

УДК 004.896

Н.Н. Войт, Д.С. Канев, С.Ю. Кириллов

МЕТОД ТРАНСЛЯЦИИ ДИАГРАММЫ С ОДНИМ ТИПОМ НАПРАВЛЕННОЙ СВЯЗИ В ИНГИБИТОРНУЮ СЕТЬ ПЕТРИ¹

Войт Николай Николаевич, кандидат технических наук, доцент, окончил факультет информационных систем и технологий Ульяновского государственного технического университета, доцент кафедры «Вычислительная техника» УлГТУ. Имеет более 150 научных статей в области интеллектуальных САПР, Case-, Cals-технологий. Область научных интересов: автоматизация проектирования потоков работ, формальные языки и грамматики, автоматы, контроль и анализ бизнес-процессов, интеллектуальные системы обработки workflows, онтологии, автоматизированные среды обучения [e-mail: n.voit@ulstu.ru].

Канев Дмитрий Сергеевич, кандидат технических наук, окончил факультет информационных систем и технологий УлГТУ. Имеет более 80 статей в области САПР. Область научных интересов: разработка и внедрение программно-аппаратных платформ, способствующих поддержке, интенсификации и повышению вовлеченности обучающихся в образовательный процесс с помощью информационных технологий. [e-mail: dima.kanev@gmail.com].

Кириллов Сергей Юрьевич, окончил факультет информационных систем и технологий УлГТУ, аспирант кафедры «Вычислительная техника» УлГТУ. Имеет более 20 статей в области САПР. Область научных интересов: визуальные языки построения бизнес-процессов и потоков работ, анализ графических диаграмм на наличие синтаксических и семантических ошибок, разработка и внедрение программно-аппаратных платформ, способствующих поддержке, интенсификации и повышению вовлеченности обучающихся в образовательный процесс. [e-mail: kirillovsyu@gmail.com].

Аннотация

Фундаментальная научная проблема теории управления бизнес-процессами заключается в повышении эффективности синтеза и обработки рабочих процессов автоматизированных систем с целью сокращения времени, затрачиваемого на их разработку и повышение качества диаграммных моделей с точки зрения контроля ошибок, сокращения семантического разрыва между анализом и исполнением бизнес-процессов.

Одним из подходов для их верификации является использование формализма сетей Петри, которые позволяют провести семантический и синтаксический анализы моделей, в том числе получить все возможные сценарии выполнения бизнес-процесса, определить никогда не выполняющиеся функции и тупиковые состояния системы, что особенно важно при анализе циклических процессов. При этом существующие методы трансляции графических диаграмм управления бизнес-процессами в сеть Петри являются специализированными и направлены на работу с одним-двумя графическими языками.

В статье предлагается метод перевода диаграмм с одним типом направленной связи в ингибиторную сеть Петри. Показано применение данного метода для трансляции потока управления ЕРС-диаграммы.

Ключевые слова: бизнес-процессы, анализ графических языков, ЕРС, сеть Петри.

doi: 10.35752/1991-2927-2019-3-57-38-45

THE METHOD OF TRANSLATION OF THE DIAGRAM WITH ONE TYPE DIRECTED LINK INTO THE INHIBITOR PETRI NET

Nikolai Nikolaevich Voit, Candidate of Science in Engineering, Associate Professor; graduated from the Faculty of Information Systems and Technologies of Ulyanovsk State Technical University; Associate Professor of the Department of Computer Engineering of Ulyanovsk State Technical University; an author of more than 150 scientific articles in the field of intelligent CAD, Case- and Cals-Technologies; research interests are in the field

¹ Исследования поддержаны грантом Министерства образования и науки Российской Федерации, проект № 2.1615.2017/4.6. Работа выполнена при финансовой поддержке РФФИ в рамках научного проекта № 17-07-01417, а также при финансовой поддержке РФФИ и Ульяновской области в рамках научного проекта № 18-47-730032.

of workflow design automation, formal languages and grammars, automata, control and analysis of business processes, intelligent workflow processing systems, ontologies, automated learning environments. e-mail: n.voit@ulstu.ru.

Dmitrii Sergeevich Kanev, Candidate of Science in Engineering; graduated from the Faculty of Information Systems and Technologies of Ulyanovsk State Technical University; an author of more than 80 articles in the field of CAD; research interests are in the field of development and implementation of software and hardware platforms that support, intensify and increase the involvement of students in the educational process with the help of information technologies. e-mail: dima.kanev@gmail.com.

