

УДК 62-50:519.216

АНАЛИЗ СЕТЕЙ ПЕТРИ ПРИ ПОМОЩИ ДЕРЕВЬЕВ ДОСТИЖИМОСТИ*

А.В. МАРКОВ, А.А. ВОЕВОДА

В данной статье описано взаимодействие банкомата и пользователя. Смоделирован режим работы банкомата при помощи визуального языка UML на диаграммах вариантов использования, деятельности и последовательности. Полученные диаграммы транслировались в сеть Петри для оценки корректности их построения с помощью генерации пространства состояний и построения дерева достижимости. Данный пример был выбран, чтобы проследить возможность анализа деревьев по частям. Были выделены два сценария работы, после чего для каждого было построено дерево достижимости. Предложен способ «склеивания» (совмещения) отдельных веток алгоритма в основное дерево достижимости.

Ключевые слова: инженерия ПО, UML-диаграммы, сети Петри, CPN-Tools, пространство состояний, дерево достижимости.

ВВЕДЕНИЕ

В последнее время при создании программного обеспечения (ПО) разработчики стали уделять повышенное внимание этапу проектирования. Тщательная проработка данного этапа дает возможность избежать ошибки и неточности на более поздних стадиях создания ПО. Одним из средств, нацеленных на автоматизацию этапа проектирования, являются графический язык UML и математический аппарат сетей Петри.

UML (англ. Unified Modeling Language – унифицированный язык моделирования) – язык графического описания для объектного моделирования в области разработки программного обеспечения [1]. Стоит отметить, что UML обладает недостаточно формальным синтаксисом, а средства для проверки непротиворечивости поведенческих диаграмм отсутствуют.

Сеть Петри – математический аппарат, который предназначен для моделирования динамических дискретных систем [2]. Представляет собой двухдольный ориентированный граф, который состоит из вершин двух типов – позиций (мест) и переходов, соединенных между собой дугами. Важным замечанием является то, что вершины одного типа не могут быть соединены непосредственно. Позиции могут содержать метки (маркеры), способные пе-

* Статья получена 16 августа 2012 г.

ремещаться по сети. Анализ построенных сетей может дать оценку корректности спроектированного ПО: указать на тупиковые состояния и неиспользуемые сценарии работы. Сложности при анализе могут возникнуть, если алгоритм работы системы слишком разветвлен и имеет огромное количество сценариев, т. е. возможен так называемый «взрыв пространства состояний». В данной работе будет предложен способ избежать затруднений при анализе больших систем на примере взаимодействия банкомата и пользователя. Моделирование системы основывается на работах [4–12].

1 ОПИСАНИЕ СИСТЕМЫ

Современная техника характеризуется высокими темпами ее модернизации и автоматизации. В настоящее время широко распространены банкоматы.

Банкомат – устройство, которое автоматизирует операции по выдаче и переводу денег, хранящихся в банке, лицу, которому они принадлежат [3]. Кредитная карта банка и соответствующий карте PIN-код идентифицируют каждого клиента.

Первое, что требуется от пользователя, – вставить карту. Далее пользователь вводит свой личный PIN-код. Если же PIN-код введен правильно, то пользователю предлагается выполнить одну из следующих операций: просмотреть баланс счета, снятие наличных, пополнение баланса на мобильном номере.

Для наглядного отображения основных аспектов логики взаимодействия банкомата и человека используем визуальный язык моделирования UML и смоделируем диаграмму вариантов использования.

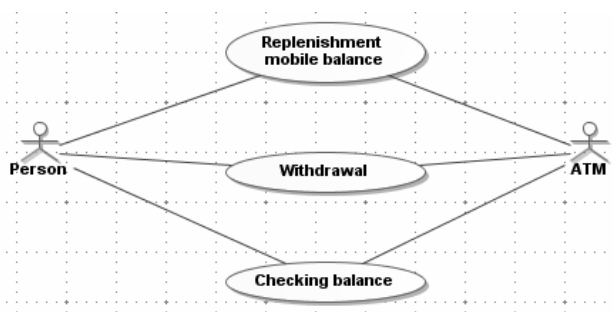


Рис. 1. Диаграмма вариантов использования взаимодействия банкомата и пользователя

На рис. 1 видно, что пользователь может пополнить баланс мобильного номера, снять наличные и проверить баланс при помощи банкомата.

