

УДК 65.012.122

Е.А. Ланцев, М.Г. Доррер

ПОЛУЧЕНИЕ АГЕНТНОЙ ИМИТАЦИОННОЙ МОДЕЛИ ИЗ ДИСКРЕТНО-СОБЫТИЙНОГО ОПИСАНИЯ БИЗНЕС-ПРОЦЕССА

Y.A. Lantsev, M.G. Dorrer

CREATING AGENT-BASED MODEL FROM THE BUSINESS PROCESS DISCRETE-EVENT MODEL

Предложена методика трансляции модели из нотации ARIS eEPC в агентную имитационную модель AnyLogic. Поставлен эксперимент по трансляции модели описанной нотацией eEPC в агентную имитационную модель AnyLogic. Результаты работы могут использоваться аналитиками для сокращения времени проектирования в процессе реализации имитационных моделей, а также упрощения процесса последующей эксплуатации имитационных моделей.

АГЕНТНОЕ ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ. eEPC. ARIS. AnyLogic. БИЗНЕС-ПРОЦЕСС.

In this paper the technique for translating model in eEPC notation into AnyLogic agent-based simulation model are proposed and developed. Performed an experiment of translating model in eEPC notation into Anylogic agent-based simulation model based on the developed algorithm. The results can be used by analysts to reduce simulation models designing and implementing time, and simplifying the exploitation process.

AGENT-BASED SIMULATION. EEPC. ARIS. ANYLOGIC. BUSINESS PROCESS.

В современной практике организационного управления широкое распространение получили графические модели бизнес-процессов. Этот класс методов организационной науки изначально был предназначен для решения проблем избыточной сложности описания и исследования систем. Однако исследование графической модели, даже выполненной в соответствии с правилами структурного подхода, представляет значительную сложность. Вместе с тем существует давно известный и обладающий мощными аналитическими возможностями при исследовании дискретных систем аппарат имитационного моделирования бизнес-систем. Но само по себе имитационное моделирование не решает оптимизационных задач, а скорее представляет собой технику оценки значений функциональных характеристик моделируемой системы, позволяя выявлять проблемные места в системе [1].

В настоящее время для моделирования и анализа бизнес-процессов используются различные подходы и средства. В пер-

вую очередь следует отметить методологии описания бизнес-процессов IDEF, ARIS, BPMN. При этом в программных продуктах, предназначенных для описания бизнес-процессов, нередко присутствуют модули, обеспечивающие проведение имитационного эксперимента на построенных событийных моделях (здесь следует упомянуть об интеграции BPWin и Arena, ARIS Simulation).

Существующие разработки в области трансляции между формальным описанием бизнес-процессов и его имитационной моделью в основном нацелены на получение в конечном итоге дискретно-событийной имитационной модели. Агентные имитационные модели в этой области менее распространены, поскольку агентное моделирование как парадигма появилось относительно недавно. Но следует отметить растущий в последнее время интерес к агентному моделированию.

Обычно при трансляции в качестве исходной нотации формального описания



бизнес-процессов используется уже существующая нотация (IDEF0, IDEF3, UML, BPMN, EPC). В качестве целевой имитационной модели может использоваться как модель в формате существующей имитационной системы (Agena, GPSS, CPN Tools), так и формат для системы имитационного моделирования собственной разработки.

В основном (за счет исторически раннего своего появления и вследствие большей распространенности) в качестве исходной формальной модели бизнес-процессов используется методология IDEF0 и IDEF3 [2–4], менее распространены подходы, основанные на нотациях UML [5] и BPMN [6]. В качестве имитационной модели получили широкое распространение сети Петри [7–9] и GPSS. Все эти подходы в большинстве своем ориентированы на получение в конечном результате дискретно-событийной имитационной модели. Агентные имитационные модели в этой области используются мало, в качестве имитационной системы применяются собственные разработки.

