

О моделировании счётчиков с бесконечным числом значений в обычных сетях Петри.

Л. В. Дворянский
leo@ispras.ru

Аннотация. В статье проведен анализ моделирования в обычных сетях Петри счётчиков с бесконечным числом состояний. Обоснован выбор отношения эквивалентности симуляции готовности в качестве отношения реализации для моделирования счётчиков. Показано, что в сетях Петри невозможно промоделировать счётчики с бесконечным числом состояний. Представлена минимальная модель счётчика с конечным числом значений.

Ключевые слова: счётчиковые машины; сети Петри; спектр фон Глэббика; отношение эквивалентности симуляции готовности.

1. Введение

При моделировании поведения дискретных динамических систем часто явно или неявно возникает необходимость моделирования дискретного счётчика. Счётчик один из основных элементов семантики исполнения дискретных динамических систем. Счётчиковые машины Минского с двумя счётчиками являются универсальными вычислителями. В других моделях счётчики неявно присутствуют в виде стека, множества управляемых состояний, буфера сообщений, определенным образом занумерованных строк и пр.

При моделировании динамических систем счётчики используются для выражения зависимости исполнения одних событий от числа произошедших других событий. К примеру, счётчики используются для ограничения длительности исполнения системы заданным числом шагов или выполнения повторяющихся событий с заданным периодом.

При моделировании счётчика с помощью сетей Петри, простейшим способом является использование позиции в качестве счётчика, при котором значению счётчика соответствует значение функции разметки на позиции. В работе [1] используется подобный подход для моделирования машин Минского с помощью сетей Петри, называемый в статье «моделирование в слабом смысле» («simulate in a weak sense»). Под «моделированием в слабом смысле» понимается то, что возможны «ложные» («cheating») срабатывания переходов,

которые не могут присутствовать в исполнении оригинальной системы со счётчиком. Неизбежность ложных срабатываний объясняется свойством монотонности входных условий переходов в операционной семантике сетей Петри (см. [2]). Такая конструкция полезна для доказательства некоторых свойств (см. [1]) формализма сетей Петри, но не подходит для моделирования исполнения системы со счётчиком.

Целью данной работы является изучение проблемы моделирования счётчиков без ложных срабатываний с помощью сетей Петри. Вначале, осуществляется выбор подходящего критерия корректности и отношения реализации для моделирования счётчиков сетями Петри. Затем рассматривается возможность моделирования счётчика со счётным множеством значений в сетях Петри. В конце приводится простая (и минимальная) модель счётчика с конечным множеством значений.

2. Предварительные сведения

2.1. Отношение квазипорядка.

Определение 1. Бинарное отношение называется квазипорядком или предпорядком, если оно рефлексивно и транзитивно.

Определение 2. Квазипорядок называется фундированным, если всякое непустое подмножество множества-носителя имеет минимальный элемент.

Определение 3. Квазипорядок \leq на множестве X называется правильным, если для любой бесконечной последовательности x_0, x_1, x_2, \dots элементов из X существуют индексы $i \leq j$ такие, что $x_i \leq x_j$.

Утверждение 1. Квазипорядок \leq на множестве X является правильным в том, и только в том, случае, когда одновременно выполняются два следующих условия (см. [3], стр. 19.):

1. X не содержит бесконечных строго убывающих цепей $x_0 > x_1 > x_2 > \dots$;
2. всякая содержащаяся в X антицепь конечна.

2.2. Операции над цепочками символов.

Определение 4. Операция конкатенации $O: V^* \rightarrow V^*$ для последовательности цепочек $\alpha = \langle \alpha_1, \alpha_2, \alpha_3 \dots \rangle$ это операция последовательного склеивания цепочек:

$$\forall i \in I: \forall j \in \overline{1, |\alpha_i|}: l = \sum_{k=1, i-1} |\alpha_k| + j: (O(\alpha))_l = \alpha_{ij} \quad (2.1)$$

где I – множество индексов набора α , $\alpha_i \in V^*$ – цепочки над алфавитом

V, V^* – множество последовательностей таких цепочек, α_{ij} – j -ый

элемент i -ой цепочки набора цепочек α .

Определение 5. Проекция $\rho: A^* \times 2^A \rightarrow A^*$ цепочки $\alpha \in A^*$ на алфавит $B \subseteq A$ это цепочка полученная исключением из α всех элементов, кроме тех которые принадлежат алфавиту B .

$$\rho(\alpha, B) = O(\langle \langle \alpha_i \mid \alpha_i \in B \rangle \rangle) \quad (2.2)$$

2.3. Дискретный счётчик.

Счётчик – одна из основных абстракций, используемых в теории вычислимости для описания поведения дискретных по времени абстрактных машин. Регистр это сущность, которая хранит одно натуральное число. Счётчик является регистром с ограниченным набором команд. Одна из первых и простейших вычислительных моделей, использующих счётчики, является регистровая (или счётчиковая) машина Минского (см.[4], стр.244).

Каждый счётчик обладает значением, определяющим его состояние в наблюдаемый момент времени. Множество возможных значений счётчика является начальным отрезком множества натуральных чисел, т.е. ограничено снизу нулевым значением и может быть ограничено или неограниченно сверху. В последнем случае значение счётчика может являться любым натуральным числом из $\overline{0, \infty}$. Наблюдение за состоянием счётчика ограничено проверкой на ноль и на положительное значение. Т.е. модель, использующая счётчик, может только различать содержит ли счётчик 0 или положительное целое число.

Определим конфигурацию счётчика (или просто счётчик) C как

$$C =_{def} \langle V, v \rangle \quad (2.3)$$

где

$V = \overline{1, N}$ – множество возможных значений счётчика, где $N \in \mathbb{N} \cup \{\infty\}$;

$v \in V$ – значение счётчика.

Определение 6. Обозначим через C_V множество всех конфигураций счётчиков с множеством значений V .

Определение 7. Функция $v: C_V \rightarrow V$ определяет значения счётчика в наблюдаемый момент времени.