Для более четкого отображения логики разрабатывалась диаграмма деятельности (рис. 2).

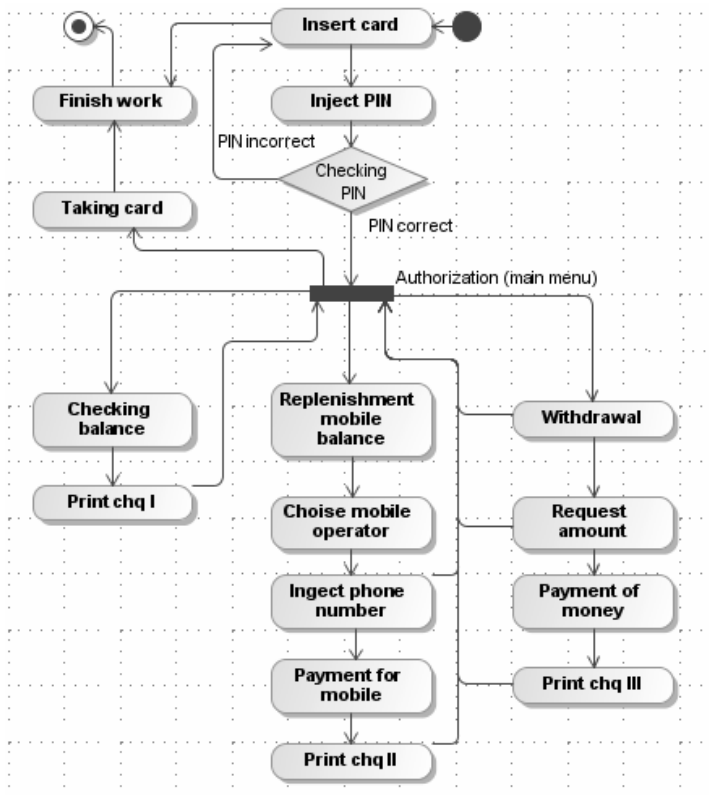


Рис. 2. Диаграмма деятельности взаимодействия банкомата и пользователя

Сперва пользователь вставляет пластиковую карту в банкомат, вводит PIN-код. После принятия PIN-кода банкоматом пользователь попадает в основное меню, где доступны основные функции для интерактивного взаимодействия со своим счетом.

Так как в языке UML отсутствует средство проверки и верификации построенных диаграмм, считается целесообразным использовать математический аппарат сетей Петри.

На рис. 3 представлена сеть Петри, полученная с помощью правил преобразования [4] из диаграммы деятельности взаимодействия банкомата и пользователя. В полученной сети действия с диаграммы деятельности представлены переходами.

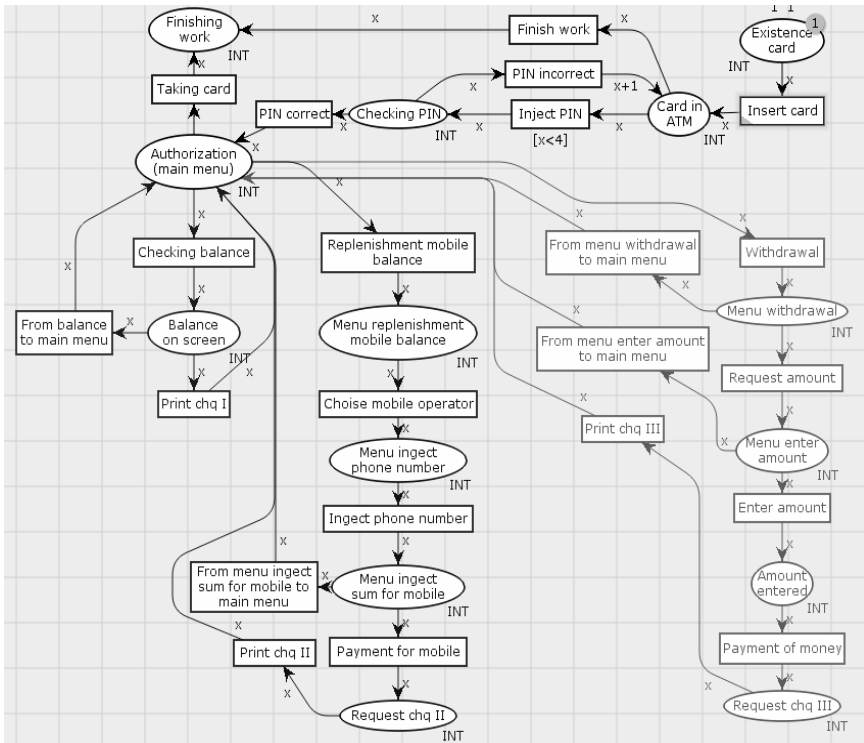


Рис. 3. Сеть Петри взаимодействия банкомата и пользователя

Моделирование сети производилось в программной среде CPN Tools (version 3.5.2, November 2012), которая обладает средством проверки при помощи генерации отчета о пространстве состояний, а также построения дерева достижимости.

