

УДК 519.876.5

## **МЕТОДИКА ИМИТАЦИОННОГО ОБЪЕКТНО-ОРИЕНТИРОВАННОГО МОДЕЛИРОВАНИЯ АВТОМАТИЗИРОВАННЫХ ПРОИЗВОДСТВЕННЫХ ПРОЦЕССОВ НА БАЗЕ МОДИФИЦИРОВАННЫХ Е-СЕТЕЙ**

*Е. С. Янакова, д-р техн. наук; Д. П. Смирнов*

Национальный исследовательский университет «МИЭТ», Москва, Россия

*Предложена новая методика имитационного объектно-ориентированного моделирования, направленная на эффективное решение задач по описанию и исследованию сложных автоматизированных производственных систем с помощью модифицированных Е-сетей и набора компонентов, представляющих производственные процессы в виде объектов.*

*Ключевые слова:* сети Петри, объектно-ориентированное моделирование, методика, производственные процессы, Е-сети.

На данный момент существует несколько основных подходов к имитационному моделированию (табл. 1). Наиболее подходящим среди них для описания автоматизированных производственных процессов является дискретно-событийный подход [1], поскольку он позволяет учесть и отразить в модели специфику исследуемой области: наличие четких регламентов проводимых работ, нормативы, централизованное управление, а также взаимодействие внутренних объектов.

Среди известных и применяемых математических аппаратов для дискретно-событийного подхода (табл. 2) наибольшими описательными возможностями обладают расширенные сети Петри.

Применение объектно-ориентированного моделирования (ООМ) в дискретно-событийном подходе не до конца исследован. Использование его совместно с модифицированными Е-сетями [2] позволит сократить описание модели и сделать его более наглядным благодаря небольшому базису, обладающему высокими описательными возможностями, наличие простого способа имитации прерывания, необходимого

для описания непредвиденных или приоритетных событий, возникающих в автоматизированных процессах, а также способности представления процессов в виде набора связанных объектов.

Для разработки методики имитационного ООМ необходимо определить набор основных компонентов (табл. 3).

Базовым компонентом моделирования является класс. Он включает в себя описание некоторого обособленного процесса с помощью элементарных сетей, связанных между собой общими позициями, и представляет собой структуру, состоящую из набора компонентов, взаимодействующих между собой. Так же, как и в объектно-ориентированном программировании, класс представляет собой шаблон для создаваемых от него объектов (экземпляров класса). При этом каждый объект является самостоятельной структурой. Количество объектов теоретически может быть неограниченным и каждому из них можно задать параметры, влияющие на описываемый им процесс. Набор этих параметров определяется классом, экземпляром которого является объект. В состав класса входят структурные, коммутативные и программные компоненты.

Помимо компонентов, располагающихся внутри класса, есть два компонента, которые взаимодействуют со всеми объектами: глобальная переменная и метка.

Связь между компонентами класса можно представить в виде соответствующей диаграммы (рис. 1). Поскольку класс является абстрактной структурой, поэтому внутренние и внешние взаимодействия происходят уже непосредственно в экземплярах класса.

---

**Янакова Елена Сергеевна**, профессор кафедры "Информатика и программное обеспечение вычислительных систем".

E-mail: helen@elvees.com

**Смирнов Дмитрий Петрович**, аспирант кафедры "Информатика и программное обеспечение вычислительных систем".

E-mail: smirnovdp@gmail.com

---

*Статья поступила в редакцию 14 марта 2016 г.*

© Янакова Е. С., Смирнов Д. П., 2016

Таблица 1

Сравнение основных подходов имитационного моделирования

Характеристики	Системно-динамический подход	Дискретно-событийный подход	Агентный подход
Способ представления системы	Совокупность переменных, соединенных между собой петлями обратной связи	Набор связанных между собой событий, происходящих в определенные моменты времени	Множество самостоятельных агентов с определенными параметрами и поведением
Время протекания процессов	Непрерывное	Дискретное	Дискретное
Уровень детализации описываемой модели	Низкий	Высокий, средний	Высокий, средний, низкий
Основное применение	Изучение поведения сложных систем на большом промежутке времени	Моделирование систем с четко выраженными событиями	Исследование децентрализованных систем