Агент – это некоторая сущность, которая обладает активностью, автономным поведением, может принимать решения в соответствии с некоторым набором правил, может взаимодействовать с окружением и другими агентами, а также может изменяться (эволюционировать) [10].

Агентное имитационное моделирование – направление в имитационном моделировании, используемое для исследования децентрализованных систем, динамика функционирования которых определяется не глобальными правилами и законами (как в других парадигмах моделирования), а наоборот, эти глобальные законы и правила являются результатом индивидуальной активности членов группы. Цель агентных моделей – получить представление об этих глобальных правилах, общем поведении системы, исходя из предположений об индивидуальном, частном поведении ее отдельных активных объектов и взаимодействии этих объектов в системе [10].

Цель данной статьи – предложить методику получения агентной имитационной

модели AnyLogic, моделирующей микроэкономическую систему на основе применяемых для бизнес-моделирования графических методологий, на примере нотации ARIS eEPC.

Методы и материалы исследования

Нотация ARIS eEPC (extended Event Driven Process chain) предназначена для детального описания бизнес-процесса и отражает логику его выполнения [11].

Модель в нотации eEPC определим как [12] $G = \{X, V\}$, где X – непустое множество объектов модели (вершин графа); V – непустое множество дуг V_n и ребер V_r , таких, что $V = V_n \cup V_r$.

Объекты модели (вершины графа): $X = \{S, F, D, I, P, R\}$, где S – множество объектов типа «событие»; F – множество объектов типа «функция»; D – множество объектов типа «документ» и «информационный носитель»; I – множество объектов типа «должность», «организационная единица»; P – множество объектов типа «продукт/услуга»; R – множество объектов типа «правило» исключающего ИЛИ (XOR), ИЛИ (OR), И (AND), задающих логику ветвления/слияния бизнес-процесса.

Множество правил ветвления R состоит из элементов множества $R_{xor} \cup R_{or} \cup R_{and}$, где R_{xor} – множество правил ветвления типа XOR; R_{or} – множество правил ветвления типа OR; R_{and} – множество правил ветвления типа AND.

Множество объектов типа I состоит из элементов $I_{pos} \cup I_{org}$, где I_{pos} – множество объектов типа «должность»; I_{org} – множество объектов типа «организационная единица».

Множество объектов типа D состоит из элементов $D_{doc} \cup D_{inf}$, где D_{doc} – множество объектов типа «документ»; D_{inf} – множество объектов типа «информационный объект».

Агентная модель в AnyLogic реализуется с помощью базового объекта – активного объекта. Активный объект имеет параметры, переменные, которые можно считать памятью агента, стейтчарты или диаграммы состояний выражают поведение: состояния

объекта и изменение состояний под воздействием событий и условий.

Определим агентную модель AnyLogic как $ABM = \{E, A\}$, где E – множество объектов, составляющих окружение агента; A – множество агентов.

Множество объектов E , составляющих окружение агента, определим как $E = \{Tm, Var, Ds, Msg\}$, где Tm – множество таймеров; Var – множество переменных; Ds – множество таблиц для сохранения статистических данных о работе модели и синхронизации; Msg – множество сообщений для взаимодействия агентов со средой и друг с другом.

Множество агентов A определим как $A = \{Sch, Par, Var, Fn\}$, где Sch – непустое множество стейтчартов (диаграмм состояний, соответствует стандарту UML); Par – множество параметров агента; Fn – множество функций агента (под функцией понимается специальный элемент AnyLogic для записи функций на языке Java).

Множество стейтчартов Sch определим как $Sch = \{Bc, St, Trn\}$, где Bc – непустое множество указателей начального состояния; St – непустое множество состояний; Trn – множество переходов.

Перевод графических элементов модели в нотации eEPC в агентную имитационную модель AnyLogic производится согласно табл. 1.

Рассмотрим подробнее методику преобразования модели в нотации eEPC в АИМ AnyLogic. В процессе изложения преобразований будем использовать соответствия для элементов моделей, приведенные в таблице.