Sergei Iurevich Kirillov, graduated from the Faculty of Information Systems and Technologies of Ulyanovsk State Technical University, Postgraduate Student at the Department of Computer Engineering of Ulyanovsk State Technical University; an author of more than 20 articles in the field of CAD; research interests are in the field of visual languages of business processes and workflows, analysis of graphic diagrams for syntactic and semantic errors, development and implementation of software and hardware platforms that support, intensify and increase the involvement of students in the educational process. e-mail: kirillovsyu@gmail.com.

Abstract

The fundamental scientific problem of the theory of business process management is to improve the efficiency of synthesis and processing of workflows of automated systems in order to reduce the time spent on their development and improve the quality of diagram models in terms of error control, reduce the semantic gap between analysis and execution of business processes.

One of the approaches for their verification is the use of the Petri nets formalism, which allows to carry out the semantic and syntactic analysis of models. In particular, to get all possible scenarios of the business process, to determine never-running functions and deadlocks of the system, which is especially important in the analysis of cyclic processes. At the same time, the existing methods of translation of graphic diagrams of business process management into the Petri net are specialized and are aimed at working with one / two graphic languages.

The paper proposes a method of transferring charts from one type of directed connections in the inhibitor Petri net. The application of this method for the translation of the control flow of the EPC diagram is shown.

Key words: business processes; graphical language analysis; EPC; Petri net.

ВВЕДЕНИЕ

В современной практике организационного управления широкое распространение получили графические модели бизнес-процессов. Однако исследование графической модели, даже выполненной в соответствии с правилами структурного подхода (ограниченный контекст, ограничение числа элементов на каждом уровне декомпозиции и т. п.), представляет значительную сложность. Поэтому обязательным этапом при моделировании бизнес-процессов предприятий является автоматическая или автоматизированная проверка полученных моделей. Вопросы анализа бездефектного завершения являются актуальными, поскольку сложность моделей постоянно возрастает, а встроенные в среду моделирования средства проверки пока являются далеко не совершенными.

Одним из подходов для их верификации является использование формализма сетей Петри. С момента своего создания в 1962 году сети Петри использовались в самых разных областях применения. Хотя сети Петри являются графическими и простыми для понимания, они имеют формальную семантику и допускают методы анализа: от проверки моделей и структурного анализа до анализа процессов и производительности. Со временем сети Петри стали прочной основой для

исследований в области управления бизнес-процессами. Сети Петри позволяют провести семантический и синтаксический анализ моделей. В том числе получить все возможные сценарии выполнения бизнес-процесса, определить никогда не выполняющиеся функции и тупиковые состояния системы, что особенно важно при анализе циклических процессов. При этом существующие методы трансляции графических диаграмм управления бизнес-процессами в сеть Петри являются специализированными и направлены на работу с одним-двумя графическими языками.

1 Анализ существующих подходов

В статье [1] рассмотрены правила преобразования UML-диаграмм активности в сети Петри и предложен алгоритм их выполнения. Состояние ожидания трансформируется в позицию, состояние действия преобразуется в переход. Разделение и слияние потоков управления, ветвления и условия преобразуется в соответствующую сеть Петри; условия ограничения представлены с помощью условий ограничения переходов.

Практически в каждой разработанной системе можно встретить последовательность и условие, разветвление и слияние. Следовательно, одним из преимуществ описанного набора правил является то, что он содержит

рекомендации по взаимному получению наиболее часто встречающихся элементов в алгоритмах UML-диаграмм активности и сетей Петри. Представление правил в алгоритмическом виде способствует их формализации.

В статье [2] рассматривается методика проектирования программного обеспечения с использованием UML-диаграммы и сети Петри, состоящая из 7 этапов:

1. Составление диаграммы прецедентов с выявлением всех сущностей в виде актёров (actors) и вариантов использования.

2. Составление диаграммы классов и выделение методов класса.

3. Выделение на основании диаграммы классов объектов, которые участвуют в работе системы и помогут определить начальные значения для маркировки всех вершин у сети Петри.

4. Построение диаграммы деятельности для описания динамических свойств системы.

5. Трансляция построенной диаграммы деятельности в сеть Петри при помощи формальных правил преобразования.

6. Анализ свойств при помощи автоматизированных программных пакетов для работы с сетями Петри, позволяющих выявить логические ошибки при моделировании диаграммы деятельности: нахождение неиспользуемых сценариев работы, выявление тупиковых веток алгоритмов и т. п. Для классов или объектов, нуждающихся в детальной проработке их логики, следует составить отдельную диаграмму деятельности и проанализировать её при помощи сетей Петри. При необходимости следует отследить динамику работы по состояниям системы. Для этого можно воспользоваться аппаратом сетей Петри, на основе результатов анализа которого происходит построение и исследование простанства состояний.