Таблица 2

Сравнительный анализ типов математических аппаратов для описания бизнес-процессов

Характеристики	Конечные автоматы	Стохастические сети	Сети Петри	Расширенные сети Петри
Возможность описания последовательного алгоритма действий	Есть	Есть	Есть	Есть
Наличие атрибутов (параметров) у динамических объектов	Нет	Есть	Нет	Есть
Описание параллельных процессов и их взаимодействий	Нет	Нет	Есть	Есть
Учет времени выполнения процесса	Нет	Есть	Нет	Есть
Наличие вероятностных функций возникновения события	Нет	Есть	Нет	Есть
Иерархическое представление системы	Нет	Нет	Нет	Есть

Таблица 3

Основные типы компонентов объектно-ориентированного моделирования

Название компонента	Разновидности	Описание
<i>Глобальные компоненты</i>		
Класс	—	Абстрактный компонент, включающий набор вспомогательных компонентов и описывающий некий реальный процесс
Глобальная переменная	Целочисленная, вещественная, текстовая и логическая	Изменяемый параметр, применяемый для получения общих данных о моделируемой системе или для взаимодействия процессов между собой
Метка	—	Динамический компонент, перемещаемый по расширенной сети Петри
<i>Структурные компоненты класса</i>		
Позиция	Простая, множественная	Компонент математического аппарата E-сети, представляющий место хранения меток, определяющее некоторое фиксированное состояние процесса в моделируемой системе
Переход	$T, X, Y, G$ и $I$	Компонент математического аппарата E-сети, определяющий условие срабатывания перехода
Объект	—	Экземпляр класса, представляющий собой самостоятельный процесс

Название компонента	Разновидности	Описание
<i>Коммутативные компоненты класса</i>		
Список связей	—	Список связи позиций и переходов, а также элементарных сетей с объектами других классов
Порт	Входной, выходной, установки начального значения	Позиция, используемая для представления экземпляра класса в качестве структурного компонента или задания начального набора меток
Условия перехода	Для входных позиций, для выходных позиций	Набор условий по выбору позиций, используемых для перехода в элементарной сети
<i>Программные компоненты класса</i>		
Локальная переменная	Целочисленная, вещественная, текстовая и логическая	Изменяемый параметр, применяемый для взаимодействия напрямую не связанных между собой процессов
Метод преобразования данных	—	Назначаемый элементарной сети набор команд, изменяющий атрибуты меток и переменные, доступные объекту
Параметр экземпляра	Целочисленный, вещественный, текстовый и логический	Представляет собой локальную переменную с возможностью задать начальное значение при определении экземпляра класса

