

**Н. В. Тюльпинова**, канд. техн. наук, доц.  
Брянский государственный технический университет, Россия  
Тел./Факс: +7 920 862 47 20; E-mail: [ninatulpinova@mail.ru](mailto:ninatulpinova@mail.ru)

## АВТОМАТИЗИРОВАННАЯ СИСТЕМА МОДЕЛИРОВАНИЯ ДИСКРЕТНЫХ ПРОИЗВОДСТВЕННЫХ СИСТЕМ

*Разработана имитационная модель дискретной производственной системы, позволяющая проводить оценку эффективности использования технологического оборудования для различных вариантов конфигурации этой системы. Приведено описание алгоритма моделирования. Представлен пользовательский интерфейс.*

**Ключевые слова:** дискретная производственная система, имитационная модель, алгоритм.

**N. V. Tyulpinova**

## THE AUTOMATED SYSTEM OF SIMULATION MODELLING OF DISCRETE INDUSTRIAL SYSTEMS

*The simulation model of discrete industrial system is developed, allowing to estimate the efficiency of technological equipment usage for various versions of the system. The algorithm of modeling is described. The user's interface is presented.*

**Key words:** discrete industrial system, simulation model, algorithm.

### 1. Введение

Эффективность современных машиностроительных производств во многом определяется уровнем управления и организации производственных и технологических процессов. Одними из ключевых элементов системы управления предприятием являются модули планирования. От методов решения задач планирования, от адекватности математических моделей систем оперативного планирования зависят точность формирования плановых заданий и управление объектами производства [1, 2]. Однако, несмотря на наличие значительного количества теоретических разработок в данной области, вопросы влияния на эффективность использования оборудования таких организационных факторов, как характер производственных связей предприятия, вероятностный характер производственного процесса, схемы организации работы оборудования – исследованы недостаточно полно. Кроме того, обоснование решений по реализации организационных резервов повышения эффективности использования оборудования довольно часто осуществляется без должного учета требований системного подхода, а используемые на практике методы и средства не позволяют в полной мере учесть влияние ряда факторов. Наряду с перечисленными проблемами следует отметить недостаточно полную теоретическую проработку вопросов количественной оценки эффективности использования оборудования. Всё вышеперечисленное свидетельствует о том, что в настоящее время возрастает важность и роль поиска новых инструментов и методов выявления и реализации организационных резервов повышения эффективности использования оборудования. Одним из таких инструментов является имитационное моделирование [3, 4, 5], которое позволяет рассматривать и изучать процессы, происходящие на предприятии с различным уровнем детализации. В связи с этим представленные ниже результаты исследований – имитационная модель дискретной производственной системы и её алгоритмическая структура – являются актуальными и представляют интерес как с научной, так и с практической точки зрения.

## 2. Основное содержание и результаты работы

В основу разработанного алгоритмического и программного обеспечения положен дискретно-событийный подход к моделированию (выделяются узловые моменты динамики в виде событий) с использованием концепции связанного распределения памяти. В процессе моделирования осуществляется переход (скачок во времени) от предыдущего события к последующему. Каждое событие выполняется мгновенно во времени, модельное время затрачивается только на переход от события к событию. Реализация событий во времени аналогична цепной реакции: при обработке каждого события планируется одно или несколько последующих (будущих) событий. При этом учитывается, что количество типов событий для модели дискретной производственной системы равно 3: событие типа 1 представляет собой поступление работы в систему; событие типа 2 – уход работы с определенного рабочего места производственной системы (это событие не всегда означает уход работы из системы вообще, а только в том случае, когда рабочее место будет последним на маршруте работы); событие типа 3 – завершение моделирования. Программный модуль для имитационного моделирования дискретных производственных систем состоит из одной основной программы (рис. 1а) и трех вызываемых из нее подпрограмм (рис. 1б – 1г), каждая из которых обрабатывает событие определенного типа.