В табл. 1 приведены отчет о вычисленном пространстве состояний сети взаимодействия банкомата и пользователя и построенное дерево достижимости (рис. 4).

Таблица 1

Отчет о свойствах сети Петри взаимодействия банкомата и пользователя

Результаты отчета	Выводы
State Space Nodes: 42 Arcs: 65 Secs: 0 Status: Full	Пространство состояний модели вычислено полностью и содержит 42 узлов, 65 дуг, вычислено полностью за 0 секунд.
DeadMarkings [4,8,17,29]	«Мертвые» маркировки в сети [4,8,17,29].
DeadTransitionInstances None	В сети нет «мертвых» переходов.

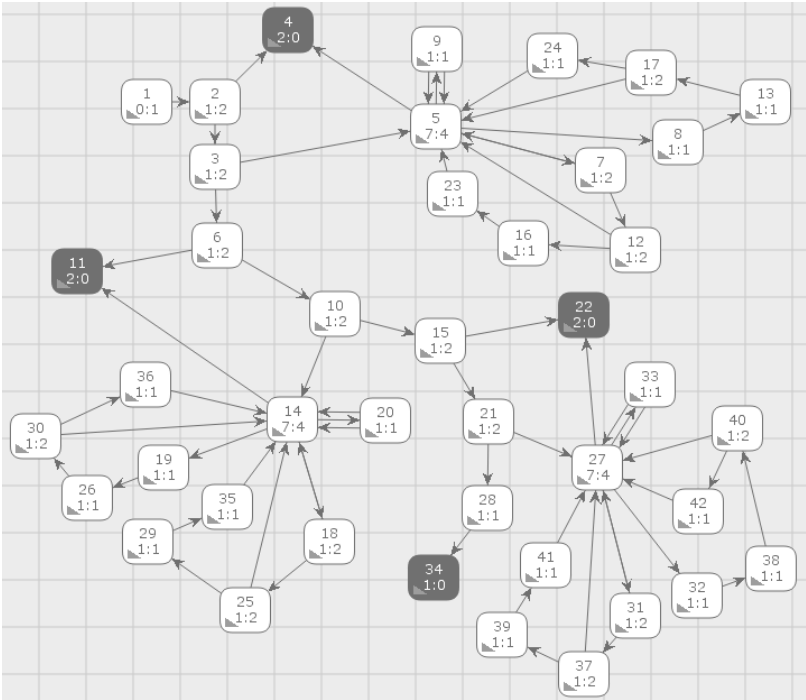


Рис. 4. Дерево достижимости сети взаимодействия банкомата и пользователя

Дерево достижимости разделилось на три похожие ветки. Черным цветом выделены состояния окончания работы алгоритма – попадание фишки в место

Finishingwork. Наличие трех веток сформировалось из-за возможности некорректного ввода PIN-кода (пользователю предоставляется три попытки).

2. ВЫДЕЛЕНИЕ СЦЕНАРИЕВ РАБОТЫ СИСТЕМЫ ИЗ ОСНОВНОГО АЛГОРИТМА

В последнее время рост функциональности и производительности проектированных ПО приводит к усложнению тестирования и верификации.

Проверка с помощью сетей Петри также может поставить разработчиков в затруднительное положение, поскольку при тестировании сложных алгоритмов возможен взрыв пространства состояний, т. е. невозможности оценить дерево достижимости полностью.

Чтобы избежать последствий, вызываемых взрывом пространства состояний, предлагается анализировать деревья частично и делать вывод обо всей сети при помощи оценок о ее частях.

Целесообразно разбить основной алгоритм на несколько возможных сценариев и оценивать каждый.