В общем виде АИМ AnyLogic, основанная на событийных моделях eEPC, состоит из активных объектов (агентов), имеющих каждый свои правила поведения (задаются диаграммой состояний на языке UML) и взаимодействующих с внешней средой и друг с другом посредством пересылки сообщений (сообщение – специально реализованный программный класс на языке Java).

Итак, имеется модель в нотации eEPC. Необходимо на ее основе создать агентную

имитационную модель в AnyLogic. Вначале создаем структуру имитационной модели. Для этого необходимо следующее.

1. Создать классы агентов имитационной модели AnyLogic на основе организационных единиц модели eEPC. Поместить созданные классы на структурную диаграмму корневого объекта «Main», моделирующего внешнюю среду (объект «Main» создается по умолчанию вместе с созданием имитационной модели, в целом имя класса непринципиально).

2. Внутри класса каждого из агентов создать диаграмму состояний по следующим принципам:

- создать гиперсостояние. В гиперсостоянии создать только одно простое состояние – состояние ожидания, в котором агент находится в самом начале моделирования, а также переходит после каждого простого состояния;

- создать простые состояния – по одному на каждую функцию eEPC, которую выполняет организационная единица;

- создать переходы для каждого из простых состояний в гиперсостояние: от простого состояния к гиперсостоянию и от состояния ожидания, вложенного в гиперсостояние, к простому состоянию. Настроить, чтобы переход от простого состояния к гиперсостоянию срабатывал по таймауту. Настроить, чтобы переход от состояния ожидания к простому состоянию срабатывал по сигналу, и для каждого такого перехода указать уникальное (в пределах стейтчарта) название сигнала.


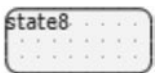


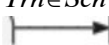










3. Создать Java-класс «сообщение» для описания взаимодействия агентов. В классе должно быть создано поле «fireEvent» строкового типа для указания сигнала перехода в диаграмме состояний.

4. На структурной диаграмме объекта «Main» создать статический таймер по количеству начальных событий в модели eEPC.

5. На структурной диаграмме объекта «Main» создать для каждого из классов агентов векторные переменные для хранения очередей сообщений (дисциплина очереди – FIFO).

Таблица 1

Описание соответствия объектов модели нотации eEPC и элементов агентной имитационной модели на языке AnyLogic

Объект модели eEPC	Графическое обозначение	Соответствующий элемент AnyLogic	Описание элемента AnyLogic
Функция $F \in X \in G$		Состояние $St \in Sch \in A \in ABM$ 	Простое состояние стейтчарта
Начальное событие $S_s \in S \in X \in G$		Таймер $Tm \in E \in ABM$ 	Наступление начальных событий в имитационной модели определяется таймером по заданному аналитиком расписанию
Событие $S_m, S_f \in S \in X \in G$		Переход $Trn \in Sch \in A \in ABM$ 	Переход из гиперсостояния стейтчарта в простое состояние
Материальные и информационные ресурсы			
Документ, информационный носитель $D \in X \in G$		Переменная $Var \in E \in ABM$ 	Переменные используются для моделирования изменяющихся характеристик, хранения результатов моделирования. Изменение количественных ресурсов происходит в состояниях стейтчарта, программируется на языке Java
Продукт/ Услуга $P \in X \in G$			
Исполнители – ресурсы, необходимые для выполнения бизнес-функции, но не изменяемые ею			
Должность $I_{pos} \in I \in X \in G$		Класс активного объекта $A \in ABM$ 	Поведение активного объекта описывается с помощью стейтчарта. В имитационной модели создаются классы активных объектов и далее в модели используется любое число экземпляров класса
Организационная единица $I_{org} \in I \in X \in G$			
Правила ветвления			
Правило ИЛИ $R_{or} \in R \in X \in G$		Переход $Trn \in Sch \in A \in ABM$ 	При переходе из простого состояния стейтчарта в гиперсостояние в методе перехода «Action» на языке Java программируется логика принятия агентом решения
Правило исключяющего ИЛИ $R_{xor} \in R \in X \in G$			
Правило И $R_{and} \in R \in X \in G$			

6. На структурной диаграмме объекта «Main» создать переменные для хранения количества созданных документов — по одной на каждый объект документа в модели eEPC.