В блоках 1–12 основной программы (рис. 1а) организован ввод исходных данных, которыми для данной модели дискретной производственной системы являются: `num_stations` – число автоматизированных рабочих мест (индекс  $j$  используется в качестве индекса рабочего места или номера задания); `num_job_types` – число типов работ (индекс  $i$  используется в качестве индекса типа работы); `mean_interarrival` – среднее значение времени между поступлением работ в систему (единица измерения – час); `length_simulation` – длительность моделируемого периода (единица измерения – 8-часовой день); `num_machines[j]` – число станков на рабочем месте  $j$ ; `num_tasks[i]` – общее число заданий для работы  $i$ ; `route[i, j]` – рабочее место для задания  $j$  работы типа  $i$ ; `mean_service[i, j]` – среднее значение времени обслуживания задания  $j$  работы типа  $i$ ; `prob_distrib_job_type[i]` – вероятность того, что работа будет относиться к типу, номер которого меньше или равен  $i$ . В рассматриваемой модели используется 3 типа случайных величин (`mean_interarrival`, `mean_service[i, j]`, `prob_distrib_job_type[i]`), законы распределения которых также являются исходными данными. Блок 13 представляет собой циклическую конструкцию, при выполнении которой все станки на всех рабочих местах первоначально устанавливаются в состояние незанятости (блок 14). В блоке 15 вызывается подпрограмма инициализации моделирования `initialization`, которая устанавливает имитационную модель в исходное состояние в момент времени, равный 0 (эта подпрограмма обнуляет часы модельного времени и накопители статистики, выделяет пространство для хранения списков, устанавливает в исходное состояние ссылки на следующую и предыдущую записи, начальный и конечный указатель каждого списка, задает необходимые значения переменных с целью обеспечения сортировки списка событий по времени их возникновения). В блоке 16 задается максимальное число атрибутов `maxatg` для одной записи. В блоке 17 планируется поступление первой работы: для этого вызывается подпрограмма `event_schedule`, которая планирует возникновение события типа 1 или `EVENT_ARRIVAL` на момент времени `random(mean_interarrival)`. В блоке 18 планируется завершение моделирования: для этого вызывается подпрограмма `event_schedule`, которая планирует возникновение события типа 3 или `EVENT_END_SIMULATION` на момент времени `8 * length_simulation` (здесь 8 означает поправку на

единицы времени, т.е. 8-часовой рабочий день). Блок 24 представляет собой цикличес-