7. В случае успешного анализа и при наличии полного набора диаграмм, не нуждающихся в доработке, смоделированные диаграммы готовы для автоматической генерации кода.

В работе [3] даны правила преобразования UML-диаграмм в сеть Петри.

1. Состояние ожидания трансформируется в позицию, а состояние действия преобразуется в переход, который начинает действие. Возможно, что два состояния действия расположены последовательно, тогда окончание первого и начало второго действий совмещают и показывают одним переходом.

2. Разделение и слияния параллельных потоков управления UML-диаграмм преобразуется в соответствующий эквивалент сети Петри.

3. Ветвление на диаграмме деятельности, обозначаемое символом решения, преобразуется в соответствующий эквивалент сети Петри.

4. Условия ограничения показывают с помощью условий ограничения переходов.

5. Критическая секция (доступ к критическому ресурсу, реализуемый с помощью критической секции) преобразуется в позицию с одной меткой в начальной мар-

кировке. При выполнении критической секции метка изымается из позиции и возвращается по завершении критической секции.

6. Семафор, используемый для моделирования занятости конкретного ресурса из пула ресурсов, преобразуется в позицию с тем же количеством меток, которые различаются друг от друга цветами в начальной маркировке. Метка с произвольным индексом должна изыматься из места и возвращаться при завершении ее использования. В простом семафоре должны быть использованы метки одного типа, и их количество в начальной маркировке равно числу ресурсов.

7. К критическим по времени исполнения секциям необходимо добавить позицию-счетчик. Для проверки максимального времени исполнения переходов критической секции нужно добавить защитные условия. Счетчик надо обеспечить переходом с противоположным защитным условием.

8. При взаимном переходе из/в иерархическую сеть Петри необходимо добавить пустые места и соответствующие им переходы, которые соединены последовательно.

Цель работы [4] состоит в описании метода трансляции (H)MSC-диаграмм в сеть Петри и применении метода для формального анализа и верификации свойств таких диаграмм. Результатом работы алгоритма является сеть Петри в формате, совместимом с системой CPN Tools. Эта сеть будет иерархической в случае, если исходная спецификация задана при помощи HMSC-диаграммы или входная диаграмма последовательностей сообщений (MSC) содержала ссылочные выражения.

Диаграммы последовательностей сообщений (MSC-диаграммы) являются популярным языком сценариев, предназначенным для формализации и анализа системных требований на этапе проектирования программного обеспечения. В данной работе для описания объявлений и выражений с данными используется синтаксис языка SDL-2010.

Рассмотрим общее описание алгоритма трансляции. На 1 этапе по входной (H)MSC-диаграмме строится её внутреннее представление, которое будем называть графом частичного порядка (H)MSC. Для каждого события в диаграмме создается узел в графе частичного порядка, который хранит информацию о множестве смежных узлов, идентификаторе процесса и структурной конструкции, к которой принадлежит данное событие.

На 2 этапе выполняется обработка графа частичного порядка. В процессе обработки происходят такие действия, как создание вспомогательных узлов графа (входные и выходные узлы для встроенных выражений MSC), развертка ссылок (построение графов частичного порядка для ссылочных MSC), поиск и обработка альтернатив с нелокальным выбором.

На 3 этапе обработанный граф частичного порядка транслируется в сеть Петри (CPN). Каждому узлу графа соответствует переход в CPN. Каждой дуге, соединяющей два узла в графе частичного порядка, соответствует позиция и две ориентированные дуги, соединяющие

два перехода в CPN. Ориентация построенных дуг сети совпадает с ориентацией дуг графа частичного порядка. Исполнению события в MSC соответствует срабатывание соответствующего перехода в результирующей CPN. Стартовым событиям MSC-диаграммы соответствуют переходы с входными местами *start* с начальной разметкой 1'(). Финальным событиям MSC-диаграммы соответствуют переходы с выходными местами *end*, не содержащие исходящих дуг. Все типы данных и переменные, объявленные в документе MSC, преобразуются в соответствующие типы данных и переменные языка CPN ML.