Над счётчиком определен набор операций

Операция инкремента $+: C_V \rightarrow C_V$ увеличивает значение счётчика на единицу.

$$\forall C \in C_V : v(C) \neq N : v(+ (C)) = v(C) + 1 \quad (2.4)$$

Операция декремента $-: C_V \rightarrow C_V$ уменьшает значение счётчика на единицу:

$$\forall C \in C_V : v(C) \neq 0 : v(- (C)) = v(C) - 1 \quad (2.5)$$

Пусть $B = \{true, false\}$, тогда

Операция проверки счётчика на нулевое значение $0_\gamma: C_V \rightarrow B$ имеет значение истина, когда счётчик имеет нулевое значение

$$\forall C \in C_V : (v(C) = 0) \Rightarrow 0_\gamma(C) = true \quad (2.6)$$

Операция проверки счётчика на положительное значение $\overline{0}_\gamma: \square_V \rightarrow B$ имеет значение истина, когда счётчик имеет положительное значение

$$\forall C \in C_V : (v(C) > 0) \Rightarrow \overline{0}_\gamma(C) = true \quad (2.7)$$

Значение счётчика меняется под воздействием операций. С каждым шагом исполнения модели значение счётчика может увеличиться на единицу, уменьшиться на единицу или остаться неизменной. Значение счётчика натуральное число, т.е. при нулевом значении счётчика операция декремента заблокирована (см.[5], стр.13). Состояние счётчика полностью определяется его значением.

2.4. Сеть Петри

В различных источниках предложены определения различающиеся способом обозначения составляющих формализма, хотя на семантическом уровне подобные формализмы эквивалентны. Подобные синтаксические и семиотические вопросы влияют лишь на удобство использования формализма. Зафиксируем одно из распространенных определений, приведенных в книге (см. [6], стр.14), сделав его чуть более строгим.

Определение 8. Сеть Петри N определяется, как кортеж

$$N = \langle P, T, F, W, m_0 \rangle \quad (2.8)$$

где

74

- P – непустое конечное множество позиций сети;
- T – непустое конечное множество переходов сети;
- F – отношение инцидентности;

$$F \subseteq P \times T \cup T \times P \quad (2.9)$$

- \mathbb{N} – множество натуральных чисел $\overline{0, \infty}$;
- W – функция кратности дуг $W : F \rightarrow \mathbb{N} \setminus \{0\}$;
- m_0 – начальная разметка $m_0 : P \rightarrow \mathbb{N}$.

При этом выполняются следующие условия:

- 1) Множества позиций и переходов не пересекаются:

$$P \cap T = \emptyset \quad (2.10)$$

- 2) Любой элемент сети инцидентен хотя бы одному элементу другого типа:

$$\forall x \in P \cup T : \exists y \in P \cup T : xFy \vee yFx \quad (2.11)$$

Определение 9. Обозначим через Π класс всех обыкновенных сетей Петри.

В некоторых случаях удобнее использовать функцию инцидентности $F : P \times T \cup T \times P \rightarrow \mathbb{N}$, которая задается на основе отношения инцидентности F и функции кратности дуг W :

$$F(x, y) = \begin{cases} W(x, y), & \text{если } xFy \\ 0, & \text{если } \neg xFy \end{cases} \quad (2.12)$$

2.4.1. Состояние сети Петри.

Для определения состояния сети Петри N необходимо ввести определение разметки.

Определение 10. Разметка это функция $m : P \rightarrow \mathbb{N}$, которая ставит в соответствие каждой позиции $p \in P$ целое число $n \in \mathbb{N}$.

Определение 11. Множество всех возможных разметок сети N обозначим, как $\mathbf{M}_N = \mathbb{N}^P$.

Определение 12. Так как, множество P конечно, то пронумеровав позиции произвольным образом, разметку можно представить в виде вектора

$$m_v = \langle m_1, \dots, m_{|P|} \rangle \quad (2.13)$$

Такой вектор полностью определяет функцию-разметку сети Петри [7].

Определим порядок \leq на разметках сети Петри используя векторное представление.

Определение 13. Порядок на разметках сетей Петри задается отношением $\leq_{\subseteq} \mathbf{M}_N \times \mathbf{M}_N$ таким, что $m \leq m'$ тогда и только тогда, когда значение m для каждой позиции меньше или равно значению m' для этой позиции.

$$\begin{aligned} \forall m, m' \in \mathbf{M}_N : m_v = \langle m_1, \dots, m_n \rangle, m'_v = \\ = \langle m'_1, \dots, m'_n \rangle : m \leq m' \Leftrightarrow \forall i \in \overline{1, n} : m_i \leq m'_i \end{aligned} \quad (2.14)$$

Утверждение 2. Пусть отношение \leq – правильный квазипорядок на множестве X . Тогда отношение \leq^n на множестве векторов размерности n из X^n также является правильным квазипорядком (см. [3], стр.22).

Утверждение 3. Отношение $\leq_{\subseteq} \mathbf{M}_N \times \mathbf{M}_N$ является правильным квазипорядком.

Доказательство. Рассмотрим (Утверждение 2), если в качестве множества X рассматривать \mathbb{N} , а в качестве квазипорядка \leq на X будем рассматривать естественный порядок «меньше или равно» на натуральных числах, то отношение порядка на разметках сетей Петри введенного в (Определение 13) соответствует отношению \leq^n .

Так как классическое отношение «меньше или равно» на множестве натуральных чисел \mathbb{N} является правильным квазипорядком, то в силу (Утверждение 2) отношение $\leq_{\subseteq} \mathbf{M}_N \times \mathbf{M}_N$ является правильным квазипорядком. \square

2.4.2. Исполнение сетей Петри.

Для формального определения исполнения сети Петри опишем входную и выходную функцию переходов. Для каждого перехода $t \in T$ входная функция $\bullet F(t) : T \rightarrow \mathbf{M}_N$ задает разметку (входное условие перехода, охрана перехода) на основе функции инцидентности:

$$\bullet F(t) = \langle f_1, \dots, f_n \rangle : \forall i \in \overline{1, n} : f_i = F(p_i, t) \quad (2.15)$$

Выходная функция $F^\bullet(t) : T \rightarrow \mathbf{M}_N$ определяется аналогично:

$$F^\bullet(t) = \langle f_1, \dots, f_n \rangle : \forall i \in \overline{1, n} : f_i = F(t, p_i) \quad (2.16)$$

Определение 14. Переход t в сети Петри N называется активным при разметке $m \in M_N$, когда

$$\bullet F(t) \leq m \quad (2.17)$$

Если переход активен он может быть выполнен. Исполнение перехода t переводит сеть Петри из разметки m в следующую разметку m' :

$$m' = m - \bullet F(t) + F^\bullet(t) \quad (2.18)$$

Исполнение активного перехода называется шагом исполнения сети Петри.