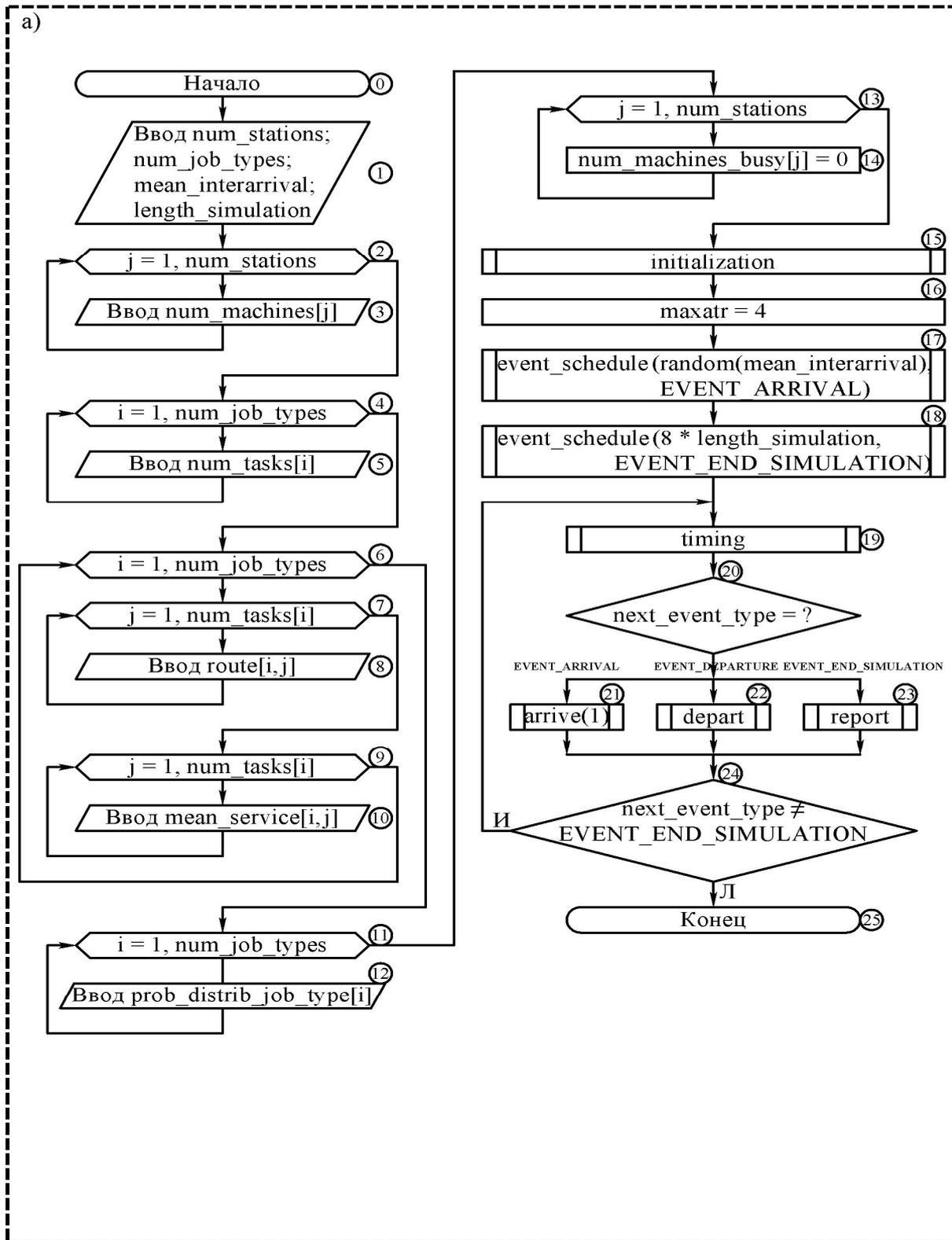
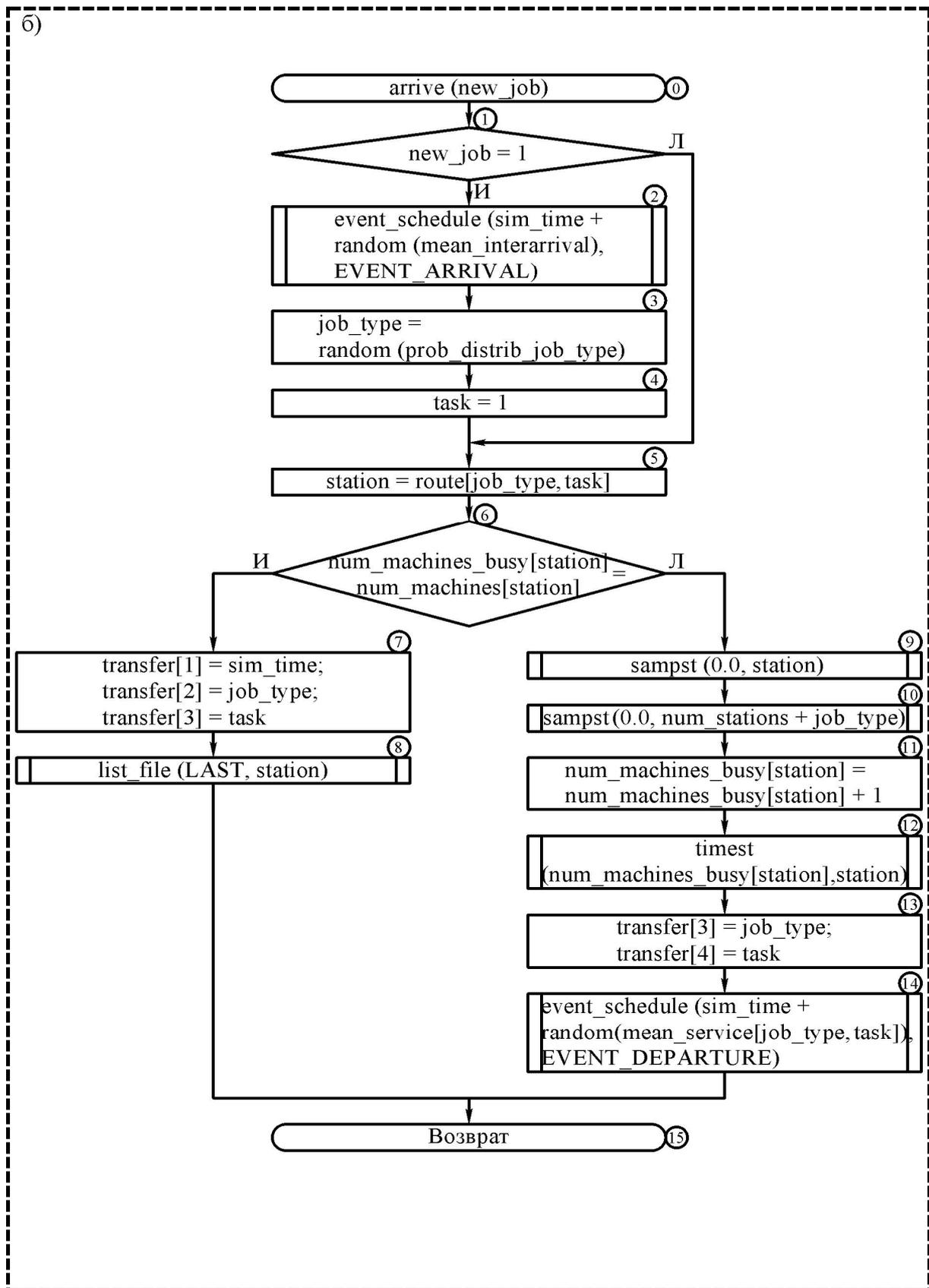
