу Утверждения 3 число компонент связности также конечно. Следовательно, граф конечен.

( $\Rightarrow$ ) Без ограничения общности достаточно рассмотреть только один из двух типов генерирующих путей. Рассмотрим генерацию вправо.

Предположим противное: в графе имеется путь с двумя такими дугами:  $(i|a|s) \xrightarrow{(+1,t)}$   $(i+1|a'|s')$  и  $(j|a|s) \xrightarrow{(+1,t)}$   $(j+1|a'|s')$ . Однако в таком случае по построению графа эта дуга должна повториться ещё раз ещё правее:  $(j+(j-i)|a|s) \xrightarrow{(+1,t)}$   $(j+1+(j-i)|a'|s')$ . Более того, она будет повторяться бесконечно много раз на границах ячеек с шагом  $(j - i)$ . Следовательно, граф бесконечен — противоречие.  $\square$

**Алгоритм 3.** *(проверка конечности графа распространения  $\mathcal{G}_{prop}(\Omega)$ )*

*GraphIsFinite( $\Omega$ )*

**Вход:** *множество типов  $\Omega = \{a_1, \dots, a_k\}$ .*

**Выход:** *ответ “конечен” или “бесконечен”.*

**Основной цикл:** *Запустим процедуру PropagationGraph( $\Omega$ ), контролируя возникновение генерирующих цепочек. Возможные варианты завершения:*

- *Граф построен  $\Rightarrow$  ответ “конечен”.*
- *В процессе построения возникла генерирующая цепочка  $\Rightarrow$  ответ “бесконечен”.*

Алгоритм сходится: в противном случае в бесконечном графе отсутствовали бы генерирующие цепочки.

**Теорема 5.** *Если размеченная клеточная P-сеть с автоматами из  $\Omega$  не локализована (неограничена в смысле (1)), то граф  $\mathcal{G}_{prop}(\Omega)$  бесконечен.*

*Доказательство.* Предположим противное: граф конечен, хотя сеть  $N$  распространяется неограниченно влево и/или вправо. Однако очевидно, что любой (в том числе бесконечный) процесс порождения автоматов (цепочка срабатываний переходов) в размеченной сети должен иметь аналог в графе распространения, поскольку тот содержит все возможные последовательности срабатываний автоматов из  $\Omega$  в условиях отсутствия каких-либо ограничений по ресурсам. И если сеть не является локальной (например, неограниченно распространяется вправо), то соответствующая её разрастанию бесконечная последовательность переходов возможна и в  $\mathcal{G}_{prop}(\Omega)$ . Следовательно, граф бесконечен вправо — противоречие.  $\square$



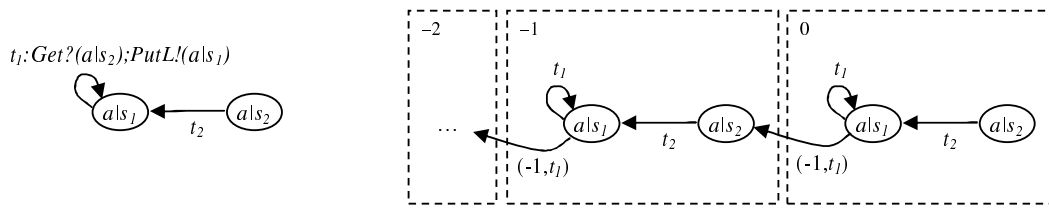


Рис. 5. Р-автомат с бесконечным графом распространения (генерирующий влево путь), который порождает локализованную сеть

Fig. 5. RDA with an infinite propagation graph (left-generating path), producing localized net

Обратное не всегда верно: существует локализованная сеть, граф для которой бесконечен.

На Рисунке 5 изображен Р-автомат, граф распространения которого содержит бесконечный генерирующий влево путь. Однако выполнение такого пути возможно только при наличии в каждой ячейке с отрицательным номером автомата ( $a|s_2$ ) (т.к. генерирующий переход  $t_1$  помечен соответствующим Get-выражением). Однако начальная разметка всегда конечна, поэтому таких автоматов на решетке может быть только конечное число. И взяться им неоткуда, так как генерируется только константа ( $a|s_1$ ), а из состояния  $s_1$  состояние  $s_2$  недоступно. Таким образом, бесконечный генерирующий путь никогда не реализуется полностью.