1. **Определение 15.** Множеством достижимости $R(N)$ сети Петри N называется множество всех разметок, достижимых из начальной разметки по правилу (2.18).

Пусть t переход в сети Петри, такой что $\bullet F(t) \leq m$. Определим операцию $\varepsilon: M_N \rightarrow M_N$ исполнения перехода t в сети Петри

$$\varepsilon(m, t) = m - \bullet F(t) + F^\bullet(t) \quad (2.19)$$

Исполнение μ сети Петри N определяется последовательным выполнением активных переходов из начальной разметки m_0 . Обозначим последовательность сработавших переходов как μ_i , таким образом, что $\mu_i(i)$ равен переходу $t \in T$ сработавшему на i -ом шаге. Каждый раз при выполнении перехода, текущая разметка и множество активных переходов могут измениться. Последовательность исполнения переходов задает последовательность разметок μ_m , первой из которых является $\mu_m(0) = m_0$, а все последующие соответствуют разметкам после срабатывания соответствующего перехода.

$$\forall i \in \overline{1, |\mu|}: \mu_m(i) = \varepsilon(\mu_m(i-1), \mu_i(i)) \quad (2.20)$$

Утверждение 4. Свойство монотонности исполнения переходов сетей Петри (см. [6], стр.19).

$$\begin{aligned} \forall m_1, m_2, m' \in M_N: m_2 = \varepsilon(m_1, t) \Rightarrow \\ (m_2 + m') = \varepsilon(m_1 + m', t) \end{aligned} \quad (2.21)$$

2.5. Системы помеченных переходов

Системы помеченных переходов получили широкое распространение в качестве унифицированного средства представления процессов дискретных систем в различных задачах верификации и тестирования основанного на моделях[8].

Определение 16. Система помеченных переходов LTS (Labelled Transition System) определяется, как множество состояний системы и отношение помеченных переходов на этом множестве (см. [9])

$$LTS =_{def} \langle S, s_0, \Lambda, \rightarrow \rangle \quad (2.22)$$

где S - множество состояний (возможно бесконечное), s_0 - начальное состояние, Λ - конечный алфавит пометок (имен действий), \rightarrow - тернарная операция соответствующая возможным переходам системы из одного состояния в другое в ответ на помеченные действия.

$$\rightarrow \subseteq S \times \Lambda \times S \quad (2.23)$$

Определение 17. Обозначим через L класс всех систем помеченных переходов.

Состояния представляются явно в виде элементов множества S . В отличие от конечных автоматов, множество S может быть бесконечно. Последнее свойство позволяет представлять в виде систем помеченных переходов системы, как с конечным, так и с бесконечным числом состояний, а также как конечно, так и бесконечно ветвящиеся процессы.

3. Контекст задачи

В рамках подхода к разработке программного обеспечения основанного на моделях (Model-Driven Development[10]), областях моделирования и анализа различных потоковых моделей бизнес-, технологических и других подобных процессов, а также в области координирования Web-сервисов стали использоваться модели с бесконечным числом состояний обладающих параллелизмом и недетерминизмом. Многие языки моделирования (UML2 Activity Diagrams, BPMN, BPEL4People и др.) используют в качестве основы для определения операционной семантики выполнения хорошо изученный формализм сетей Петри. Преимущества использования традиционного формализма заключается в строгости семантики и изученности вопросов, связанных с разрешимостью и сложностью многих задач анализа таких моделей. Конечно, при построении таких моделей руководствуются удобством моделирования, поэтому в семантику вносятся расширения, которые увеличивают выразительную мощь моделей и модели выходят за рамки обыкновенных сетей Петри. Однако некоторые конструкции удается

выразить и на языке сетей Петри. Для таких конструкций можно применять все известные алгоритмы анализа на сетях Петри.

Данная работа посвящена анализу одной из сложностей при моделировании реальных моделей с помощью обыкновенных сетей Петри, а именно моделирования счётчика с бесконечным числом состояний.

С другой стороны развитие теории алгебры процессов привело к развитию понятия соответствия процессов. То, что один процесс ведет себя аналогично другому (моделирует его) может быть выражено различными отношениями эквивалентности и предпорядка (см. например [11], [12]). Выбор конкретного отношения зависит от рассматриваемой задачи и зависит от семантики языков процессов и ожидаемого соответствия. В данной работе предложен и обоснован выбор отношения симуляции готовности (ready simulation) в качестве такого отношения.

4. Выбор отношения реализации

Корректность моделирования динамической системы обычно обуславливается тем, что модель в некотором определенном смысле ведет себя аналогично моделируемой системе. Критерий корректности определяется в зависимости от решаемой задачи (см. [11]), и обычно задается в виде некоторого отношения между моделью и моделируемым объектом. Такое отношение называется отношением реализации (implementation relation). Отношение реализации может быть установлено между процессами, на основе отношения между множествами трасс процессов, множествами возможных переходов, множествами переходов, которые не могут выполняться (refusal set) и другими артефактами исполнения процессов. Выбор подходящего отношения зависит от особенностей задач и объектов моделирования.

4.1. Процессы счётчика и сетей Петри

Процессы рассматриваются с позиции унифицированного параллелизма (uniform concurrency) [13], т.е. все операции рассматриваются как атомарные и нам не важен внутренний характер действий.

Процессы обеих моделей допускают бесконечные исполнения, поэтому не являются конечными. В случае счётчика простейшим бесконечным исполнением является бесконечное повторное исполнение операции инкремента и операции декремента. Пример бесконечного исполнения в сетях Петри - бесконечное исполнение перехода с одним входящим и одним исходящим ребрами, инцидентными одной позиции.

Обе модели задают конечно-ветвящиеся процессы. В случае счётчиков, множество возможных действий ограничено операциями, заданными на счётчиках. В случае сетей Петри, число ветвлений ограничено конечным множеством переходов, т.к. каждое действие моделируется переходом. Из

конечности ветвления следует, что обе модели также являются конечно-отображаемыми (image finite) процессами.