Каждое событие из базового набора элементов MSC моделируется одним переходом CPN. Согласно стандарту MSC событию приема сообщения предшествует событие его отправки. Каждое сообщение моделируется двумя переходами в CPN. Порядок следования переходов, соответствующих событиям отправки и приема сообщения, соблюдается при помощи места, которое связывает эти переходы и гарантирует правильную очередность между их срабатываниями.

Предикатные (охранные) условия используются для проверки в процессе исполнения диаграммы истинности предиката, заданного на языке данных MSC. Такие условия могут размещаться в начале встроеного выражения или MSC-диаграммы.

Если проверочное условие ложно, то исполнение событий диаграммы, которые следуют после охранного условия, прекращается. В процессе трансляции каждое предикатное условие преобразуется в один переход сети со спусковой функцией, вычисляющей значение установленного предиката.

В статье рассмотрены трансляции базовых элементов MSC, элементов MSC с данными, структурных элементов. Также предложена оценка размера результирующей CPN.

В статье [5] предлагается формальная семантика для отображения нотации моделирования бизнес-процессов (BPMN) в цветные сети Петри. Выбор CPN обусловлен наличием формальных методов верификации, анализа пространства состояний и инвариантов. А концепция цветных маркеров позволяет моделировать данные процесса.

Анализируемые элементы ограничены ядром BPMN и включают следующие типы. Общие элементы, которые используются в нескольких типах диаграмм (например, «Процесс», «Взаимодействие», «Хореография»). Общие элементы предоставляют разработчикам моделей возможность показывать дополнительную информацию о процессе, такую как операции, определение ошибок и ресурсы. Рассмотрена декларация CPN ML, соответствующая стандартной схеме конструкций BPMN 2.0. Эти объявления позволяют преобразовывать элементы BPMN и их атрибуты один в один в сеть Петри. Шлюзы используются для управления потоком действий. В статье используется матрица отображения различных типов шлюзов на соответствующие сети Петри. BPMN-задачи сопоставляются с объявлениями CPN ML. Набор цветов CPN ML создается для каждого типа

задачи BPMN, чтобы охватить все его дополнительные атрибуты. В BPMN у действия могут быть атрибуты, определяющие его дополнительное поведение, такое как зацикливание и параллельные циклы по объектам. В статье предлагается заменить их конструкцией управления потоком выполнения, используя различные типы шлюзов. Для представления подпроцесса BPMN используются иерархические CPN. Каждый подпроцесс представлен как вложенная сеть Петри. Повторно используемая задача – это определение задачи, которая может быть вызвана из любого процесса с помощью операции вызова. В CPN повторно используемая задача и её вызов моделируются с помощью вложенных сетей Петри.

В статье [6] рассматривается только ограниченное подмножество нотации BPMN, используемое для моделирования процессов оркестровки. В качестве узлов на диаграмме выступают объекты потока управления, включающие: операции, логические операторы и события. Отличие предлагаемого подхода в отказе от поведенческой эквивалентности, что позволит упростить метод отображения.

Предложенный подход различает узлы, которые производят изменение объекта управления, приводящее к смене его состояния, и узлы, которые его не меняют, а маршрутизируют. Операции трансформируют объект процесса, они ассоциируются с переходами. Дуги определяют порядок исполнения операций процесса.

Логические операторы ветвления и слияния, такие как «И», «ИЛИ», моделируются эквивалентной сетью Петри. Нотация BPMN допускает «сокращенную» форму записи, когда один графический элемент объединяет сразу операцию и логический оператор. Для анализа такие элементы разворачиваются в отдельные операцию и логический оператор. Циклическое исполнение операции моделируется парой переходов. Первый из них отвечает за выполнение работы, а второй – служит для проверки условия.

Стартовые, завершающие и промежуточные события, размещаемые в потоке, моделируются переходом сети Петри. События, прикрепляемые к границам операций, моделируются логическими операторами «ИЛИ»/«И» в зависимости от того, останавливает ли событие поток управления или нет.

На основе проведенного обзора можно сделать вывод об использовании единого подхода для трансляции различных типов диаграмм в сеть Петри: последовательное преобразование элементов диаграмм в эквивалентные подмножества сети Петри. При этом существующие методы трансляции графических диаграмм являются специализированными и направлены на работу с одним-двумя графическими языками.

2 ТРАНСЛЯЦИЯ ДИАГРАММЫ В ИНГИБИТОРНУЮ СЕТЬ ПЕТРИ

Рассмотрим задачу трансляции некоторой диаграммы в ингибиторную сеть Петри. Входным параметром является диаграмма, модель которой имеет вид [7–12]:

$$G = (V, E, TV, TE),$$

где V – множество вершин,

E – множество связей, $E \subset (V \times V)$

TV – множество типов вершин,

TE – множество типов связей.