В силу Теоремы 5 Алгоритм 3 позволяет обнаруживать неограниченность сети в смысле (1) (то есть отсутствие локализованности). Действительно, достаточно проверить конечность графа  $\mathcal{G}_{prop}(\Omega)$ .

Заметим, что данный алгоритм является не разрешающей, а полурешающей процедурой для пространственной ограниченности, поскольку допускает “false positive”: для некоторых ограниченных сетей он может дать ответ “не ограничена” (пример на Рисунке 5). В то же время он всегда правильно подтверждает неограниченность, и поэтому является практически значимым.

Кроме того, граф распространения в случае своей конечности позволяет оценивать сверху диаметр локальных клеточных сетей:

**Теорема 6.** Пусть граф  $\mathcal{G}_{prop}(\Omega)$  конечен и содержит  $n$  “ячеек”. Тогда диаметр любой клеточной сети с автоматами из  $\Omega$  в ходе её функционирования не может вырасти более, чем на  $n$  относительно своего начального значения.

*Доказательство.* Очевидно, поскольку никакой из начальных автоматов не может породить потомков на большем расстоянии.  $\square$

### 3. Заключение

В работе показано, что проблема пространственной ограниченности клеточных Р-сетей (и, следовательно, бесконечных сетей Петри) является в общем случае неразрешимой. Вводятся три варианта определения пространственной ограниченности: локализованность, ограниченность площади и ограниченность диаметра.

Предложено понятие графа распространения Р-автоматов в одномерной клеточной сети. Разработан алгоритм построения графа распространения, использующий новое свойство “пространственной монотонности”. Предложен нетривиальный критерий локализованности модели, основанный на проверке конечности графа распространения. Предложен метод оценки сверху диаметра локализованной клеточной Р-сети с конечным графом распространения.

В качестве возможных направлений дальнейших исследований мы рассматриваем поиск более слабых критериев ограниченности, а также изучение некоторых других “пространственных” свойств клеточных ресурсных моделей: “объема” модели, устойчивых конфигураций, проблем замещения и т.п.

Автор выражает благодарность Горностаевой Екатерине за помощь в подготовке текста статьи.

### Список литературы / References

- [1] Башкин В. А., “Сети активных ресурсов”, *Моделирование и анализ информационных систем*, **14**:4 (2007), 13–19; [Bashkin V.A., “Nets of active resources”, *Model. Anal. Inform. Syst.*, **14**:4 (2007), 13–19, (in Russian).]
- [2] Башкин В. А., “Формализация семантики систем с ненадежными агентами при помощи сетей активных ресурсов”, *Программирование*, **36**:4 (2010), 3–15; English transl.: Bashkin V. A., “Formalization of semantics of systems with unreliable agents by means of nets of active resources”, *Progr. and Comp. Soft.*, **36**:4 (2010), 187–196.
- [3] Зайцев Д. А., “Верификация вычислительных решеток с особыми краевыми условиями бесконечными сетями Петри”, *Моделирование и анализ информационных систем*, **19**:6 (2012), 21–33; English transl.: Zaitsev D. A., “Verification of computing grids with special edge conditions by infinite Petri nets”, *Automatic Control and Computer Sciences*, **47**:7 (2013), 403–412.
- [4] Котов В. Е., *Сети Петри*, Наука, М., 1984; [Kotov V. E., *Seti Petri*, Nauka, Moskva, 2008, (in Russian).]
- [5] Кузьмин Е. В., Чалый Д. Ю., “О разрешимости проблем ограниченности для счетчиковых машин Минского”, *Моделирование и анализ информационных систем*, **15**:1 (2008), 16–26; [Kuzmin E. V., Chalyu D. Ju., “On the decidability of boundedness problems for counter Minsky machines”, *Model. Anal. Inform. Syst.*, **15**:1 (2008), 16–26. (in Russian).]
- [6] Alur R., Dill D., “Automata for modeling real-time systems” (Automata, Languages and Programming), *Lecture Notes In Computer Science*, **443** (1990), 322–335.
- [7] Bashkin V. A., “Nets of active resources for distributed systems modeling”, *Joint Bulletin of NCC&IIS, Comp. Science*, **28** (2008), 43–54.
- [8] Bashkin V. A., Lomazova I. A., “Resource Driven Automata Nets”, *Fundamenta Informaticae*, **109**:3 (2011), 223–236.
- [9] Bashkin V. A., Lomazova I. A., “Cellular Resource Driven Automata Nets”, *Fundamenta Informaticae*, **120**:3-4 (2012), 245–259.