Счётчик является детерминированной моделью, т.к. каждое последующее состояние однозначно определяется текущим значением счётчика и выполняемой операции, т.е.

$$\forall a \in \{+, -, 0, \bar{0}\} : \left(a(C_1) = C_2 \right) \& \left(a(C_1) = C_3 \right) \rightarrow C_2 = C_3 \quad (4.1)$$

Помеченные сети Петри относят к недетерминированным моделям, т.к. одним действием могут быть помечены несколько переходов и в результате выполнения одноименных переходов модель может переходить в различные состояния.

Мы не будем рассматривать внутренние события, которые могут осуществляться во время исполнения процесса, но не влияют на его функционирование. Поэтому наблюдаемые процессы являются непрерывными (concrete), т.е. такими, между явными действиями которых ничего не происходит. (Для систем помеченных переходов формально это означает, что все возможные цепочки τ событий конечны и игнорируются).

Таким образом, счётчик определяется классом не конечных, конечно-ветвящихся, детерминированных и непрерывных процессов. Формализм сетей Петри лежит в классе не конечных, конечно-ветвящихся, недетерминированных и непрерывных процессов.

Для того, чтобы иметь возможность сравнивать поведение порождаемое счётчиком и сетевыми моделями, необходимо привести счётчик и сети Петри к единому представлению. Существует много различных способов представления систем – системы помеченных переходов (labeled transition system), процессные графы (process graph), структуры заданные на множестве событий (event structure), сетевые модели, проективные пределы на рядах конечных термов и другие. Отношение эквивалентности может быть определено на любом из этих представлений. Представление процессов в виде систем помеченных переходов является удобным и распространенным способом представления произвольных процессов, чье поведение может быть выражено с помощью отношения действий.

Определим отображение $L_C : C \rightarrow \mathbf{L}$ счётчиков в системы помеченных переходов, как

$$L_C(C) = \langle S_C, s_{C0}, \Lambda_C, \rightarrow_C \rangle \quad (4.2)$$

где S_C класс состояний, занумерованный множеством значений счётчика

$$S_C = \langle s_i : i \in V \rangle \quad (4.3)$$

$$s_{C0} = s_{v_0} \quad (4.4)$$

$$\Lambda_C = \{+, -, 0_\gamma, \overline{0}_\gamma\} \quad (4.5)$$

Отношение действия \rightarrow_C определено, как

$$\forall i, j \in V : \forall a \in \{+, -\} : (\langle s_i, a, s_j \rangle \in \rightarrow_C) \Leftrightarrow (j = a(i)) \quad (4.6)$$

$$\forall i, j \in V : \forall a \in \{0_\gamma, \overline{0}_\gamma\} : (\langle s_i, a, s_j \rangle \in \rightarrow_C) \Leftrightarrow (i = j \& a(i)) \quad (4.7)$$

Определим отображение $L_N : \Pi \rightarrow \mathbf{L}$ сетей Петри в системы помеченных переходов, как

$$L_N(N) = \langle S_{M_N}, s_{m_0}, \Lambda_N, \rightarrow_N \rangle \quad (4.8)$$

$$S_{M_N} = \{s_i : i \in M_N\} \quad (4.9)$$

$$\Lambda_N = \Lambda_C \quad (4.10)$$

Введем функцию $\lambda : T \rightarrow \Lambda_N$ пометки переходов сети Петри, которая ставит в соответствие переходу пометку (действие), которое он моделируют.

1. Отношение действия \rightarrow_N определено, как

$$\begin{aligned} \forall i, j \in M : \forall a \in \Lambda_N : (\langle s_i, a, s_j \rangle \in \rightarrow_N) \Leftrightarrow \\ \Leftrightarrow (\exists t \in T : t \leq i \& \lambda(t) = a \& j = \varepsilon(i, t)) \end{aligned} \quad (4.11)$$

Можно установить отображение $L_{NT} : \rightarrow_N \rightarrow T_N$ между элементами отношения \rightarrow_N и переходами сети Петри N :

$$\begin{aligned} \forall \langle s_{m_1}, a, s_{m_2} \rangle \in \rightarrow_N : L_{NT}(\langle s_{m_1}, a, s_{m_2} \rangle) = t_a \Leftrightarrow \\ \Leftrightarrow m_2 = \varepsilon(m_1, t_a) \end{aligned} \quad (4.12)$$

Используя отношения L_C и L_N можно устанавливать отношение между множеством процессов порождаемых счётчиком C и сетью Петри N сравнивая системы помеченных переходов $L_C(C)$ и $L_N(N)$.

4.2. Отношение реализации (implementation relation).

Два процесса считаются эквивалентными, если они показывают одинаковое наблюдаемое поведение при одинаковых воздействиях, как следствие принципа равенства неразличимого. Отношения эквивалентности различаются способом наблюдения за поведением процесса. Способ наблюдения выбирается с учетом особенностей решаемой задачи.

В нашей задаче, единственной операцией позволяющей наблюдать за состоянием счётчика является проверка на ноль. Поэтому, при одинаковом воздействии на счётчик и на сетевую модель, они должны обеспечивать одинаковые значения операции 0_γ на всем протяжении исполнения. При этом внешняя система может обратиться к операции проверки на ноль значения счётчика на любом шаге. Т.е. процессы счётчика и модели счётчика должны давать одинаковое значение проверки на ноль в любой наблюдаемый момент. Операция 0_γ моделируется, как дополнительный переход на нулевом состоянии счётчика.

Окружение счётчика может воздействовать на счётчик с помощью операций инкремента и декремента. Таким образом, множество возможных действий системы помеченных переходов должно соответствовать множеству возможных действий самого счётчика в заданный момент исполнения. Т.е. всегда, когда над счётчиком можно выполнить операции инкремента или декремента, те же операции должны быть выполнимы над его моделью. Так как окружение счётчика может обратиться к любой из указанных операций определенных над счётчиком, возникает следующее требование.

Требование 1. В каждый наблюдаемый момент процесс корректной модели должен разрешать все, и только те действия, которые разрешены на процессе счётчика.

Так как окружение определяет исполнение каждого следующего действия, то счётчик и его сетевая модель должны быть эквивалентны на каждом шаге исполнения.

Требование 2. (Требование 1) должно исполняться на каждом шаге наблюдаемого исполнения процессов счётчика и модели.

Под вторым требованием понимается, что в результате одинакового воздействия на счётчик и на модель на каждом шаге будет выполняться

Требование 1.