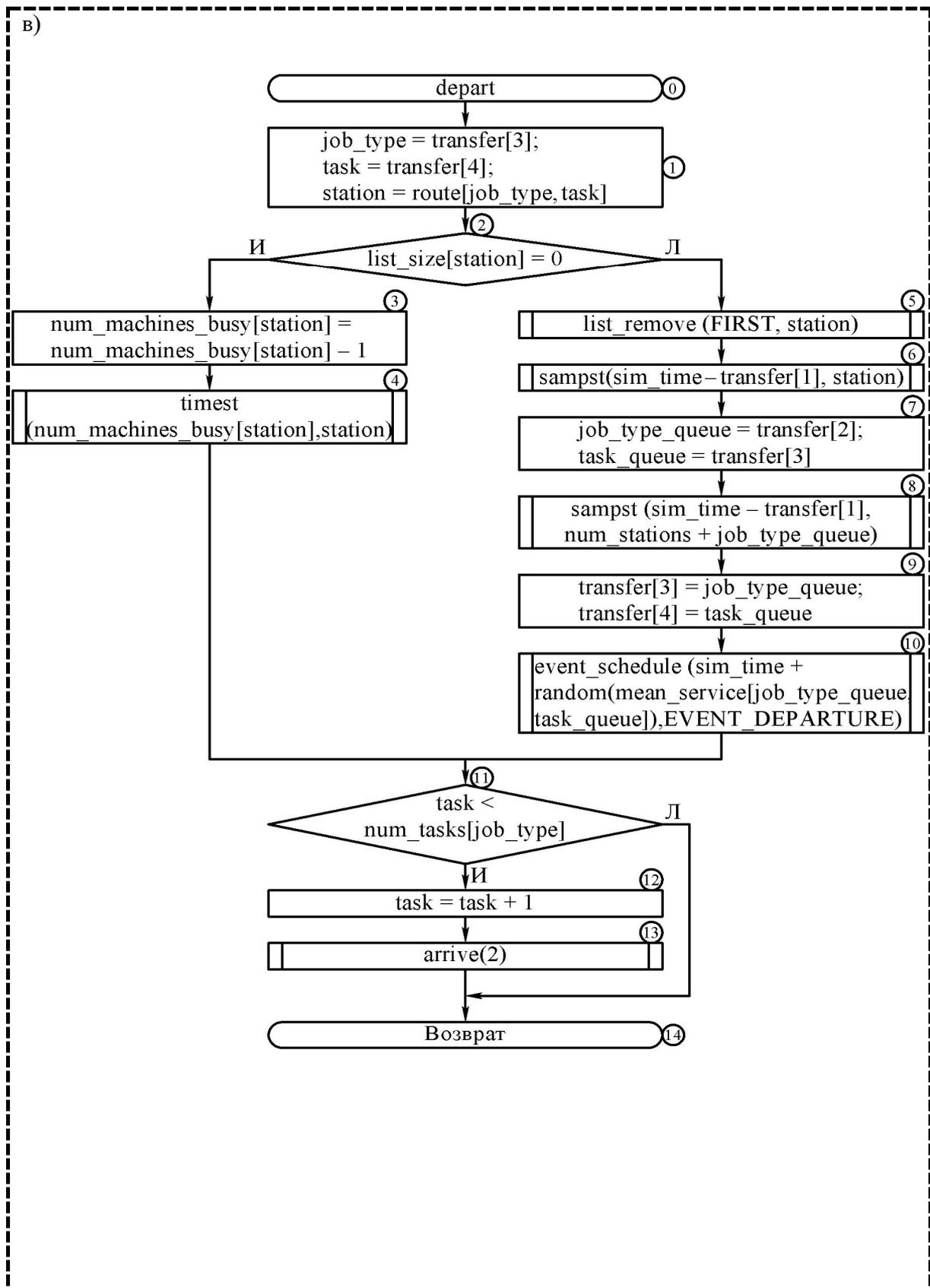