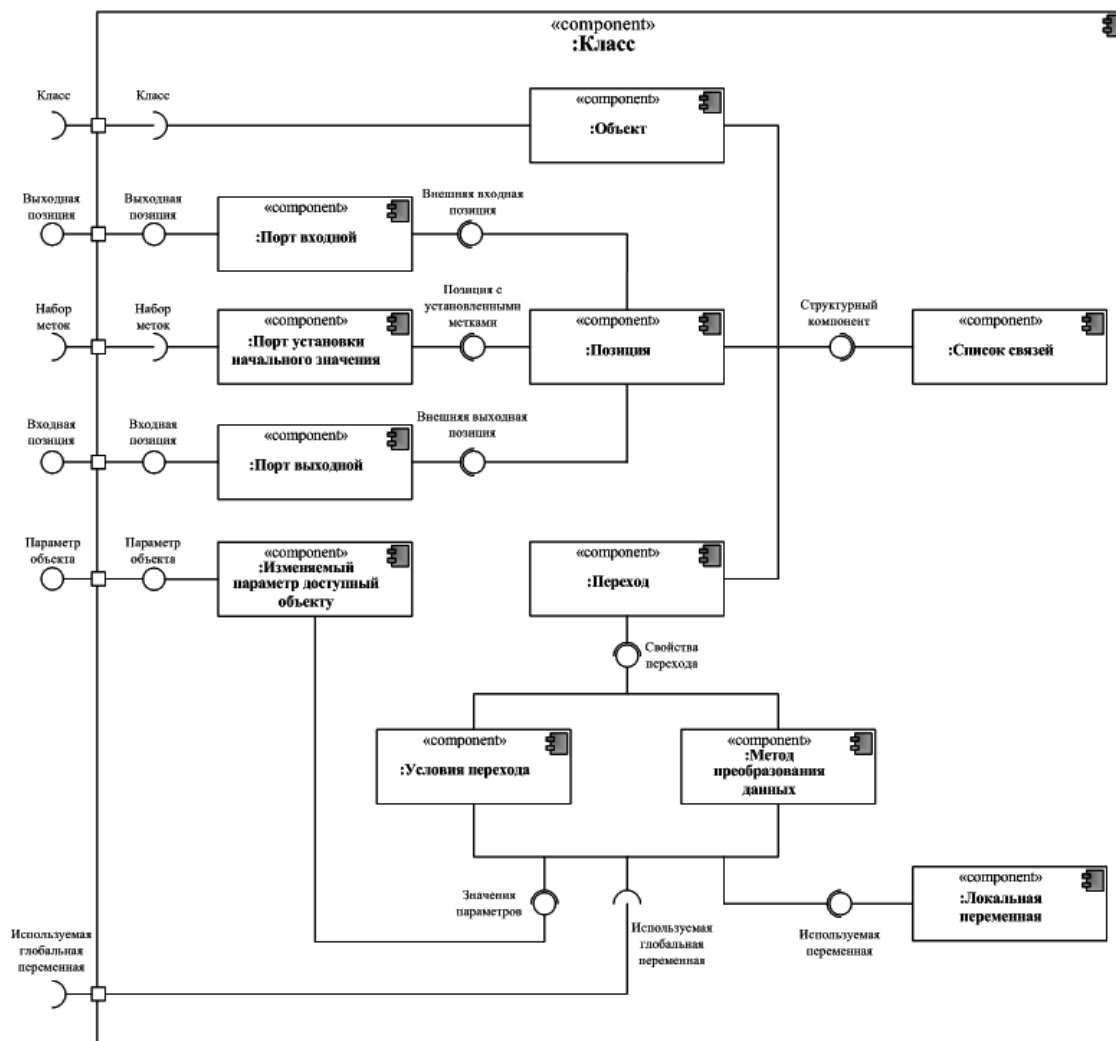


Рис. 1. Представление класса в виде диаграммы компонентов UML

Разработанный набор компонентов позволяет реализовать механизм наследования, давая возможность описывать новые классы на основе уже существующих. Таким образом, появляется возможность делать описание модели более логичным и менее нагруженным повторяющейся информацией. Полиморфизм в данном наборе компонентов выражен способностью метода преобразования данных обрабатывать входную информацию в зависимости от количества и типов атрибутов метки, проходящей через переход, а также в способности объектов с помощью параметров менять функционирование процесса, описываемого классом. Инкапсуляция проявляется в способности класса скрывать в себе описание расширенной сети Петри и внутреннее ее функционирование. Каждый класс представляет собой законченный процесс, давая возможность влиять на внутреннюю структуру только на уровне задания параметров объекту.

Для описания организации связей между структурными компонентами ООМ, можно представить класс в виде структуры:

$$C = (T, P, E, L, V, S, M, U),$$

где  $T$  — множество переходов;

$P$  — множество позиций;

$E$  — множество входящих экземпляров других классов;

$L$  — список связей;

$V$  — переменные и доступные параметры экземпляра данного класса;

$S$  — порты (входные, выходные, задания начальных значений);

$M$  — методы преобразования данных;

$U$  — условия перехода.

Список связей включает в себя входные и выходные функции, которые определяют связи между структурными компонентами. Их можно представить следующим образом:

$$I(t) = P_i, O(t) = P_j,$$

где  $t \in T$ ,  $P_i$  — входная функция;

$$P_j \subseteq P, I$$

$O$  — выходная функция.