Отношение реализации, заданное этими неформальными требованиями назовем \leq_L .

Определение 18. Отношение реализации $\leq_L \subseteq \mathcal{L} \times \mathcal{L}$ это бинарное отношение такое, что для процессов $p, q \in \mathcal{L}$ выполняется $p \leq_L q$, если

$$I(p) = I(q) \quad (4.13)$$

$$(p \xrightarrow{a} p') \Rightarrow (\forall q' \in \mathcal{L} : (q \xrightarrow{a} q') \Rightarrow (p' \leq_L q')) \quad (4.14)$$

где функция $I : \mathcal{L} \rightarrow 2^A$ задает множество разрешенных действий процесса. В работе [11] представлен спектр отношений эквивалентности и предпорядков для конечно-ветвящихся, непрерывных процессов. Частичный порядок «устанавливает строго больше отождествлений на процессах, чем» задает на отношениях эквивалентности решетку спектра (см. Рис. 1).

Счётчик порождает детерминированные процессы, а для детерминированных процессов спектр фон Глэббика сильно упрощается, что было показано в работе [14]. Но класс сетей Петри являются классом недетерминированных моделей. Отношение реализации должно учитывать возможность недетерминированного поведения модели, так как возможно, что сеть Петри, моделирующая бесконечный счётчик, будет необходимо являться недетерминированной.

Покажем, что отношение реализации является более точным, чем отношения эквивалентности завершённой симуляции (complete simulation) и завершённых трасс (complete trace).

Рассмотрим контрпример (Рис. 2), предложенный в работе [11] для установления неравенства между отношениями эквивалентности завершённой симуляции, завершённых трасс и симуляции готовности (ready simulation). Пусть действие a моделирует операцию декремента, b операцию инкремента, а c операцию 0 . Тогда процесс p на рисунке слева может представлять счётчик в состоянии, когда его значение равно 1, а процесс q на рисунке справа может соответствовать сети Петри моделирующей такой счётчик.



Рис. 1 Спектр линейного-ветвящегося времени.

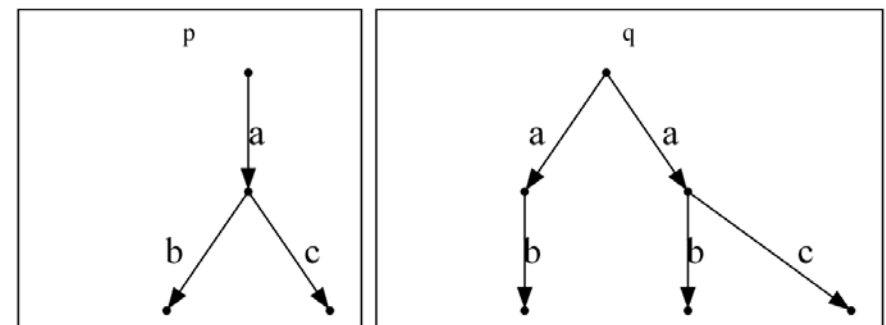


Рис. 2 Процессы эквивалентные относительно отношений завершённых трасс и завершённой симуляции, и неэквивалентные относительно отношения симуляции готовности.

Как было показано в работе [11], эти процессы эквивалентны относительно отношений завершённой симуляции и завершённых трасс. Но процесс q допускает исполнение $\sigma = (a)$, после которого действие c не разрешено. Тогда как процесс p всегда разрешает действие c после исполнения $\sigma = (a)$. Таким образом, нарушается условие (4.14). Из этого следует несоответствие отношения завершённой симуляции и завершённых трасс установленным требованиям к критерию корректности моделирования. Это распространяется на все менее точные отношения, которые лежат ниже эквивалентности завершённой симуляции и завершённых трасс на спектре (см. Рис. 1).

В то же время легко заметить, что определение отношения реализации включает в себя определения отношения симуляции (simulation) и отношение готовности (readiness). Отношение симуляции готовности (ready simulation) является наименьшей верхней гранью этих отношений, относительно порядка «строго больше отождествлений, чем».

Определение 19. Отношение эквивалентности симуляции готовности это бинарное отношение $=_{RS} \subseteq \mathbf{L} \times \mathbf{L}$ на системах помеченных переходов, которое ставит в соответствие процессы $p \in \mathbf{L}$ и $q \in \mathbf{L}$, если

$$I(p) = I(q) \quad (4.15)$$

$$\forall a \in Act : (p \xrightarrow{a} p') \Rightarrow (\exists q' \in \mathbf{L} : q \xrightarrow{a} q' \& p' =_{RS} q') \quad (4.16)$$

$$\forall a \in Act : (q \xrightarrow{a} q') \Rightarrow (\exists p' \in \mathbf{L} : p \xrightarrow{a} p' \& p' =_{RS} q') \quad (4.17)$$

Покажем, что для случая, когда один процесс является детерминированным, отношение реализации \leq_L совпадает с отношением эквивалентности $=_{RS}$.

Утверждение 5. Пусть $p \in \mathbf{L}$ детерминированный процесс, тогда $p \leq_L q \Leftrightarrow p =_{RS} q$.

\Rightarrow : Покажем, что отношение \leq_L является отношением $=_{RS}$. Пусть $p \leq_L q$ и $p \xrightarrow{a} p'$, тогда

$$1). I(p) = I(q)$$

Следует из определения \leq_L .

$$2). \forall a \in Act : (p \xrightarrow{a} p') \Rightarrow (\exists q' \in \mathbf{L} : q \xrightarrow{a} q' \& p' \leq_L q')$$

$$(p \xrightarrow{a} p') \Rightarrow a \in I(p) \Rightarrow a \in I(q) \Rightarrow (\exists q'' : q \xrightarrow{a} q'')$$

$(p \leq_L q) \& (p \xrightarrow{a} p') \& (q \xrightarrow{a} q'') \Rightarrow p' \leq_L q''$ по второму условию определения \leq_L .

$$3). \forall a \in Act : (q \xrightarrow{a} q') \Rightarrow (\exists p' \in \mathbf{L} : p \xrightarrow{a} p' \& p' \leq_L q')$$

$$(q \xrightarrow{a} q') \Rightarrow a \in I(q) \Rightarrow a \in I(p) \Rightarrow (\exists p'' : p \xrightarrow{a} p'')$$

$(p \leq_L q) \& (p \xrightarrow{a} p'') \& (q \xrightarrow{a} q') \Rightarrow p'' \leq_L q'$ по второму условию определения \leq_L .