Введём ограничение на входные данные: диаграмма может содержать только один тип направленной связи, которая соответствует ребру в сетях Петри.

На выходе получаем ингибиторную сети Петри, модель которой имеет вид:

$$N = (P, T, F),$$

где P – множество позиций,

T – множество переходов,

F – набор дуг, $F \subset (P \times T \times ING) \cup (T \times P \times ING)$,

$ING = \{0, 1\}$ – признак ингибиторной дуги.

Вершины входной диаграммы транслируются в сети Петри, реализующие эквивалентную функцию. Ингибиторная сеть Петри является полной по Тьюрингу, что позволяет реализовать на ней любую вычислимую функцию. Однонаправленные связи диаграммы транслируются в соответствующие дуги сети Петри.

Алгоритм отображения:

Шаг 1. Определение функции трансляции вершины диаграммы в расширенную модель сети Петри.

Для каждого типа диаграммы необходимо определить функцию трансляции вершины диаграммы в расширенную модель сети Петри ($Ftrans: V \rightarrow VN$). Расширенная модель сети Петри имеет вид:

$$VN = (V, N, PIN, TOUT),$$

где V – вершина диаграммы,

N – сеть Петри,

PIN – кортеж входных позиций,

$TOUT$ – кортеж выходных переходов.

Для каждой входящей связи v должна существовать входная позиция в сети Петри $vn[N]$, и для каждой исходящей связи должен существовать переход.

Шаг 2. Формирование множества расширенных сетей Петри.

Для каждой вершины $v \in V$ диаграммы $g \in G$ формируем трансляцию в расширенную сеть Петри $vn \in VN$ и получаем множество AVN :

$$AVN = \{Ftrans(v) | v \in V\}.$$

Шаг 3. Объединение множества расширенных сетей Петри в выходную сеть Петри.

Пусть pn – пустая сеть Петри $(\emptyset, \emptyset, \emptyset)$. Скопируем все дуги, позиции и переходы всех расширенных сетей Петри в pn :

$$pn[P] = \{p | \exists a \in AVN, p \in a[N][P]\},$$

$$pn[T] = \{t | \exists a \in AVN, t \in a[N][T]\},$$

$$pn[F] = \{f | \exists a \in AVN, f \in a[N][F]\}.$$

Шаг 4. Трансляция связей диаграммы.

На основе каждой связи $e \in E$ диаграммы соединяем позиции и переходы выходной сети Петри. Для каждой связи находим начальную и конечную вершины, и соответствующие им расширенные сети Петри $vn1 \in VN, vn2 \in VN$. Из кортежа выходных переходов для $v1$ находим переход, соответствующий выбранной связи, из кортежа входных позиций для $v2$ находим позицию, соответствующую выбранной связи. Соединяем дугой выбранные выходной переход и входную позицию:

$$pn[F] = pn[F] \cup \{(vn_in[TOUT][e],$$

$$vn_out[PIN][e], 0) | \forall e \in E, \exists vn_in \in AVN,$$

$$\exists vn_out \in AVN, vn_in[V] = e1, vn_out[V] = e2\}.$$

Пример

На рисунке показан пример трансляции диаграммы. Сети Петри P1, P2, P3, P4 соответствуют вершинам V1, V2, V3, V4. Соединены они между собой дугами A1, A2, A3, A4, A5, соответствующими связям E1, E2, E3, E4, E5. Для соединения вершин V1 и V2 создана дуга, исходящая из перехода t1 сети P1 и входящая в позицию p1 сети P2.