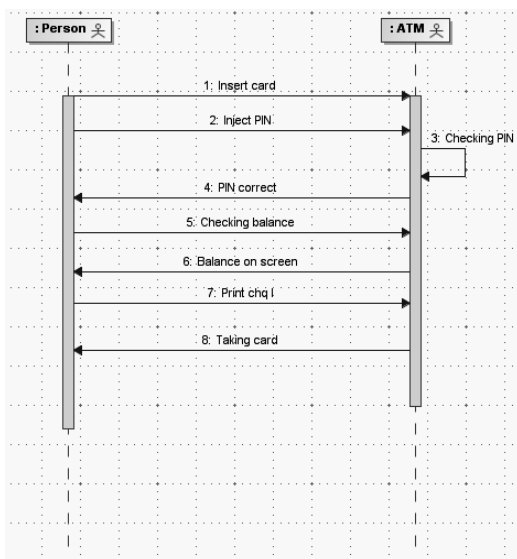


Рис. 5. Диаграмма последовательности сценария проверки баланса

Первым выбран сценарий проверки баланса счета (рис. 5), при котором пользователь безошибочно с первого раза вводит PIN-код, проверяет баланс, печатает чек и забирает пластиковую карту.

Построенная диаграмма была транслирована в сеть Петри (рис. 6), а дерево достижимости для данного сценария выглядит следующим образом (рис. 7).

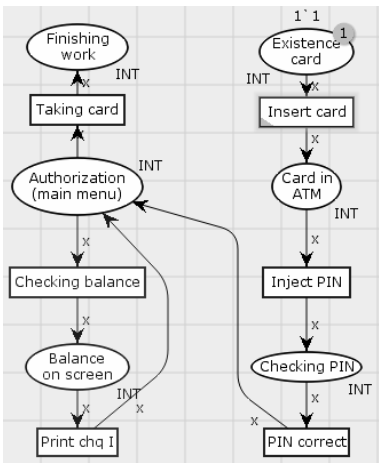


Рис. 6. Сеть Петри сценария проверки баланса

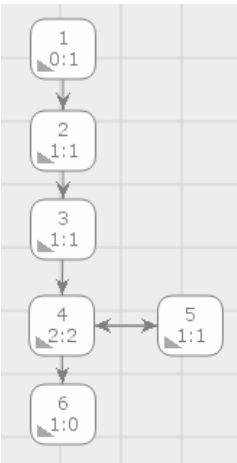


Рис. 7. Дерево достижимости сценария проверки баланса

Данный сценарий можно отследить и на основной сети Петри (рис. 3). С помощью пакета CPNTools проанализируем сеть при помощи генерации пространства состояний и дерева достижимости.

Таблица 2

Отчет о свойствах сети Петри сценария проверки баланса

Результаты отчета	Выводы
State Space Nodes: 6 Arcs: 6 Secs: 0 Status: Full	Пространство состояний модели вычислено полностью и содержит 6 узлов, 6 дуг, высчитано полностью за 0 секунд.
DeadMarkings [5]	«Мертвых» маркировки в сети [5].
DeadTransitionInstances None	В сети нет «мертвых» переходов.

Дерево достижимости имеет 6 состояний. Сеть является безопасной, так как количество фишек в позициях не превышает одну.

Вторым был выбран сценарий снятия наличных из банкомата, при котором пользователь также безошибочно с первого раза вводит PIN-код, выбирает пункт меню «Снятие наличных», вводит необходимую сумму, получает платеж, печатает чек и забирает карту (рис. 8).