Используемый математический аппарат предполагает не более одного входа и не более одного выхода из позиции. В связи с этим имеются следующие ограничения:

$$I(t_i) \cap I(t_j) = \emptyset, O(t_i) \cap O(t_j) = \emptyset,$$

где  $i \neq j$ ;  $i, j = 1, 2, \dots, k$ ;  $k = |T|$  — число переходов.

Позиции в разных классах могут именоваться одинаково, и от одного класса может быть создано несколько экземпляров, поэтому доступ к позициям вне экземпляра осуществляется через его имя. В случае, когда один экземпляр входит в состав другого, обращение к позиции производится через их перечисление:  $p_1 : e_1 : e_2$ . Здесь объект  $e_1$  содержит позицию  $p_1$  и входит в состав  $e_2$ .

Связь элементарных сетей между собой происходит за счет совместного использования позиций. Одна и та же позиция является входом у одной элементарной сети и выходом у другой. Объекты так же, как и элементарные сети, связываются с помощью общих позиций. Однако доступны для взаимодействия только позиции, отмеченные в виде портов.

Механизм взаимодействия можно представить на примере двух классов:

$$C_1 = (T_1, P_1, E_1, L_1, V_1, S_1, M_1, U_1),$$

где  $I_1, O_1 \in L_1$ , и

$$C_2 = (T_2, P_2, E_2, L_2, V_2, S_2, M_2, U_2),$$

где  $I_2, O_2 \in L_2$ .

На рис. 2 изображена модель, состоящая из последовательно соединенных объектов. Используемые объекты в данной модели принадлежат соответствующим классам  $e_1 \in C_1$ ,  $e_2 \in C_2$ , которые описываются следующими составляющими:  $T_1 = \{t_1\}$ ,  $P_1 = \{p_1, p_2, p_3\}$ ,  $I_1(t_1) = \{p_1, p_2\}$ ,  $O_1(t_1) = \{p_3\}$ ,  $E_1 = \{\}$ ;  $T_2 = \{t_2\}$ ,  $P_2 = \{p_4, p_5, p_6, p_7\}$ ,  $I_2(t_2) = \{p_4, p_5\}$ ,  $O_2(t_2) = \{p_6, p_7\}$ ,  $E_2 = \{\}$ . У первого класса определен выходной порт  $S_1(Out) = \{p_3\}$ , а у второго класса входной порт  $S_2(In) = \{p_4\}$ . Связь их экземпляров будет определена с помощью общей позиции  $u \sim p_4 : e_2 \sim p_3 : e_1$ . Внутри общего класса входные и выходные функции данных объектов можно переопределить следующим образом:  $O_1(t_1) = \{u\}$ ,  $I_2(t_2) = \{u, p_5\}$ .