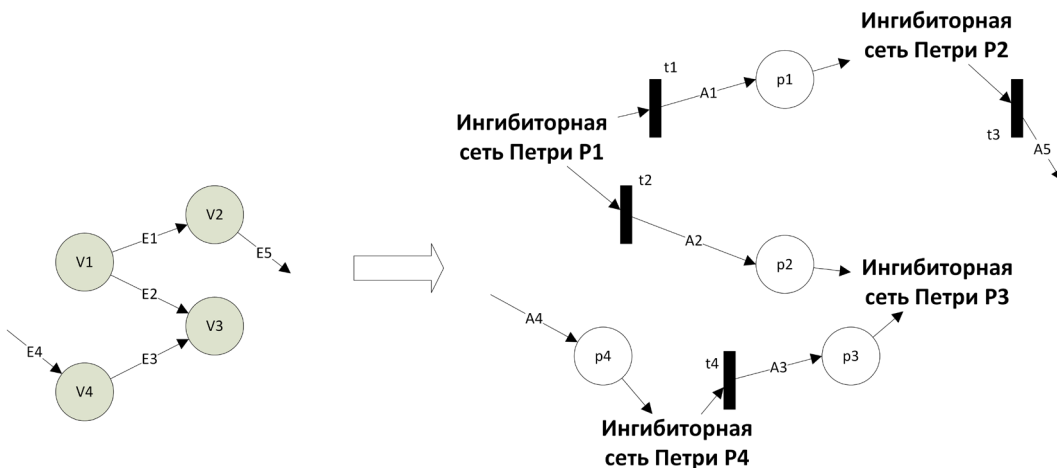


Рис. Трансляция диаграммы в сеть Петри

3 ТРАНСЛЯЦИЯ ПОТОКА УПРАВЛЕНИЯ ЕРС-ДИАГРАММЫ В СЕТИ ПЕТРИ

Событийная цепочка процессов (ЕРС) – тип диаграмм, используемых для моделирования, анализа и реорганизации бизнес-процессов. В то же время ЕРС-диаграммы могут использоваться для моделирования поведения отдельных частей системы при реализации функций и служить заменой традиционных блок-схем (поведенческого моделирования). ЕРС-метод был разработан Августом-Вильгельмом Шеером в начале 1990-х годов.

Рассмотрим трансляцию ЕРС-диаграммы. Модель для нотации ЕРС состоит из множества типов вершин $TV = \{\text{Событие, Функция, Информация, Документ, Файл, Кластер, Набор объектов, Сообщение, Продукт, Организационная единица, Должность, Исполнитель, Местоположение, Приложение, Модуль, AND, OR, XOR, Цель, Термин}\}$ и множества типов связей $TE = \{\text{Поток управления, Организационный поток, Поток ресурсов, Информационный поток, Поток информационных услуг, Поток товарно-материальных ценностей}\}$. Непосредственно за поток управления отвечают следующие типы вершин: Событие, Функция, AND, OR, XOR; и тип связи: Поток управления. Таким образом, поток управления ЕРС-диаграммы подходит под ограничения

одного типа связи. Определим функцию трансляции вершины диаграммы в расширенную модель сети Петри (см. табл.).

Для каждого элемента диаграммы строим сети Петри в соответствии таблицей и соединяем их между собой.

4 РЕАЛИЗАЦИЯ

Предложенный метод реализован на платформе .Net Framework 4.5 для трансляции ЕРС-диаграмм из Microsoft Visio 2017 в сеть Петри для Platform Independent Petri Net Editor.

Алгоритм работы программы состоит из нескольких шагов:

- подключиться к запущенной копии Microsoft Visio;
- считать элементы ЕРС-диаграммы и связи между ними;
- последовательно преобразовать каждый элемент ЕРС-диаграммы в эквивалентную расширенную сеть Петри в соответствии с функцией трансляция вершин диаграммы;
- соединить расширенные сети Петри в единую сеть Петри на основе связей исходной ЕРС-диаграммы;
- сохранить полученную сеть Петри в формате Platform Independent Petri Net Editor;
- показать полученную сеть Петри на экране.

Таблица

Пример трансляции элементов ЕРС-диаграммы в расширенные сети Петри

Элемент	Графический символ	Расширенная сеть Петри
Функция		 $PIN = (P1)$ $TOUT = (T2)$
Оператор OR (слияние)		 $PIN = (P5, P6)$ $TOUT = (T13)$
Оператор XOR (слияние)		 $PIN = (P7, P6)$ $TOUT = (T10)$

Разработанный инструмент имеет следующие возможности: трансляция EPC-диаграмм, построенных в Microsoft Visio, в том числе и выделенной части диаграммы; вывод полученной сети Петри на экран с поддержкой сохранения изображения в файл, автоматического позиционирования элементов, масштабирования и печати; сохранение сети Петри в файл в формате Platform Independent Petri Net Editor. Программа опубликована в сети интернет по адресу: https://lited.ru/?page_id=215.

ЗАКЛЮЧЕНИЕ

В статье рассмотрена задача трансляции различных типов диаграммы в ингибиторную сеть Петри. Изложен универсальный метод трансляции диаграммы с одним типом направленной связи, описаны модели диаграммы, ингибиторной сети Петри и алгоритм трансляции. Показано применение метода для трансляции потока управления EPC-диаграммы, состоящего из 8 типов элементов, в сеть Петри и приведены примеры трансляции различных диаграмм.