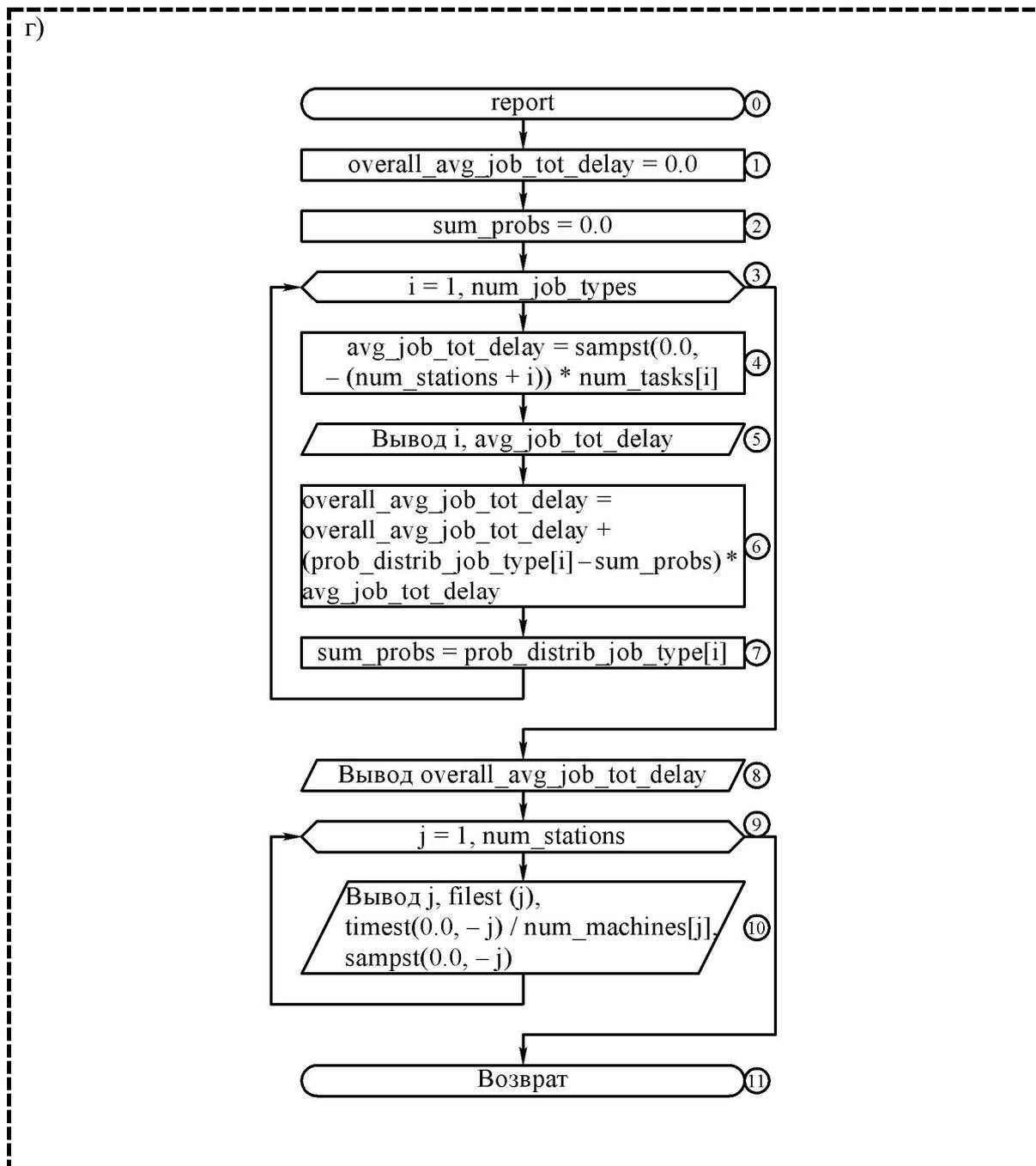
Рис. 1. Алгоритм имитационного моделирования дискретной производственной системы