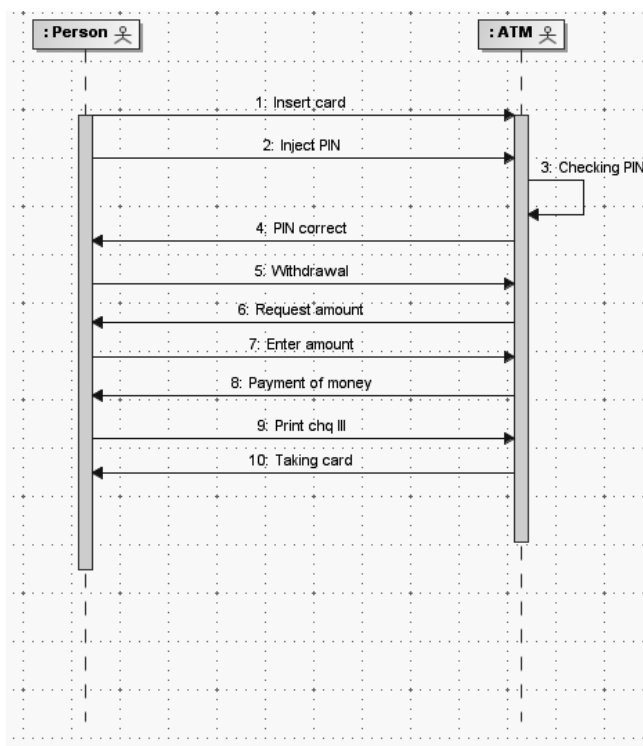


Рис. 8. Диаграмма последовательности сценария снятия наличных со счета

Построенная диаграмма была транслирована в сеть Петри (рис. 9), а дерево достижимости для данного сценария выглядит следующим образом (рис. 10).

Данный сценарий можно отследить и на основной сети Петри (рис. 3). С помощью пакета CPNTools проанализируем сеть при помощи генерации пространства состояний и дерева достижимости.

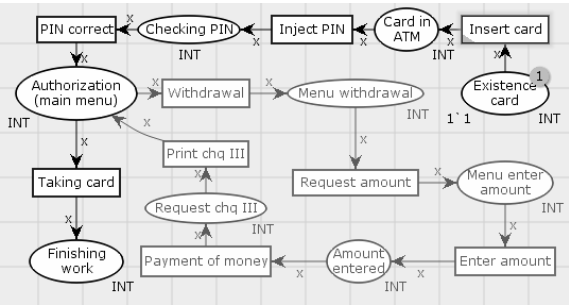


Рис. 9. Сеть Петри сценария снятия наличных со счета

Таблица 3

Отчет о свойствах сети Петри сценария снятия наличных со счета

Результаты отчета	Выводы
State Space Nodes: 9 Arcs: 9 Secs: 0 Status: Full	Пространство состояний модели вычислено полностью и содержит 9 узлов, 9 дуг, вычислено полностью за 0 секунд.
DeadMarkings [5]	«Мертвых» маркировки в сети [5].
DeadTransitionInstances None	В сети нет «мертвых» переходов.

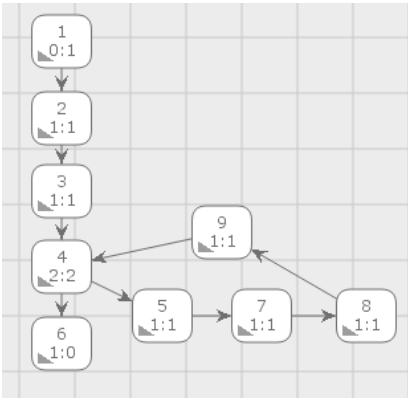


Рис. 10. Дерево достижимости снятия наличных с банкомата

Дерево достижимости имеет 6 состояний. Сеть является безопасной, так как количество фишек в позициях не превышает одну.

Для построения общего дерева достижимости смоделируем общий алгоритм работы двух сценариев. Он представлен на частичной диаграмме деятельности (рис. 11).

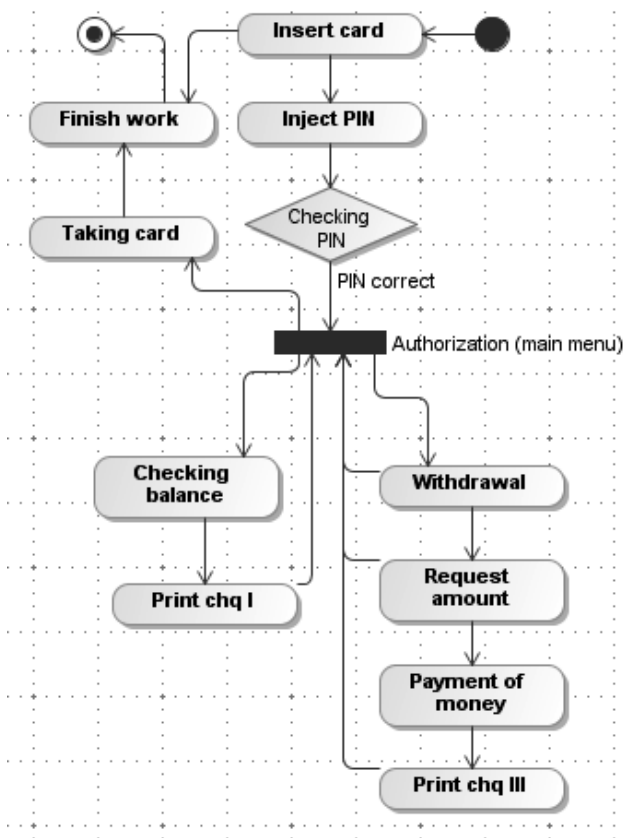


Рис. 11. Диаграмма деятельности выделенных сценариев работы

Состояния используемые в двух сценариях отображаются на диаграмме деятельности. Построенная диаграмма была транслирована в сеть Петри (рис. 12).