После описания структуры АИМ задать логику работы модели.

7. Срабатывание начальных событий в eEPC моделируется статическими таймерами, где прописывается расписание работы — циклическое срабатывание через определенные промежутки времени. При срабатывании таймера происходит создание и отсылка сообщений агентам, которые имеют состояния стейтчарта, соответствующие функциям модели eEPC, следующим за начальным событием. Сигнал для перехода в состояние стейтчарта записывается в поле «fireEvent» сообщения.

8. При получении агентом сообщения происходит переход в простое состояние согласно сигналу в сообщении, если имеется свободный экземпляр агента, иначе сообщение помещается в общую для всех агентов этого класса очередь. Анализ и обработка общей очереди производится каждым агентом после перехода в состояние ожидания (после выхода из простого состояния).

9. Вероятностная логика правил ветвления модели eEPC «ИЛИ» и «исключающего ИЛИ» (OR и XOR) реализуется в методе (под методом здесь подразумевается соответствующий термин объектно-ориентированного программирования) перехода «Action» на языке Java при смене состояний (простого состояния на гиперсостояние). Таким образом, моделируется принятие агентом решения — в какое следующее состояние он должен перейти далее или же он должен перейти в режим ожидания и отправить сообщение (с указанием сигнала для стейтчарта в поле «fireEvent») другому агенту для обработки.

10. Параллельное выполнение модели функций eEPC в имитационной модели также реализуется в методе перехода «Action» на языке Java, при смене состояний стейтчарта, рассылкой сообщений необходимым агентам. Синхронизация правила ветвления «И» (AND) происходит при помощи вну-

тренних таблиц (DataSet), в которые записывается факт произошедших событий.

Получившаяся в результате преобразований по предложенной методике АИМ является «заготовкой» для дальнейшего более детального моделирования: сосредоточение непосредственно на агентах, их поведении и коммуникациях.

В качестве примера применения методики рассмотрим создание агентной имитационной модели на основе модели бизнес-процесса «Формирование заказа на поставку товара», описанного в нотации eEPC (см. рис. 1). Это модель действующего бизнес-процесса на торговом предприятии.

Вначале строим модель в AnyLogic на основе предложенной методики. В представленной модели eEPC введено три организационные единицы: «Служба безопасности», «Отдел выписки» и «Диспетчер», соответственно они будут являться в имитационной модели агентами. На рис. 1 пунктирными стрелками показаны соответствия элементов модели eEPC и элементов агентной имитационной модели AnyLogic на примере агента «сотрудник службы безопасности» и агента «диспетчер».

Фрагменты созданной агентной имитационной модели AnyLogic на основе модели в нотации eEPC представлены на рис. 2.

На рис. 2 а показано дерево объектов модели, на рис. 2 б, в, г отображается состав объектов, вложенных в класс «Main». Среда (окружение) моделируется классом «Main» (содержит переменные, таймер, таблицы), в который вложены активные объекты (рис. 2 г) — агенты (sB_depart, v_depart, dispatcher), пример стейтчартов для агентов можно видеть на рис. 1. Поскольку активные объекты технически — это классы на языке Java, то не представляется возможным давать им имена на русском языке и для пояснений приходится ограничиваться текстовыми комментариями на самой модели. В данном примере, чтобы не перегружать модель комментариями, не сделаны комментарии к переменным (например, переменная «cancelOrders» хранит количество отмененных заказов, перемен-