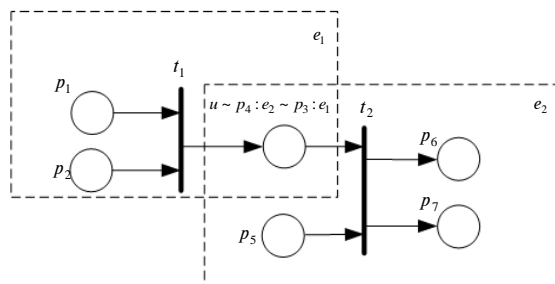


Рис. 2. Модель из последовательно соединенных объектов

На рис. 3 рассмотрена ситуация, когда объект  $e \in C_2$  входит в состав класса  $C_1$ . Классы описываются следующими составляющими:  $T_1 = \{t_1, t_2\}$ ,  $P_1 = \{p_1, p_2, p_3, p_4, p_5\}$ ,  $I_1(t_1) = \{p_1, p_2\}$ ,  $O_1(t_1) = \{p_3\}$ ,  $I_1(t_2) = \{p_4\}$ ,  $O_1(t_2) = \{p_5\}$ ,  $E_1 = \{e\}$ ;  $T_2 = \{t_3\}$ ,  $P_2 = \{p_6, p_7, p_8, p_9\}$ ,  $I_2(t_3) = \{p_6, p_7\}$ ,  $O_2(t_3) = \{p_8, p_9\}$ ,  $E_2 = \{e\}$ . В классе  $C_2$  определены входные и выходные порты:  $S_2(In) = \{p_6\}$  и  $S_2(Out) = \{p_8\}$ . Связь между объектом  $e$  и классом  $C_1$  организована с помощью следующих общих позиций  $u_1 \sim p_6 : e \sim p_3 : C_1$  и  $u_2 \sim p_8 : e \sim p_4 : C_1$ . Входные и выходные функции данных сущностей можно переопределить:  $O_1(t_1) = \{u_1\}$ ,  $I_1(t_2) = \{u_2\}$ ,  $I_2(t_3) = \{u_1, p_6\}$ ,  $O_2(t_3) = \{u_1, p_9\}$ .

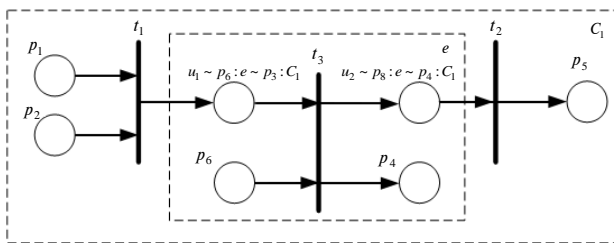


Рис. 3. Модель из класса и входящего в него объекта

При наличии входного и выходного порта объекты можно последовательно соединять друг с другом. Начальной точкой построения модели является класс *Main*. Его основной задачей является объединение процессов в конечную модель.

Методика имитационного ООМ включает в себя **семь основных этапов**.

На **первом этапе** определяется задача исследования и проводится анализ исследуемого объекта, представляющего собой процесс или систему. Результатом данного анализа является выявление набора производимых действий, определение их взаимосвязи между собой и зависимостей от различных динамических объектов (документы, информация, материальные ресурсы и т. п.), а также определение последовательности их выполнения.

На **втором этапе** создается формальное описание исследуемого объекта, для чего применяется алгоритм построения объектно-ориентированной модели, который использует разработанные компоненты моделирования, чтобы представлять исследуемые объекты в виде расширенных сетей Петри. Алгоритм включает в себя набор последовательных действий. Разбие-

ние исследуемого объекта на процессы. Например, если рассматривать функционирование некоторого производственного предприятия, то его работу можно разбить на отдельные процессы: взаимодействие с поставщиками, складское хранение, производство, доставка и т. д.

1. Представление процессов в виде взаимодействующих между собой экземпляров класса. В случае, если процессы повторяются или имеют небольшие различия, они могут быть описаны с помощью одного класса либо через наследование общего базового класса. Результатом выполнения данного пункта является общее описание взаимодействий между классами с помощью их экземпляров и глобальных переменных, где для каждого экземпляра определено, что поступает на вход и какие динамические объекты формируются по завершению процесса.

2. Определение действий, производимых в процессах исследуемого объекта. Например, в складском хранении это может быть: приемка товара, организация хранения, выдача и т. д.

3. В случае, если все полученные действия являются элементарными относительно данного исследования, т. е. не требуют большей детализации, тогда выполняется следующий пункт, иначе найденное не элементарное действие представляется в виде процесса исследуемого объекта, и алгоритм переходит к выполнению пункта 2.

4. Представление действий в качестве элементарных сетей, состоящих из переходов и входных и выходных позиций. При этом входные и выходные позиции представляют необходимые условия для их выполнения, к которым относятся наличие различных запросов, материальных и информационных ресурсов, а также наличие свободных мест для полученных результатов.

5. Определение локальных переменных, а также, с какими атрибутами используются метки в каждом классе. Для выполнения данного пункта необходимо определить какими характеристиками обладают динамические объекты, представляемые в модели в виде меток, а также проанализировать зависимости, влияющие на выполнения каждого действия. По полученной информации определяются для каждой элементарной сети атрибуты для меток, а также переменные для организации различных дополнительных условий и связей. Следует учитывать, что связанные между собой экземпляры класса в местах соединения должны принимать метки с одинаковыми атрибутами.