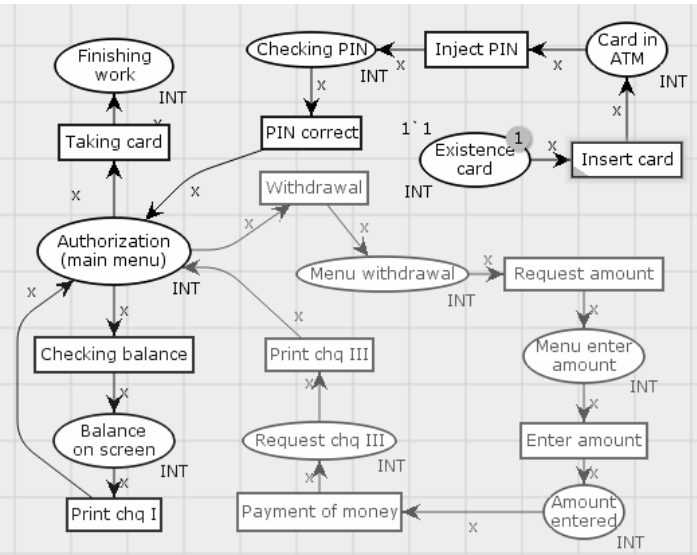


Рис. 12. Сеть Петри выделенных сценариев работы

Алгоритм двух сценариев можно отследить и на основной сети Петри (рис. 3). С помощью пакета CPNTools проанализируем сеть при помощи генерации пространства состояний (табл. 4) и дерева достижимости (рис. 13).

Таблица 4

Отчет о свойствах сети Петри выделенных сценариев работы

Результаты отчета	Выводы
State Space Nodes: 10 Arcs: 11 Secs: 0 Status: Full	Пространство состояний модели вычислено полностью и содержит 10 узлов, 11 дуг, высчитано полностью за 0 секунд.
DeadMarkings [5]	«Мертвых» маркировки в сети [5].
DeadTransitionInstances None	В сети нет «мертвых» переходов.

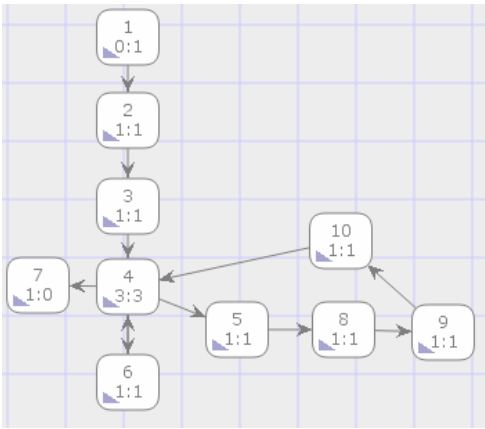


Рис. 13. Дерево достижимости выделенных сценариев работы

Представленное на рис. 13 дерево достижимости было построено при помощи анализа общего алгоритма для двух сценариев. Но основной целью работы является «склеивание» (совмещение) отдельных веток алгоритма в одну конструкцию, т. е. построение деревьев достижимости выбранных сценариев с их последующим соединением.

3. РУЧНОЕ ПОСТРОЕНИЕ ДЕРЕВЬЕВ ДОСТИЖИМОСТИ

Построим вручную дерево с маркировкой для первого сценария работы. В табл. 5 приведены обозначения мест для упрощенной и более наглядной маркировки деревьев, а на рис. 13 представлено дерево достижимости, построенное вручную.