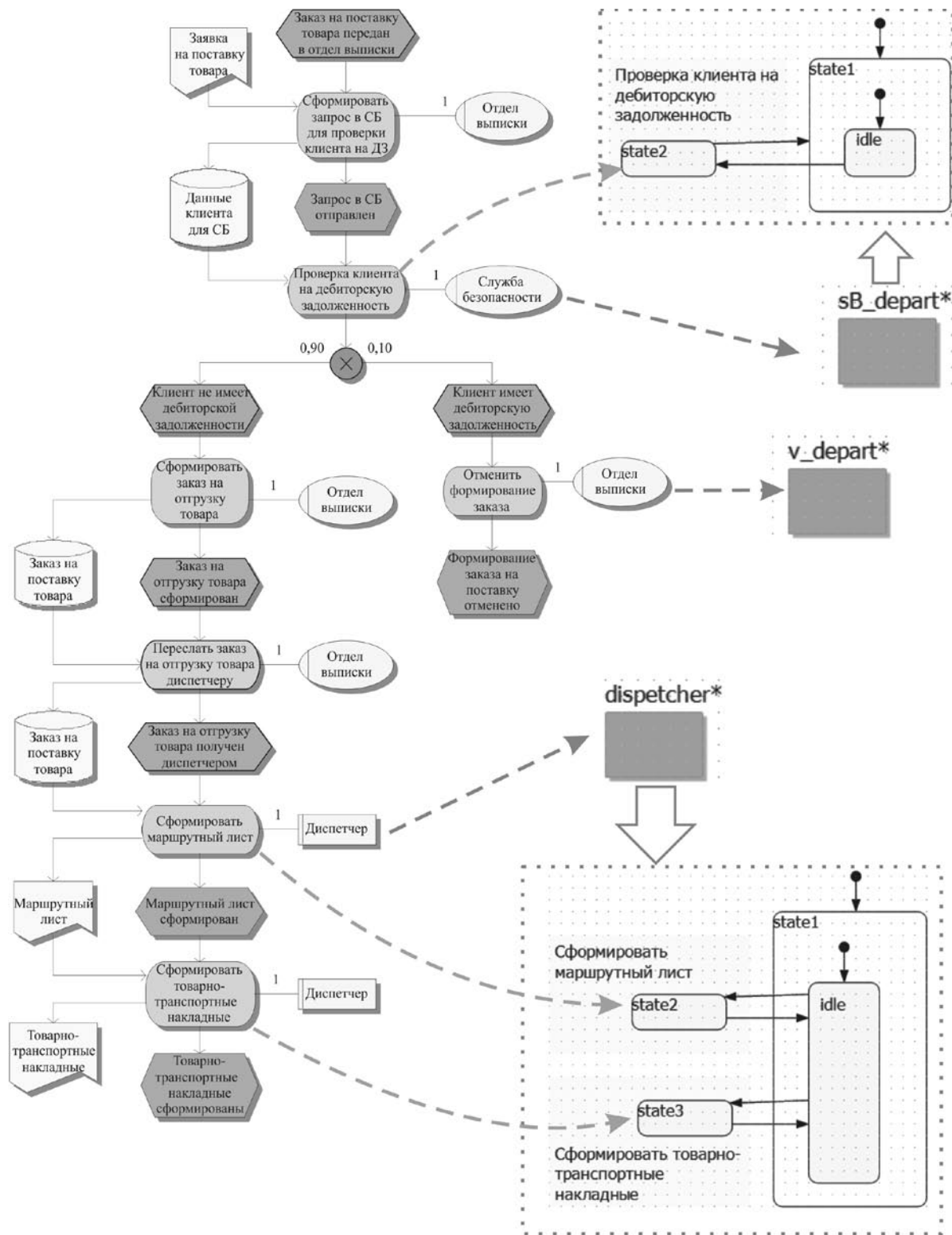


Рис. 1. Формирование заказа на поставку товара

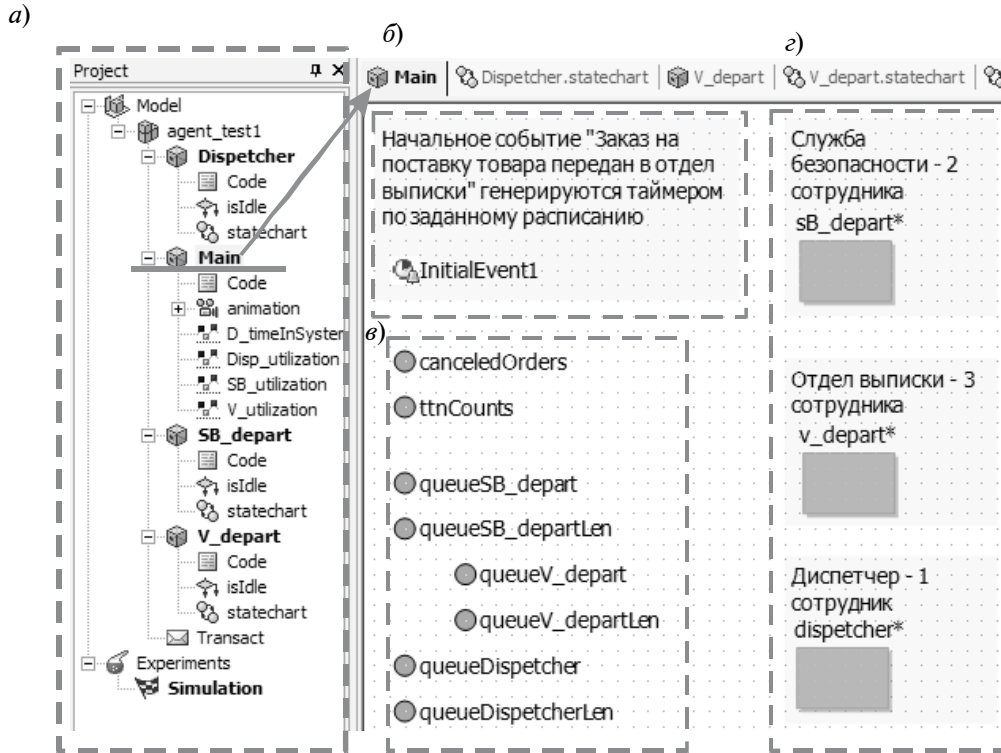


Рис. 2. Активные объекты и их окружение в агентной имитационной модели AnyLogic

ная «ttnCounts» хранит количество сообщений, сгенерированных таймером, и т. д.).

После получения структуры агентной имитационной модели определим численные характеристики объектов модели eEPC, необходимые для проведения имитационного моделирования. Определим время выполнения функций eEPC (зададим время константами на основе среднего значения), количество исполнителей, задействованных в выполнении функции (зададим константами, на рис. 1 в модели eEPC указано их количество на ребре, связывающем организационную единицу и функцию). Определим общее количество исполнителей (организационные единицы) и соответственно экземпляров агентов: «Отдел выписки» – три сотрудника, «Служба безопасности» – два сотрудника, «Диспетчер» – один сотрудник. Вероятности наступления событий в правилах ветвления «исключающего ИЛИ» (XOR) определим как вероятность события «Клиент не имеет дебиторской задолженности» – 0,95, вероятность события «Клиент

имеет дебиторскую задолженность» – 0,05. Наступление начального события «Заказ на поставку товара передан в отдел выписки» будет происходить один раз каждую единицу модельного времени.

Для верификации полученной агентной имитационной модели AnyLogic, сравним ее выходные данные с выходными данными имитационной модели встроенного в ARIS ToolSet имитатора ARIS Simulation.

Сравним усредненные выходные данные (табл. 2) по загруженности исполнителей после имитационных экспериментов в системах AnyLogic и ARIS Simulation за одинаковый модельный промежуток времени (восемь модельных часов) и с одинаковыми входными данными (описаны выше) после одинакового количества экспериментов – всего проводилось десять экспериментальных прогонов моделей в каждой из систем AnyLogic и ARIS Simulation.