6. Определение в элементарных сетях методов преобразования данных и условий соверше-

ния перехода (действия). Составляющие данные позволяют описать условия, при которых действие совершается, и то, что в результате него получается.

7. Указание входных, выходных портов и портов установки начальных значений. Определяются позиции, в которых будут устанавливаться метки, представляющие собой динамические объекты, перед началом моделирования, а также позиции, с помощью которых экземпляры классов будут связываться и передавать информацию в виде меток.

8. Связывание элементарных сетей и экземпляров класса между собой посредством способа организации связей компонентов в ООМ и объединение их в единую модель с помощью класса *Main*.

9. Составление спецификации используемых в модели атрибутов меток, возможные места расстановки меток и описание глобальных переменных.

В результате выполнения алгоритма создается модель, которая включает в себя следующие составляющие: набор классов и связанных между собой объектов, объединенных корневым классом *Main*; глобальные переменные; спецификацию используемых атрибутов меток и возможные места начальной расстановки меток.

На **третьем этапе** методики для возможности проведения имитационных экспериментов с помощью программного обеспечения данная модель описывается с помощью предметно-ориентированного языка моделирования.

На **четвертом этапе** определяются группы начальных значений для исследуемого объекта. К этим значениям относятся: начальное расположение динамических объектов в модели, представляемых метками, определение общих параметров исследуемого объекта с помощью

глобальных переменных, условия остановки моделирования и определение мест в модели для сбора статистики.

На **пятом этапе** в специализированном программном обеспечении производится отладка модели с помощью проведения предварительных экспериментов над каждым экземпляром класса моделируемого объекта с использованием тестовых начальных значений. В основном проверяется соответствие модели исследуемому объекту.

На **шестом этапе** проводятся имитационные эксперименты с помощью специализированного программного обеспечения по определенным ранее начальным значениям. Результатом его является набор таблиц значений и графиков зависимостей по различным исследуемым параметрам.

На **седьмом этапе** формируется отчет по полученным во время экспериментов результатам, и составляется заключение по поставленной задаче.

### Заключение

Разработана методика имитационного объектно-ориентированного моделирования на базе модифицированных *E*-сетей, позволяющая описывать автоматизированные производственные процессы в виде групп связанных элементарных сетей, представляющих собой экземпляры объектов, и отличающаяся новым подходом к проектированию моделей. Для ее применения разработан соответствующий язык описания [3] и программное обеспечение для проведения имитационных экспериментов. Данную методику возможно использовать в различных существующих средствах имитационного моделирования, таких как *Anylogic*, создав соответствующие библиотечные элементы.

### ЛИТЕРАТУРА

1. Ходырев И. А. Математическое моделирование динамики показателей деятельности предприятия на основе журналов событий информационных систем: Дис. канд. техн. наук: 05.13.18. — Санкт-Петербург, 2014.
2. Костин А. Е., Илюшечкина Л. В. Модифицированные *E*-сети для исследования систем распределенной обработки информации // Автоматика и вычислительная техника. 1988. № 6. С. 27—35.
3. Смирнов Д. П. Способ моделирования распределенных автоматизированных процессов и производств // Фундаментальные исследования. 2016. № 2. С. 94—98.

## **METHODS OF OBJECT-ORIENTED SIMULATION MODELING OF AUTOMATED PRODUCTION PROCESSES BASED ON EXTENDED PETRI NETS**

*E. S. Yanakova, D. P. Smirnov*

National Research University of Electronic Technology "MIET", Moscow, Russia

*The article suggests a new method of object-oriented simulation which is aimed at the effective solution of problems on the description and research of complex automated production systems through of the modified E-network and a set of components that represent business processes as objects.*

*Keywords:* Petri nets, object-oriented modeling, methods, manufacturing processes, E-nets.

Bibliography — 3 references.

*Received March 14, 2016*

\* \* \*