Таблица 5

Обозначение маркировки
для сценария проверки баланса

Existencecard	01
Cardin ATM	02
Checking PIN	03
Authorization	04
Balanceonscreen	05
Finishingwork	06

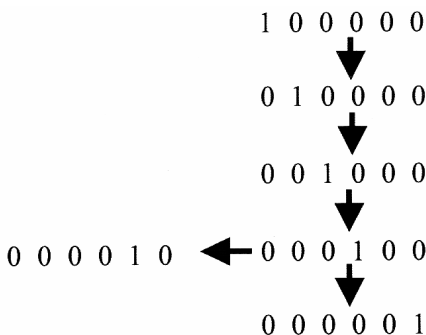


Рис. 14. Построенное вручную дерево ДОС-тижимости для сценария проверки баланса

Следующий шаг – построение дерева достижимости с наглядной маркировкой для второго выбранного сценария работы. В табл. 6 приведено обозначения мест для упрощенной и более наглядной маркировки деревьев, а на рис. 15 представлено дерево достижимости, построенное вручную.

Таблица 6

Обозначение маркировки для сценария снятия наличных со счета

Existencecard	01
Cardin ATM	02
Checking PIN	03
Authorization	04
Menuwithdrawal	05
Finishingwork	06
Menuenteramount	07
Amountentered	08
Requestchq III	09

Если сравнить рис. 14 и 15, видно, что количество состояний в каждом дереве различно, что приводит к невозможности соединить их непосредственно.

Чтобы избежать данных трудностей, предлагается использовать одинаковую размерность деревьев. Добиться этого можно при помощи алгоритма, из которого выделены сценарии. То есть количество состояний для каждого сценария будет равняться количеству состояний алгоритму, из которого они по-

лучены. При отсутствии состояния в данном сценарии будет использоваться пустая маркировка – нулевая. Таким образом, можно сохранить размерность, избежать трудностей.

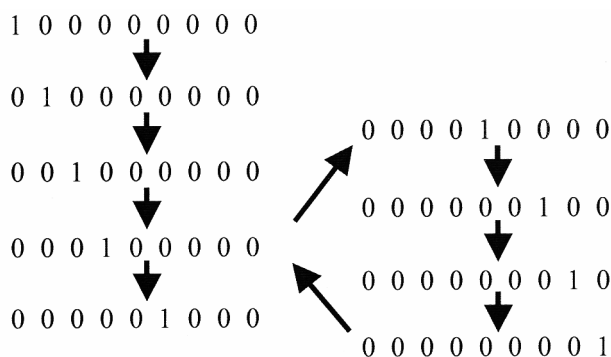


Рис. 15. Построенное вручную дерево достижимости для сценария снятия наличных со счета

В табл. 7 приведено обозначение мест для упрощенной и более понятной маркировки деревьев для двух сценариев работы.

Таблица 7

**Обозначение маркировки
для выделенных сценариев работы**

Existencecard	01
Cardin ATM	02
Checking PIN	03
Authorization	04
Menuwithdrawal	05
Balanceonscreen	06
Finishingwork	07
Menuenteramount	08
Amountentered	09
Requestchq III	10

Придерживаясь табл. 7, составим заново деревья достижимости, представленные на рис. 14 и 15.

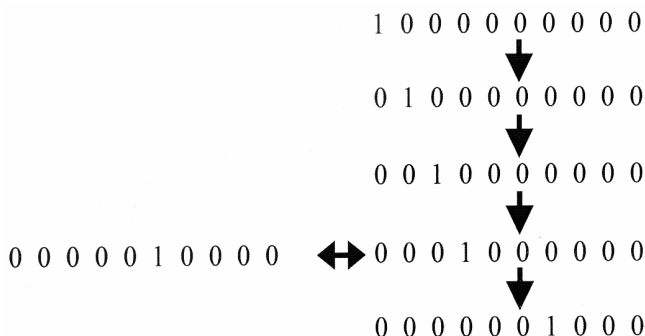


Рис. 16. Построенное вручную дерево достижимости для сценария проверки баланса

В конструкции дерева (рис. 16) не используются маркировки 5, 8–10.

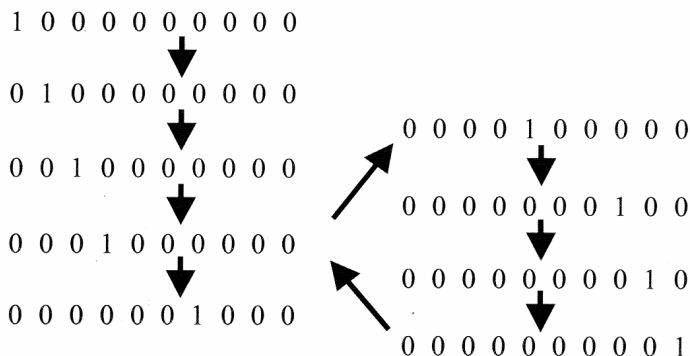


Рис. 17. Построенное вручную дерево достижимости для сценария снятия наличных со счета

В конструкции дерева (рис. 17) не используются маркировка.