Таким образом, сравнивая сводные данные по загруженности исполнителей

Таблица 2

Сводные усредненные данные по загруженности исполнителей, %

	ARIS Simulation	AnyLogic
Отдел выписки	97,8	98,6
Служба безопасности	88,5	83,7
Диспетчер	55,7	55,6

(см. табл. 2), полученные для одной и той же исходной модели eEPC, делаем вывод, что имитационная модель ARIS Simulation и модель AnyLogic, полученная посредством перевода по предложенной методике, показывают близкие результаты.

Следовательно, имитационная модель, разработанная в соответствии с предложен-

ной методикой, обладает прогностической способностью, не уступающей традиционным имитационным моделям.

Результаты работы могут использоваться аналитиками для сокращения времени реализации имитационных моделей, упрощения процесса эксплуатации имитационных моделей.

СПИСОК ЛИТЕРАТУРЫ

1. **Таха, Хемди А.** Введение в исследование операций [Текст] / Хемди А. Таха. – М.: ИД «Вильямс», 2005. – 912 с.
2. **Замятина, О.М.** Моделирование и комплексный анализ бизнес-процессов предприятия [Текст] / О.М. Замятина. – Томск, 2004. – 174 с.
3. **Жевнерчук, Д.В.** Методика построения имитационных моделей с применением больших массивов данных на основе интеграции IDEF3, OLAP, GPSS технологий [Текст] / Д.В. Жевнерчук. – Ижевск, 2006. – 146 с.
4. **Тихонов, С.В.** Моделирование бизнес-процессов с использованием аппарата теории систем массового обслуживания [Текст] / С.В. Тихонов // Сб. тр. II конф. II школа-семинар молодых ученых. Управление большими системами. – Воронеж: Научная книга, 2007. – Т. 2. – С. 53–59.
5. **Ханова, А.А.** Организация принятия решений в виде цикла управления эффективностью организации [Текст] / А.А. Ханова, А.С. Пономарёва // Вестник астраханского государственного технического университета. Управление, вычислительная техника и информатика. – 2011. – № 2. – С. 171–177.
6. **Валькман, Ю.Р.** Определение базиса для построения мультиагентной системы имитационного моделирования бизнес-процессов в нотации BPMN [Текст] / Ю.Р. Валькман, К.В.

Муцаковская // Институту проблем моделювання в енергетиці. Збірник наукових праць. – 2012. – № 62. – С. 101–108.

7. **Van der Aalst, W.M.P.** The Application of Petri Nets to Workflow Management [Text] / W.M.P. Van der Aalst // J. of Circuits, Systems and Computers, 1998. – № 8(1). – P. 21–66.

8. **Van der Aalst, W.M.P.** Business Process Management [Text] / W.M.P. van der Aalst, A.H.M. ter Hofstede, M. Weske // A Survey. Internat. Conf. on Business Process Management, LCNS. – Berlin: Springer-Verlag, 2003. – Vol. 2678. – P. 1–12.

9. **Jensen, K.** Coloured Petri nets: Basic concepts, analysis methods and practical use [Text] / K. Jensen // Basic concepts. – Berlin: Springer-Verlag, 1996. – Vol. 1.

10. **Карпов, Ю.Г.** Имитационное моделирование систем. Введение в моделирование с AnyLogic 5 [Текст] / Ю.Г. Карпов. – СПб.: БХВ-Петербург, 2006. – 400 с.

11. **Шеер, А.В.** Бизнес-процессы. Основные понятия. Теория. Методы [Текст] / А.В. Шеер. – М.: Весть-МетаТехнология, 1999. – 182 с.

12. **Дорпер, М.Г.** Событийный анализ бизнес-процессов в нотации eEPC [Текст] / М.Г. Дорпер, Е.А. Ланцев, А.И. Шаргаева // Тр. XV Междунар. ЭМ конф. по эвентологической математике и смежным вопросам. – Красноярск: СФУ, НИИППБ, КГТЭИ, 2011. – С. 89–96.