- [10] Bashkin V. A., Lomazova I. A., Novikova Yu. A., “Timed Resource Driven Automata Nets for Distributed Real-Time Systems Modelling” (Parallel Computing Technologies), *Lecture Notes In Computer Science*, **7979** (2013), 13–25.
- [11] Bednarczyk M. A., Bernardinello L., Pawlowski W., Pomello L., “Modelling mobility with Petri Hypernets” (Recent Trends in Algebraic Development Techniques), *Lecture Notes In Computer Science*, **3423** (2005), 28–44.
- [12] Cook M., “Universality in Elementary Cellular Automata.”, *Complex Systems*, **15**:1 (2004), 1–40.
- [13] Finkel A., Schnoebelen Ph., “Well-Structured Transition Systems Everywhere!”, *Theoretical Computer Science*, **256**:1–2 (2001), 63–92.
- [14] Jensen K., *Colored Petri Nets: Basic Concepts, Analysis Methods and Practical Use*, Springer, 1994.
- [15] Köhler-Bußmeier M., “Hornets: Nets within Nets combined with Net Algebra” (ICATPN’2009), *Lecture Notes In Computer Science*, **5606** (2009), 243–262.
- [16] Lomazova I. A., “Nested Petri Nets – a Formalism for Specification and Verification of Multi-Agent Distributed Systems”, *Fundamenta Informaticae*, **43**:1–4 (2000), 195–214.
- [17] Nehaniv C.L., “Asynchronous Automata Networks Can Emulate Any Synchronous Automata Network”, *Int. J. of Algebra and Computation*, **14**:5–6 (2004), 719–739.
- [18] Petri C. A., *Kommunikation mit Automaten. PhD thesis*, Bonn Institute für Instrumentelle Mathematik, 1962.
- [19] Valk R., “Petri Nets as Token Objects: An Introduction to Elementary Object Nets” (ICATPN’98), *Lecture Notes In Computer Science*, **1420** (1998), 1–25.

---

**Bashkin V. A.**, "On the Spatial Boundedness of Cellular RDA-nets", *Modeling and Analysis of Information Systems*, **24**:4 (2017), 391–409.

**DOI:** 10.18255/1818-1015-2017-4-391-409

**Abstract.** Cellular resource driven automata nets (CRDA-nets) is a generalization of the concept of two-level resource nets (Petri nets) with an infinite regular system grid. This formalism is a hybrid of Petri nets and asynchronous Cellular Automata and is designed for modeling multi-agent systems with dynamic spatial structure. Spatial boundedness is a property that guarantees the preservation of the finiteness of “geometric dimensions” of the active part of the system (for example, the living space) during its lifetime. Three variants of spatial boundedness for cellular RDA-nets are defined: localization, bounded diameter and bounded area. The properties of the corresponding algorithmic problems are investigated, their undecidability in the general case is proved. A non-trivial criterion for the localization of an one-dimensional CRDA-net is proposed, based on the new concept of the RDA propagation graph. An algorithm is described for constructing a propagation graph, using the method of saturation of generating paths. A method for estimating the diameter of an 1-dim CRDA with a bounded propagation graph is presented.

**Keywords:** multiagent systems, verification, Petri nets, cellular automata, resource driven automata nets, spatial boundedness

**On the authors:**

Vladimir A. Bashkin, orcid.org/0000-0002-2534-1026, PhD,  
P.G. Demidov Yaroslavl State University,  
14 Sovetskaya str., Yaroslavl 150003, Russia, e-mail: v\_bashkin@mail.ru

**Acknowledgments:**

This work was supported by the Russian Fund for Basic Research (17-07-00823).