Таким образом, \leq_L соответствует определению отношения $=_{RS}$.

\Leftarrow : Покажем, что отношение $=_{RS}$ является отношением \leq_L . Пусть $p =_{RS} q$, тогда

$$1). I(p) = I(q)$$

Следует из определения $=_{RS}$.

$$2). \forall a \in Act : (p \xrightarrow{a} p') \rightarrow (\forall q' \in \mathbf{L} : (q \xrightarrow{a} q') \rightarrow (p' =_{RS} q'))$$

Пусть $(p \xrightarrow{a} p')$, тогда

$\forall q' \in \mathbf{L} : (q \xrightarrow{a} q') \rightarrow (\exists p'' : p \xrightarrow{a} p'' \& p'' =_{RS} q')$ по третьему условию определения $=_{RS}$.

$(p \xrightarrow{a} p') \& (p \xrightarrow{a} p'') \Rightarrow (p' = p'') \Rightarrow p' =_{RS} q'$ в силу детерминированности процесса p .

Таким образом, $=_{RS}$ соответствует определению отношения \leq_L . \square

Отношение эквивалентности симуляции готовности $=_{RS}$ удовлетворяет требованиям к отношению реализации формализованным в (Определение 18), и может быть принято в качестве критерия корректности моделирования счётчика сетью Петри.

5. Моделирование счётчиков с помощью сетей Петри

Рассмотрим несколько способов моделирования счётчиков в сетях Петри. Простейшим способом моделирования счётчика является позиция сети Петри. Если поставить в соответствие значение счётчика и значения разметки на позиции, то можно рассматривать данную позицию как примитивный счётчик. Его можно инкрементировать, увеличивая значение разметки на позиции, и аналогично декрементировать. Но в силу монотонности семантики сетей Петри [2] невозможно промоделировать с помощью одной позиции событие, которое срабатывает при заданном значении счётчика, но не срабатывает при большем значении счётчика. Таким образом, в приведенной модели, мы не можем промоделировать действие $0_?$, которое разрешено при нулевом значении счётчика, но запрещено при положительном значении счётчика.

5.1. Необходимое условие моделирования счётчика сетями Петри.

Рассмотрим тривиальный счётчик C_2 с множеством возможных значений $V = \{0, 1\}$ и нулевым начальным состоянием.

$$C_2 = \langle \{0, 1\}, 0 \rangle \quad (5.1)$$

Процессный граф, соответствующий системе помеченных переходов $L_C(C_2)$, представлен на (Рис. 3).

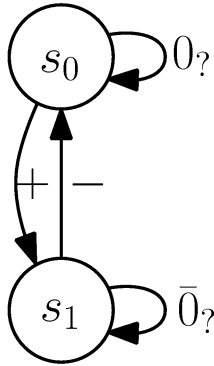


Рис. 3 Процессный граф счётчика с двумя значениями.

Поведение моделирующей сети Петри N_2 должно быть эквивалентно поведению моделируемого счётчика C_2 относительно $=_{RS}$. Состоянию s_0

должно соответствовать некоторое состояние s_{μ_0} , такое что $s_0 =_{RS} s_{\mu_0}$. В силу определения отношения $=_{RS}$ $I(s_0) = I(s_{\mu_0})$. Следовательно, в сети N_2 при разметке μ_0 должны быть активны переходы соответствующие операциям $\{+, 0_?\}$ и пассивны переходы соответствующие операциям $\{-, \bar{0}_?\}$. Т.е.

$$\forall t \in T_{N_2} : (\lambda(t) \in \{+, 0_?\}) \Rightarrow (\bullet F(t) \leq \mu_0) \quad (5.2)$$

$$\forall t \in T_{N_2} : (\lambda(t) \in \{-, \bar{0}_?\}) \Rightarrow (\bullet F(t) \not\leq \mu_0) \quad (5.3)$$

Так как s_1 достижимо из s_0 ($s_0 \xrightarrow{+} s_1$), то, в силу определения отношения $=_{RS}$ (4.16) должно существовать состояние s_{μ_1} , такое что $s_1 =_{RS} s_{\mu_1}$. Т.е. в N_2 должна существовать разметка μ_1 , такая что

$$\forall t \in T_{N_2} : (\lambda(t) \in \{-, \bar{0}_?\}) \Rightarrow (\bullet F(t) \leq \mu_1) \quad (5.4)$$

$$\forall t \in T_{N_2} : (\lambda(t) \in \{+, 0_?\}) \Rightarrow (\bullet F(t) \not\leq \mu_1) \quad (5.5)$$

Из приведенных неравенств следует, что $\mu_1 \not\leq \mu_0$ и $\mu_0 \not\leq \mu_1$, т.е. разметки μ_0 и μ_1 несравнимы. Будем обозначать несравнимость разметок, как $\mu_0 \mid \mu_1$. Таким образом, пространство состояний сети Петри N_2 , моделирующей счётчик с двумя состояниями, будет содержать две несравнимые разметки.

Рассмотрим счётчик конечного объема C_n с n возможными состояниями (Рис. 4).

Утверждение 6. Множество достижимости сети Петри N_n , моделирующей счётчик C_n с n возможными различимыми состояниями, необходимо содержит n несравнимых разметок.

Пусть сеть Петри N_n корректно моделирует счётчик C_n с n различными состояниями $N_n =_{RS} C_n$ и ее множество достижимых разметок не содержит n несравнимых разметок.

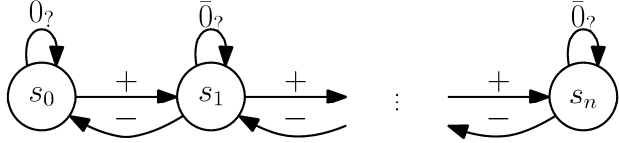


Рис. 4. Процессный граф счётчика с n значениями.