Продолжение рис. 1



Продолжение рис. 1



Окончание рис. 1

кую конструкцию, объединяющую многократно повторяющуюся последовательность блоков 19–23. Условием выхода из данного цикла является возникновение события типа 3 или EVENT\_END\_SIMULATION. В блоке 19 вызывается синхронизирующая подпрограмма timing, которая определяет тип следующего события next\_event\_type и обновляет значение часов модельного времени sim\_time в соответствии со временем возникновения следующего события. В блоке 20 в зависимости от типа следующего события осуществляется выбор одной из трех возможных альтернатив вычислительного процесса: управление передается или подпрограмме arrive (1) обработки событий типа

1 или EVENT\_ARRIVAL (блок 21), или подпрограмме depart обработки событий типа 2 или EVENT\_DEPARTURE (блок 22), или подпрограмме report обработки событий типа 3 или EVENT\_END\_SIMULATION (блок 23). После выхода из цикла блока 24 моделирование завершается. Описанная последовательность действий представляет собой один прогон имитационной модели. Для получения достоверных результатов следует провести такое количество вычислительных экспериментов, которое обеспечит требуемую точность расчетов, а затем провести осреднение выходных параметров по всем имитационным прогонам.

Подпрограмма, представленная на рис. 1б, обрабатывает событие поступления работы в систему. Подпрограмма arrive начинается с проверки значения аргумента new\_job (блок 1), по которому определяется, как будет использоваться arrive: если new\_job = 1, то это означает, что в систему поступила новая работа, и arrive используется как событийная подпрограмма; если new\_job = 2, то это означает, что работа покидает одно рабочее место и поступает на следующее рабочее место ее маршрута, и arrive используется как несобытийная служебная подпрограмма. Если происходит поступление новой работы в систему, то выполняется последовательность блоков 2–4. В блоке 2 планируется следующее событие поступления работы: для этого вызывается подпрограмма event\_schedule, которая планирует возникновение события типа 1 или EVENT\_ARRIVAL на момент времени sim\_time + random(mean\_interarrival). В блоках 3, 4 генерируется тип работы job\_type, а также номер задания task поступившей работы. В блоке 5 определяется рабочее место station, на которое поступает работа (по ее типу и номеру задания). В блоке 6 проверяется: есть ли на этом рабочем месте свободные станки. Если все станки заняты, то работа помещается в конец очереди на этом рабочем месте, для чего в блоке 8 вызывается подпрограмма list\_file. Массив transfer используется для передачи записей в списки и из списков, применяемых при моделировании, transfer[i] относится к атрибуту i записи. В блоке 7 в записи для каждой работы сохраняются следующие данные: 1 – время поступления на данное рабочее место; 2 – тип работы; 3 – номер текущего задания. Если на этом рабочем месте есть свободный станок, то начинается обслуживание поступившей работы (блоки 9–14). Работа на этом рабочем месте имеет задержку, равную 0, которая регистрируется подпрограммой sampst как для рабочего места (блок 9), так и для типа работы (блок 10). В блоке 11 станок переводится в состояние занятости (num\_machines\_busy[station] – число занятых станков на рабочем месте station). В блоке 12 с помощью подпрограммы timest реализуется сбор статистики. В блоке 13 определяются два последних атрибута записи события. В блоке 14 планируется событие ухода этой работы с рабочего места: для этого вызывается подпрограмма event\_schedule, которая планирует возникновение события типа 2 или EVENT\_DEPARTURE на момент времени sim\_time + random(mean\_service[job\_type, task]). Затем управление передается основной программе (блок 15).

Подпрограмма, представленная на рис. 1в, обрабатывает событие ухода работы с определенного рабочего места. В блоке 1 определяется рабочее место, которое покидает работа (подпрограмма depart получает значения job\_type и task из записи события ухода, а номер рабочего места station из массива route). В блоке 2 проверяется: отсутствует ли очередь к этому рабочему месту (list\_size[station] – количество записей в списке station). Если в очереди на рабочем месте нет работ, то в блоке 3 станок переводится в состояние незанятости, а в блоке 4 с помощью подпрограммы timest реализуется сбор статистики. Если на рабочем месте есть очередь, то из нее с помощью подпрограммы list\_remove (блок 5) извлекается первая работа с типом job\_type\_queue и номером задания task\_queue, позволяющими отличить ее от предыдущей работы, покидающей рабо-

чее место. Посредством программы sampst в блоке 6 вычисляется задержка работы для этого рабочего места, в блоке 8 вычисляется та же задержка для этого типа работы. В блоке 9 определяются два последних атрибута записи события. В блоке 10 планируется завершение обслуживания этой работы на данном рабочем месте: для этого вызывается подпрограмма event\_schedule, которая планирует возникновение события типа 2 или EVENT\_DEPARTURE на момент времени sim\_time + random (mean\_service[job\_type\_queue, task\_queue]). В блоке 11 проверяется: есть ли у работы, покинувшей рабочее место, другие невыполненные задания. Если логическое условие блока 11 выполняется, то к значению переменной task для этой работы прибавляется 1 (блок 12), а данная работа отправляется на следующее рабочее место ее маршрута (блок 13), это реализуется посредством вызова подпрограммы active с аргументом new\_job = 2. Затем управление передается основной программе (блок 14).

Подпрограмма, представленная на рис. 1г, обрабатывает событие завершения моделирования, и генерирует отчет о результатах одного имитационного прогона. В результате выполнения блоков 1–8 рассчитываются и выводятся на экран: 1) значения средней общей задержки avg\_job\_tot\_delay во всех очередях для каждого типа работы i; 2) значение полной средней общей задержки работ overall\_avg\_job\_tot\_delay. В результате выполнения блоков 9–10 для каждого рабочего места j рассчитываются и выводятся на экран: 1) среднее число работ в очереди filest(j); 2) средний коэффициент загрузки оборудования timest(0.0, -j)/num\_machines[j]; 3) средняя задержка в очереди sampst(0.0, -j). Затем управление передается основной программе (блок 11).