REFERENCES

1. **Takha Khemdi A.** Vvedenie v issledovanie operatsii. – Moscow: Izdatel'skii dom «Vil'iams», 2005. – 912 s. (rus)
2. **Zamiatina O.M.** Modelirovanie i komplek-

snyi analiz biznes-protsessov predpriatiia. – Tomsk, 2004. – 174 s. (rus)

3. **Zhevnerchuk D.V.** Metodika postroeniia imitatsionnykh modelei s primeneniem bol'shikh mas-

sivov dannykh na osnove integratsii IDEF3, OLAP, GPSS tekhnologii. — Izhevsk, 2006. — 146 s. (rus)

4. **Tikhonov S.V.** Modelirovanie biznes-protsessov s ispol'zovaniem apparata teorii sistem massovogo obsluzhivaniia / Sb. trudov II konf. II shkola-seminar molodykh uchenykh Upravlenie bol'shimi sistemami. — Voronezh: Nauchnaia kniga, 2007. — Т. 2. — С. 53–59. (rus)

5. **Khanova A.A., Ponomareva A.S.** Organizatsiia priniatiia reshenii v vide tsikla upravleniia effektivnost'iu organizatsii / Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo universiteta. Upravlenie, vychislitel'naia tekhnika i informatika. — 2011. — № 2. — С. 171–177. (rus)

6. **Val'kman Yu.R., Mutsakovskaia K.V.** Opredelenie bazisa dlia postroeniia mul'tiagentnoi sistemy imitatsionnogo modelirovaniia biznes-protsessov v notatsii BPMN / Institutu problem modeliuvaniia v energetitsi. Zbirnik naukovikh prats'. — 2012. — № 62. — С. 101–108.

7. **van der Aalst W.M.P.** The Application of Petri Nets to Workflow Management / The J. of Cir-

uits, Systems and Computers. — 1998. — № 8 (1). — P. 21–66.

8. **van der Aalst W.M.P., ter Hofstede A.H.M., Weske M.** Business Process Management: A Survey / International Conf. on Business Process Management (BPM 2003). — Berlin: Springer-Verlag, 2003. — LCNS Vol. 2678. — P. 1–12.

9. **Jensen K.** Coloured Petri nets: Basic concepts, analysis methods and practical use. — Berlin: Springer-Verlag, 1996. — Vol. 1. Basic concepts.

10. **Karpov Yu.G.** Imitatsionnoe modelirovanie sistem. Vvedenie v modelirovanie s AnyLogic 5. — St. Petersburg: BKhV-Petersburg, 2006. — 400 s. (rus)

11. **Sheer A.V.** Biznes-protsessy. Osnovnye poniatiia. Teoriia. Metody. — Moscow: Vest'-MetaTekhnologiiia, 1999. — 182 s. (rus)

12. **Dorrer M.G., Lantsev E.A., Shargaeva A.I.** Sobytiinyi analiz biznes protsessov v notatsii eEPC // Tr. XV Mezhdunar. EM konf. po eventologicheskoi matematike i smezhnym voprosam. — Krasnoiar'sk: SFU, NIIPPB, KGTEI, 2011. — С. 89–96. (rus)

ЛАНЦЕВ Евгений Алексеевич — аспирант кафедры схемотехники Сибирского государственного технологического университета.

660049, Россия, г. Красноярск, пр. Мира, д. 82.

E-mail: red_snow@list.ru

LANTSEV, Yevgeniy A. *Siberian State Technological University.*

660049, pr. Mira, 82, Krasnoiar'sk, Russia.

E-mail: red_snow@list.ru

ДОРРЕР Михаил Георгиевич — доцент кафедры схемотехники Сибирского государственного технологического университета, кандидат технических наук.

660049, Россия, г. Красноярск, пр. Мира, д. 82.

E-mail: mdorrer@mail.ru

DORRER, Mikhail G. *Siberian State Technological University.*

660049, pr. Mira, 82, Krasnoiar'sk, Russia.

E-mail: mdorrer@mail.ru