Рассмотрим исполнение счётчика $\sigma = +^{n-1} -^{n-1}$, т.е. сначала $n-1$ раз осуществляется операция инкремента, затем $n-1$ раз осуществляется операция декремента. Счётчик проходит все свои возможные состояния, при этом внутреннее значение счётчика на каждом шаге равно

$$\forall i \in \overline{0, | \sigma |} : \nu(C_n) = \rho(\sigma_i, +) - \rho(\sigma_i, -) \quad (5.6)$$

где σ_i исполнение на i шагу. Следовательно, счётчик принимает нулевые значения на 0 и последнем шаге. Это означает, что только на 0 и последнем шаге предикат 0_i истинен, а в системе помеченных переходов $L_C(C_n)$ действие 0_i разрешено, а $\bar{0}_i$ запрещено. Для промежуточных шагов предикат 0_i ложен, действие 0_i запрещено и $\bar{0}_i$ разрешено.

Т.к. в силу критерия корректности моделирования $L_C(C_n) =_{RS} L_N(N_n)$, то на каждом шаге исполнения множество разрешенных и запрещенных действий процессов систем $L_C(C_n)$ и $L_N(N_n)$ должны совпадать.

Рассмотрим первый отрезок исполнения σ_{n-1} . Пусть нулевому состоянию s_0 системы $L_C(C_n)$ соответствует нулевое состояние s_{μ_0} системы $L_N(N_n)$, т.е. нулевое состояние счётчика моделируется разметкой μ_0 . Отрезок исполнения σ_{n-1} задает n состояний $s_0 \dots s_n$ системы $L_C(C_n)$. Состояниям

$s_0 \dots s_n$ соответствуют состояния $s_{\mu_0} \dots s_{\mu_n}$ системы $L_N(N_n)$, которые в свою очередь имеют прообразами разметки сети Петри $\mu_0 \dots \mu_n$. В силу допущения о том, что множество достижимых разметок N_n не содержит n несравнимых разметок, имеются две разметки μ_1 и μ_2 , такие, что либо $\mu_1 \leq \mu_2$, либо $\mu_1 \geq \mu_2$. Рассмотрим оба случая.

Пусть $\mu_1 \xrightarrow{+l} \mu_2$ и $\mu_1 \leq \mu_2$. На (Рис. 5, а) схематично изображено исполнение сети Петри с такой парой. Допустим $\mu_0 \xrightarrow{+k} \mu_1$. Тогда $\mu_1 \xrightarrow{-k} \mu'_0$ и $\bullet F(t_{0_i}) \leq \mu'_0$. Но, в силу монотонности исполнения сетей Петри, $\mu_1 \leq \mu_2 \Rightarrow (\mu_2 \xrightarrow{-k} \mu_3) \& (\mu'_0 \leq \mu_3)$, т.е. $\bullet F(t_{0_i}) \leq \mu_3$. Так как исполнение $\mu_0 \xrightarrow{\delta} \mu_3$, содержит $l+k$ инкрементов и k декрементов, то в силу (5.6) значение счётчика в этом состоянии будет равняться $\nu(L_C^{-1}(s_3)) = l$, и следовательно действие $\bar{0}_i$ будет разрешено, а $\bullet F(t_{0_i}) \leq \mu_3$. Таким образом, в состоянии s_{μ_3} будут разрешены одновременно действия 0_i и $\bar{0}_i$, однако в состоянии $s_0 \xrightarrow{\delta} s_3$ действие 0_i будет запрещено. Т.е. $0_i \in I(\delta(s_{\mu_0})) \setminus I(\delta(s_0))$ и следовательно $s_0 \not\leq_{RS} s_{\mu_0}$.

Теперь допустим, что $\mu_1 \xrightarrow{+l} \mu_2$ и $\mu_1 \geq \mu_2$. На (Рис. 5, б) схематично изображено исполнение сети Петри с такой парой. Допустим $\mu_0 \xrightarrow{+k} \mu_1$. Тогда $\mu_2 \xrightarrow{-k} \mu_3$ и $\bullet F(t_{0_i}) \leq \mu_3$. Но, в силу монотонности исполнения сетей Петри, $\mu_1 \geq \mu_2 \Rightarrow (\mu_1 \xrightarrow{-k} \mu'_0) \& (\mu'_0 \geq \mu_3)$, т.е. $\bullet F(t_{0_i}) \leq \mu'_0$. Так как исполнение $\mu_0 \xrightarrow{\delta} \mu'_0$, содержит k инкрементов и k декрементов, то в силу (5.6) значение счётчика в этом состоянии будет равняться $\nu(L_C^{-1}(s'_0)) = 0$, и следовательно действие 0_i будет разрешено, а $\bullet F(t_{0_i}) \leq \mu'_0$. Таким образом, в состоянии $s_{\mu'_0}$ будут разрешены

одновременно действия $0_?$ и $\overline{0}_?$, однако в состоянии $s_0 \xrightarrow{\delta} s'_0$ действие $\overline{0}_?$ будет запрещено. Имеем $\overline{0}_? \in I(\mathcal{D}(s_0)) \setminus I(\mathcal{D}(s_{\mu_0}))$ и следовательно $s_0 \not\sim_{RS} s_{\mu_0}$. \square

Утверждение 7. Не существует сети Петри N_{\aleph} , моделирующей счётчик C_{\aleph} со счетным множеством возможных значений.

В силу (Утверждение 6), множество достижимости сети Петри, моделирующей счетчик, должно содержать множество несравнимых разметок, мощности большей или равной мощности возможных значений моделируемого дискретного счётчика.

Если множество возможных значений счётчика счётное, тогда множество достижимости моделирующей его сети Петри N_{\aleph} должно содержать счётное множество несравнимых разметок.

Множество несравнимых разметок является антицепью относительно порядка на разметках сети Петри. Но порядок на разметках сети Петри является правильным в силу (Утверждение 3).