На основе рассмотренного алгоритма разработана автоматизированная система, позволяющая проводить расчеты как для одного варианта конфигурации производственной системы (рис. 2а, 2б), так и для нескольких вариантов одновременно (рис. 2в)

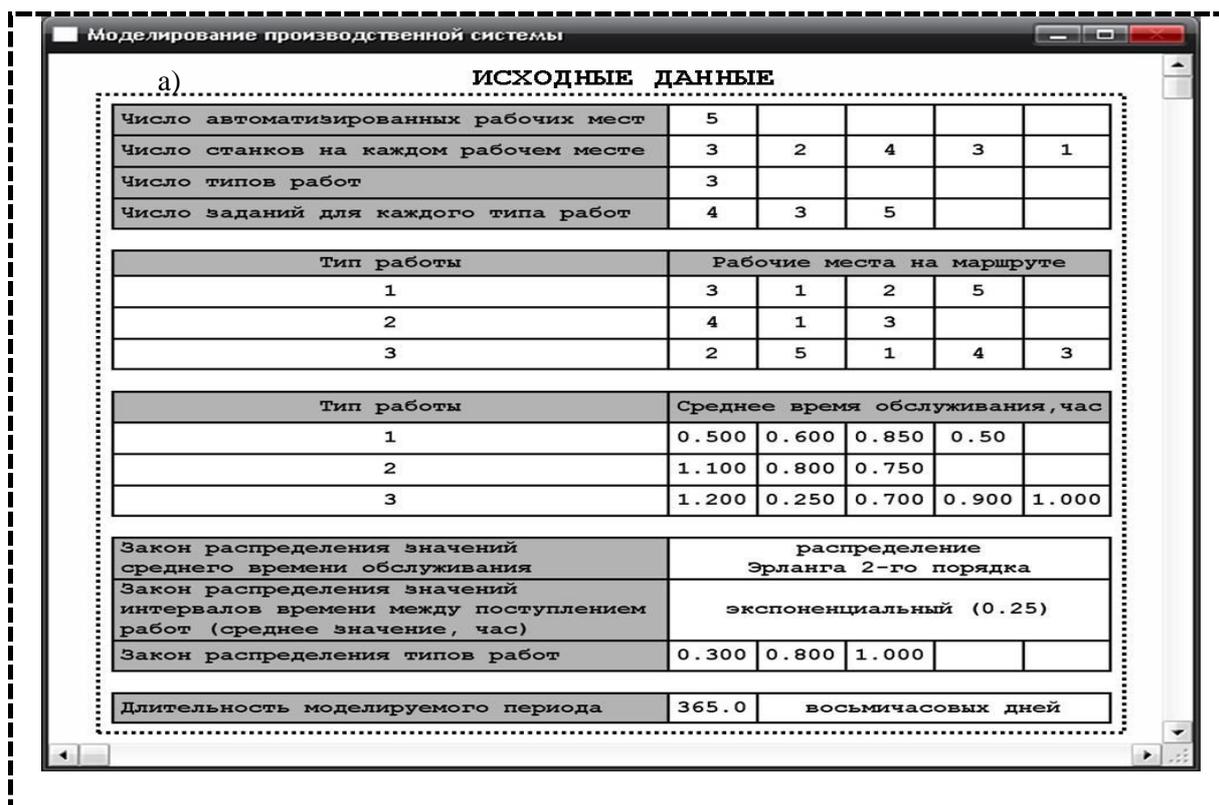
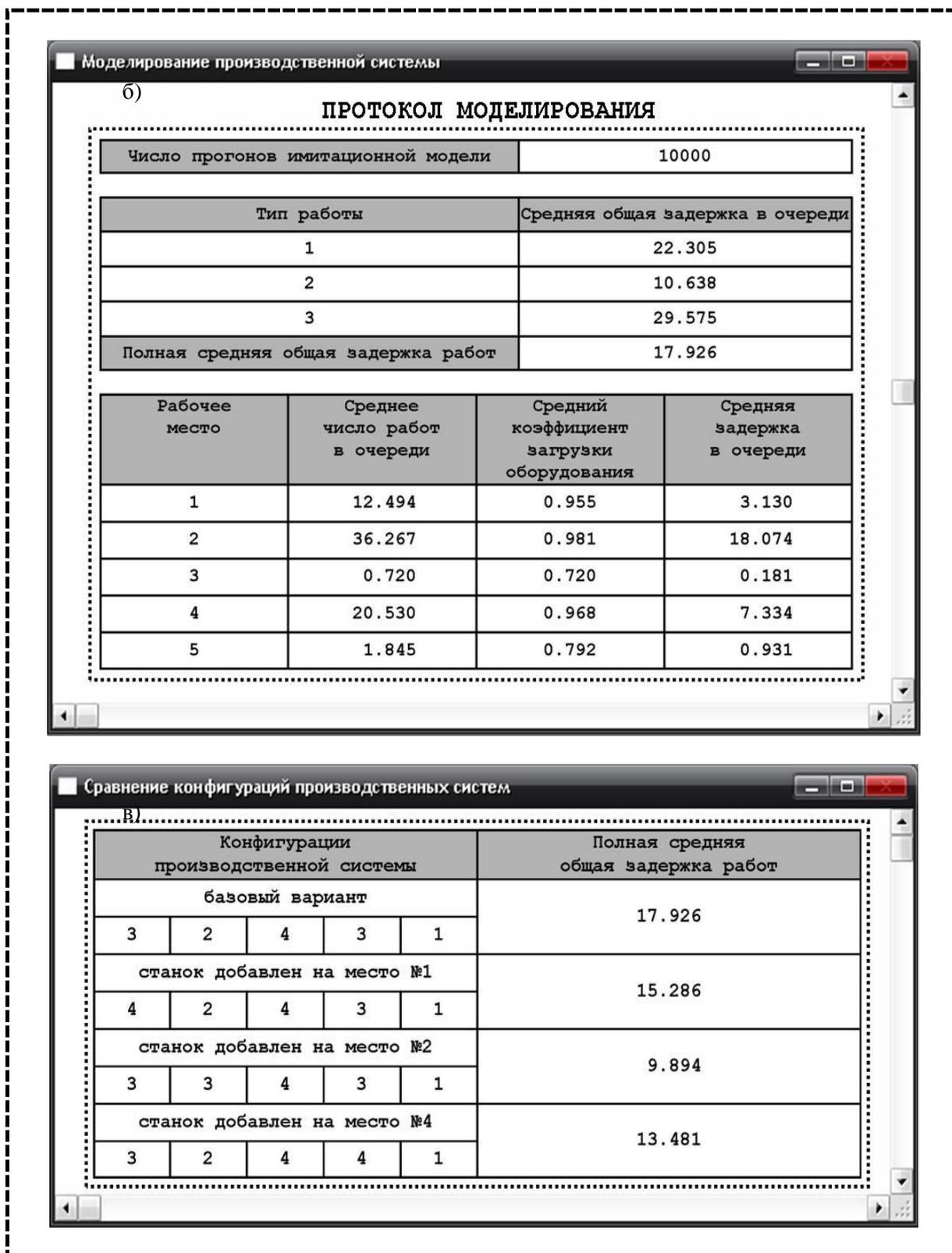