СПИСОК ЛИТЕРАТУРЫ

1. Марков А.В., Романиков Д.О. Алгоритм автоматической трансляции диаграммы активности в сеть Петри // Доклады Академии наук высшей школы Российской Федерации. – 2014. – № 1. – С. 104–112.
2. Воевода А.А., Марков А.В. Методика автоматизированного проектирования программного обеспечения функционирования сложных систем на основе совместного использования UML-диаграмм и сетей Петри // Современные технологии. Системный анализ. Моделирование. – 2014. – № 2 (42). – С. 110–115.
3. Марков А.В. Разработка программного обеспечения при совместном использовании UML-диаграмм и сетей Петри (обзор) // Сб. науч. тр. Новосибирского государственного технического университета. – 2013. – № 1. – С. 96–131.
4. Черненко С.А., Непомнящий В.А. Анализ и верификация MSC-диаграмм распределённых систем с помощью раскрашенных сетей Петри // Моделирование и анализ информационных систем. – 2014. – Т. 21, № 6. – С. 94–106.
5. Ramadan M., Elmongui H.G., Hassan R. BPMN formalisation using coloured petri nets // Proceedings of the 2nd GSTF Annual International Conference on Software Engineering & Applications (SEA 2011). 2011. pp. 83–90.
6. Фёдоров И.Г. Метод отображения исполняемой модели бизнес-процесса в сети Петри // Статистика и экономика. – 2013. – № 4. – С. 191–196.
7. Анализ конструкторско-технологических потоков работ в условиях крупного радиотехнического предприятия / А.Н. Афанасьев, Н.Н. Войт, М.Е. Уханова, И.С. Ионова, В.В. Епифанов // Радиотехника. – 2017. – № 6. – С. 49–58. – URL: <https://elibrary.ru/item.asp?id=29826173> (дата обращения: 28.08.2019).
8. Афанасьев А.Н., Войт Н.Н. Интеллектуальная агентная система анализа моделей потоков проект-

ных работ // Автоматизация процессов управления. – 2015. – № 4 (42). – С. 52–61. – URL: <https://elibrary.ru/item.asp?id=25144484> (дата обращения: 28.08.2019).

9. Метакомпилятор RV-грамматик / А.Н. Афанасьев, Н.Н. Войт, Р.Ф. Гайнуллин, С.И. Бригаднов, В.С. Хордов, О.Г. Шаров // Вестник УлГТУ. – 2016. – № 4 (76). – С. 48–52. – URL: <https://elibrary.ru/item.asp?id=28092781> (дата обращения: 28.08.2019).

10. Афанасьев А.Н., Войт Н.Н. Грамматико-алгебраический подход к анализу и синтезу диаграмматических моделей гибридных динамических потоков проектных работ // Информационно-измерительные и управляющие системы. – 2017. – Т. 15, № 12. – С. 69–78. – URL: <https://elibrary.ru/item.asp?id=32465657> (дата обращения: 28.08.2019).

11. Афанасьев А.Н., Войт Н.Н., Уханова М.Е. Контроль и анализ денотативных и сигнификативных семантических ошибок диаграмматических моделей потоков работ в проектировании автоматизированных систем // Радиотехника. – 2018. – № 6. – С. 84–92. – URL: <https://elibrary.ru/item.asp?id=35295882> (дата обращения: 28.08.2019).

12. Войт Н.Н. Методы и средства автоматизации проектирования потоков работ // Информационно-измерительные и управляющие системы. – 2018. – № 11. – С. 84–89.