По (Утверждение 1), множество-носитель правильного порядка не может содержать бесконечной антицепи. Таким образом, множество достижимости любой сети Петри и, в частности N_{\aleph} , не может содержать счётного множества несравнимых разметок. \square

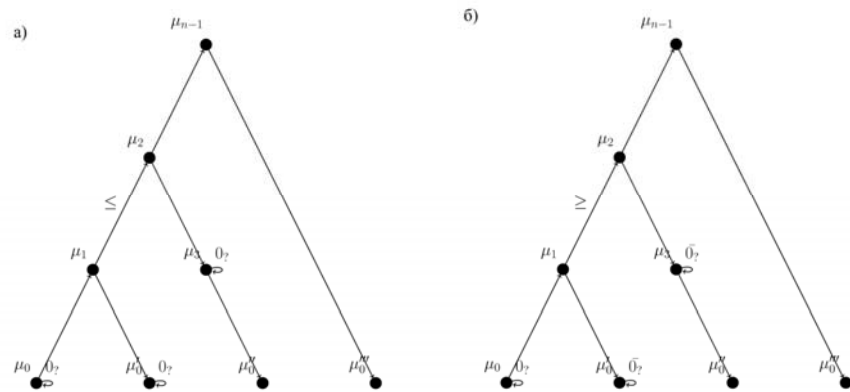


Рис. 5 Схема исполнения модели счётчика с а) возрастающей и б) убывающей парой состояний.

5.2. Модель счётчика конечного объема.

Покажем, что существует сеть Петри с n несравнимыми разметками во множестве достижимых состояний, которая моделирует счётчик с n возможными состояниями.

Это хорошо известная конструкция, которая часто явно и неявно применяется при моделировании процессов с помощью сетей Петри.

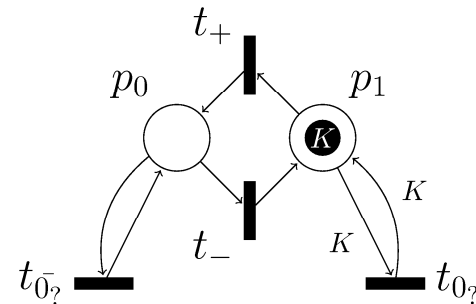


Рис. 6 Модель счётчика конечного объема.

Рассмотрим сеть Петри N_F изображенную на (Рис. 6). Множество позиций состоит из двух элементов $P_F = \{p_0, p_1\}$. Четыре перехода $T = \{t_+, t_-, t_{0_?}, t_{0_?}\}$ моделируют операции счётчика. Начальная разметка $\mu_{F_0} = \langle 0, K \rangle$ определяет $K + 1$ возможных значений счётчика. Множество достижимых состояний такой сети Петри состоит из $K + 1$ разметок $R(N_F) = \{\mu_{F_i} : i \in \overline{0, K} \ \& \ \mu_i = \langle i, K - i \rangle\}$, причем легко заметить, что все разметки несравнимы друг с другом, т.е. имеется $K + 1$ несравнимых разметок. Переход $t_{0_?}$ будет активен только в нулевой разметке $\mu_{F_0} = \langle 0, K \rangle$, т.к. значение любой разметки с индексом не равным нулю, для позиции p_1 равно $K - i$. Переход $t_{0_?}$ не активен при нулевой разметке, т.к. значение разметки на p_0 равно 0 , и активен при любой другой разметке μ_i имеющей на p_0 значение i .

6. Заключение

В данной статье рассмотрены проблемы моделирования счётчиков с помощью сетей Петри. В рамках работы осуществлен выбор критерия корректности моделирования счётчика сетями Петри. Доказано, что для моделирования счётчика с n значениями необходимо n несравнимых разметок сети Петри. Так же показано, что невозможно выразить счётчик с бесконечным счётным числом значений традиционной сетью Петри.

Причиной ограниченности выразительной мощности сетей Петри и подобных сетевых и потоковых моделей является монотонность входной функции по отношению к правильному порядку на разметках сети. Первый способ моделирования - оценка числа несравнимых состояний в пространстве состояний моделируемой системы. Если мощность такого множества конечна, то система выразима на языках сетей Петри и эквивалентна по мощности разрешимости сетям Петри. Если множество различных состояний не конечно, то необходимо использовать конечную аппроксимацию счётчиков с помощью предложенной модели, либо использовать более мощные модели. Однако в последнем случае необходимо учитывать, что теряется мощность разрешимости модели.

Литература

- [1] Decidability questions for bisimilarity of Petri nets and some related problems. **Jancar, Petr.** 1994. Proc. of STACS'94. Т. 775, стр. 581-592.
- [2] **Ломазова И.А.** Вложенные сети Петри: моделирование и анализ распределенных систем с объектной структурой. Москва : Научный мир, 2004.
- [3] **Кузьмин Е. В., Соколов В. А.,** Структурированные системы переходов. Москва : ФИЗМАТЛИТ, 2006.
- [4] **Минский М.** Вычисления и автоматы. Москва : Мир, 1971.
- [5] **Бурдонов И.Б., Косачев А.С., Кулямин В.В.** Теория соответствия для систем с блокировками и разрушениями. Москва : ФИЗМАТЛИТ, 2008.
- [6] **Котов, В. Е.** Сети Петри. Москва : Наука, 1984. стр. 160 стр.
- [7] **Питерсон Дж.** Теория сетей Петри и моделирование систем. Москва : Мир, 1984.
- [8] Тестирование конформности на основе соответствия состояний. **Бурдонов, И. Б. и Косачев, А. С.** Москва : ИСП РАН, 2010 г., Труды Института системного программирования РАН, Т. 18. ISSN 2220-6426 (Online), ISSN 2079-8156 (Print).
- [9] **Roscoe, A. W.** A mathematical theory of communicating processes. 1982.
- [10] Collaborative Software Engineering Using Metamodel-Driven Approach. **Semenov, V.** 16th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE 2007), Paris : IEEE, 2007 г., стр. 178-179.
- [11] The Linear Time - Branching Time Spectrum I. **Glabbeek, R. J.** Berlin : Springer, 1990. Lecture Notes in Computer Science. Т. 458.
- [12] Полное тестирование с открытым состоянием ограниченно недетерминированных систем. **Бурдонов, И. Б. и Косачев, А. С.** Москва : б.н., 2009 г., Труды Института системного программирования РАН, Т. 17, стр. 193-208. ISSN 2220-6426 (Online), ISSN 2079-8156 (Print).

- [13] Contrasting themes in the semantics of imperative concurrency. **J.W. de Bakker, J.N. Kok, J.-J.Ch. Meyer, E.-R. Olderog, J.I.Zucker.** New York : Springer, 1986. Current Trends in Concurrency, LNCS. Т. 224, стр. 51-121.
- [14] Concurrency and automata on infinite sequences. **Park, D.M.R.** [ред.] P. Deussen. London : Springer, 1981. 5th GI Conference, LNCS 104. Т. 104, стр. 167-183.