Рис. 2. Пользовательский интерфейс автоматизированной системы



Окончание рис. 2

с возможностью сопоставления и сравнения их между собой, а также выбора оптимального варианта конфигурации. При этом обеспечивается удобство ввода и отображения исходных данных, а также наглядное представление результатов моделирования. Кроме того, программный модуль позволяет выявить и количественно оценить характер изменения всех выходных параметров модели дискретной производственной системы в зависимости от параметров ее конфигурации.

### **3. Заключение**

Предложенный инструментарий (разработанное алгоритмическое и программное обеспечение) и получаемые с его помощью результаты имитационного моделирования дискретной производственной системы могут быть использованы промышленными предприятиями в целях количественной оценки эффективности использования оборудования и выбора оптимального варианта конфигурации производственной системы, при этом критериями эффективности использования оборудования являются выходные параметры рассматриваемой имитационной модели. Предлагаемая модель позволяет сформировать план загрузки оборудования на заданный период времени, спрогнозировать дату выполнения заказов, а также обеспечивает незамедлительное динамическое обновление плана загрузки при возникновении изменений в производственном процессе.

### **ЛИТЕРАТУРА**

1. Плотников А. М. Анализ современного состояния и тенденции развития имитационного моделирования в Российской Федерации (по материалам конференции «Имитационное моделирование. Теория и практика» (ИММОД)) / А. М. Плотников, Ю. И. Рыжиков, Б. В. Соколов, Р. М. Юсупов // Труды СПИИРАН. – Санкт-Петербург, 2013. – № 2(25). – С. 42–112.
2. Толуев Ю. И. Инженерные традиции в имитационном моделировании производственных и логистических систем / Ю. И. Толуев, Д. А. Иванов // Имитационное моделирование. Теория и практика: Сборник докладов пятой юбилейной всероссийской научно-практической конференции ИММОД-2011. – СПб.: ОАО «ЦТСС». 2011. – Том 1. – С. 75–82.
3. Алиев Т. И. Основы моделирования дискретных систем / Т. И. Алиев. – СПб.: СПбГУ ИТМО, 2009. – 363 с.
4. Кобелев Н. Б. Имитационное моделирование / Н. Б. Кобелев, В. А. Половников, В. В. Девятков. – М.: КУРС: НИЦ Инфра-М, 2013. – 368 с.
5. Рыжиков Ю. И. Имитационное моделирование. Теория и технологии / Ю. И. Рыжиков. – СПб.: КОРОНА принт; М.: Альтекс-А, 2004. – 384 с.

Поступила в редколлегию 21.01.2016 г