Сравнивая рис. 16 и 17, можно увидеть одинаковые состояния, которые являются общими для обоих деревьев и будут совмещены при слиянии возможных разметок сценариев. Общее дерево, построенное вручную, выглядит следующим образом (рис. 18).

- [1] Буч Г. UML. Руководство пользователя / Г. Буч, Д. Рамбо // ДМК, 2001 – 432.
- [2] Питерсон Дж. Теория сетей Петри и моделирование: пер. с англ. / Дж. Питерсон. – М.: Мир, 1984.
- [3] Козлов В.А. Моделирование работы банкомата / В.А. Козлов, О.А. Комалёва // проект. – СПбГУ ИТМО, 2006. – С. 56.
- [4] Коротиков С.В. Применение сетей Петри в разработке программного обеспечения центров дистанционного контроля и управления: дис. ... канд. техн. наук / С.В. Коротиков. – Новосибирск: НГТУ, 2007.
- [5] Воевода А.А. О компактном представлении языков раскрашенных сетей Петри / А.А. Воевода, Д.О. Романников // Сб. науч. тр. НГТУ. – 2008. – № 3(53). – С. 105–108.
- [6] Зимаев И.В. О возможности автоматической трансляции UML-диаграмм деятельности в сети Петри / И.В. Зимаев // Сб. науч. тр. НГТУ. – 2010. – № 1(59). – С. 149–156.
- [7] Воевода А.А. О компактном представлении языков сетей Петри: сети с условиями и временные сети / А.А. Воевода, А.В. Марков // Сб. науч. тр. НГТУ. – 2010. – № 2(60). – С. 77–83.
- [8] Воевода А.А. Тестировании UML-диаграмм с помощью аппарата сетей Петри на примере разработки ПО для игры «Змейка» / А.А. Воевода, А.В. Марков // Сб. науч. тр. НГТУ. – 2010. – № 3(61). – С. 51–61.
- [9] Марков А.В. Описание разрабатываемой системы «Поиск манипулятором кратчайшего пути в лабиринте» / А.В. Марков // Сб. науч. тр. НГТУ. – 2011. – № 3(65). – С. 105–112.
- [10] Романников Д.О. Пример применения методики разработки ПО с использованием UML-диаграмм и сетей Петри / Д.О. Романников, А.В. Марков // Научный вестник НГТУ. – 2012. – № 1(67). – С. 175–181.
- [11] Романников Д.О. Разработка программного обеспечения с применением UML-диаграмм и сетей Петри для систем управления локальным оборудованием: дис. ... канд. техн. наук / Д.О. Романников. – Новосибирск: Изд-во НГТУ, 2012.
- [12] Воевода А.А. Рекурсия в сетях Петри / А.А. Воевода, А.В. Марков // Сб. науч. тр. НГТУ. – 2012. – № 3(69). – С. 115–122.

Марков Александр Владимирович – аспирант кафедры автоматики Новосибирского государственного технического университета. Основное направление научных исследований – исследование свойств сетей Петри и UML-диаграмм. Имеет более 10 публикаций.

E-mail: muviton3@mail.ru.

Воевода Александр Александрович – профессор кафедры автоматики Новосибирского государственного технического университета. Направление научных исследований – теория автоматического управления. Более 200 публикаций по данному направлению.

E-mail: voevoda@ucit.ru.

A.V. Markov A.A. Voevoda

Analysis of Petri nets by trees reachable

This article describes interaction of ATM and user. Simulated mode ATM with a visual language for UML use-case diagrams, activity and sequence. The diagram broadcast Petri net to assess correctness of their design by generating the state space and the construction of the reachability tree. This example was chosen to follow ability to analyze the trees in parts. Have identified two scenarios, and for each tree was constructed reachability. We propose a way to «glue» (combination) of the individual branches of the main tree algorithm reach.

Key words: software engineering, UML-diagrams, Petri nets, CPN Tools, state space, reachability tree.