REFERENCES

1. Markov A.V., Romannikov D.O. Algoritim avtomaticheskoi transliatsii diagrammy aktivnosti v set Petri [Algorithm of Automatic Conversion of the Activity Diagram into Petri-Net Structure Formats]. *Doklady Akademii nauk vysshei shkoly Rossiiskoi Federatsii* [Proceedings of the Russian Higher School Academy of Sciences], 2014, no. 1, pp. 104–112.
2. Voevoda A.A., Markov A.V. Metodika avtomatizirovannogo projektirovaniia programmogo obespecheniia funktsionirovaniia slozhnykh sistem na osnove sovmestnogo ispolzovaniia UML-diagramm i setei Petri [Methodology of Computer-Aided Design Software of Complex Systems Based on Combined Use of UML-Diagrams and Petri Nets]. *Sovremennye tekhnologii. Sistemnyi analiz. Modelirovanie* [Modern Technologies System Analysis Modeling], 2014, no. 2 (42), pp. 110–115.
3. Markov A.V. Razrabotka programmogo obespecheniia pri sovmestnom ispolzovanii UML-diagramm i sete Petri (obzor) [Software Development at the Combined Use of UML-Diagrams and Petri Nets]. *Sb. nauch. tr. Novosibirskogo gosudarstvennogo tekhnicheskogo universiteta* [Proc. of Novosibirsk State Technical University], 2013, no. 1, pp. 96–131.
4. Chernenok S.A., Nepomniashchii V.A. Analiz i verifikatsiia MSC-diagramm raspredelennykh sistem s pomoshchiu raskrashennykh setei Petri [Analysis and Verification of Message Sequence Charts of Distributed Systems with the Help of Coloured Petri Nets]. *Modelirovanie i analiz informatsionnykh sistem* [Modeling and Analysis of Information Systems], 2014, vol. 21, no. 6, pp. 94–106.

5. Ramadan M., Elmongui H.G., Hassan R. BPMN Formalisation Using Coloured Petri Nets. *Proceedings of the 2nd GSTF Annual International Conference on Software Engineering & Applications (SEA 2011)*. 2011, pp. 83–90.
6. Fedorov I.G. Metod otobrazheniia ispolniaemoi modeli biznes-protssessa v seti Petri [Method of Displaying an Executable Business Process Models into Petri Nets]. *Statistika i ekonomika* [Statistics and Economics], 2013, no. 4, pp. 191–196.
7. Afanasev A.N., Voit N.N., Ukhanova M.E., Ionova I.S., Epifanov V.V. Analiz konstruktorsko-tekhnologicheskikh potokov rabot v usloviakh krupnogo radiotekhnicheskogo predpriiatiia [Design Engineering Workflow Analysis at Large Radio Engineering Enterprises]. *Radiotekhnika* [Journal Radioengineering], 2017, no. 6, pp. 49–58. Available at: <https://elibrary.ru/item.asp?id=29826173> (accessed: 28.08.2019).
8. Afanasev A.N., Voit N.N. Intellektualnaia agentnaia sistema analiza modelei potokov proektnykh rabot [Intelligent Agent System for Analyzing Design Workflow Models]. *Avtomatizatsiia protsessov upravleniia* [Automation of Control Processes], 2015, no. 4 (42), pp. 52–61. Available at: <https://elibrary.ru/item.asp?id=25144484> (accessed: 28.08.2019).
9. Afanasev A.N., Voit N.N., Gainullin R.F., S.I. Brigadnov, Khorodov V.S., Sharov O.G. Metakompiliator RV-grammatik [RV-Grammars Metacompiler]. *Vestnik UIGTU* [Bull. of UIGTU], 2016, no. 4 (76), pp. 48–52. Available at: <https://elibrary.ru/item.asp?id=28092781> (accessed: 28.08.2019).
10. Afanasev A.N., Voit N.N. Grammatiko-algebraicheskii podkhod k analizu i sintezu diagrammaticheskikh modelei gibridnykh dinamicheskikh potokov proektnykh rabot [Grammar-Algebraic Approach to the Analysis and Synthesis Diagrammatically Models of Hybrid Dynamic Design Workflows]. *Informatsionno-izmeritelnye i upravliaiushchie sistemy* [Journal Information-Measuring and Control Systems], 2017, vol. 15, no. 12, pp. 69–78. URL: <https://elibrary.ru/item.asp?id=32465657> (accessed: 28.08.2019).
11. Afanasev A.N., Voit N.N., Ukhanova M.E. Kontrol i analiz denotativnykh i signifikativnykh semanticheskikh oshibok diagrammaticheskikh modelei potokov rabot v proektirovanii avtomatizirovannykh system [Control and Analysis of Denotative and Significant Semantic Errors Diagrammatic Models of Design Flows in Designing Automated Systems]. *Radiotekhnika* [Journal Radioengineering], 2018, no. 6, pp. 84–92. Available at: <https://elibrary.ru/item.asp?id=35295882> (accessed: 28.08.2019).
12. Voit N.N. Metody i sredstva avtomatizatsii proektirovaniia potokov rabot [Automated Methods and Software for Designing of Workflows]. *Informatsionno-izmeritelnye i upravliaiushchie sistemy* [Journal Information-Measuring and Control Systems], 2018, no. 11, pp. 84–89.