

УДК 004.75

## ОРГАНИЗАЦИЯ ИМИТАЦИОННЫХ ЭКСПЕРИМЕНТОВ ПРИ ПРОЕКТИРОВАНИИ РАСПРЕДЕЛЕННЫХ КОМПЬЮТЕРНЫХ СИСТЕМ С РЕЗЕРВИРОВАННОЙ ПЕРЕДАЧЕЙ ДАННЫХ

С.А. Паршутина<sup>а</sup>

<sup>а</sup> Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация  
Адрес для переписки: svetlana.parshutina@gmail.com

### Информация о статье

Поступила в редакцию 26.06.17, принята к печати 25.07.17

doi: 10.17586/2226-1494-2017-17-5-841-849

Язык статьи – русский

**Ссылка для цитирования:** Паршутина С.А. Организация имитационных экспериментов при проектировании распределенных компьютерных систем с резервированной передачей данных // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17. № 5. С. 841–849. doi: 10.17586/2226-1494-2017-17-5-841-849

### Аннотация

**Предмет исследования.** Исследованы варианты многопутевого распределения копий запросов (пакетов) через сеть и их резервированного обслуживания серверами кластера. Рассмотренные альтернативы различаются по способу выбора пути следования пакетов в сети с пересекающимися в общих коммутационных узлах маршрутами (статического или динамического) и условию досрочного выхода копий запросов из системы в результате их потерь в каналах связи, возникновения битовых ошибок, переполнения приемо-передающих буферов сетевого и вычислительного оборудования или комбинации этих причин. **Методология исследования.** Рассматривается зависимость эффективности резервированных передач, определяемой вероятностью безошибочной и своевременной доставки в кластер хотя бы одной из резервных копий, от интенсивности входного потока запросов. На основе описанных вариантов в системе имитационного моделирования AnyLogic 7 создается набор имитационных моделей с параметрами, отражающими требования к моделируемой системе. Результаты имитационных экспериментов сохраняются в базе данных Microsoft SQL Server 2008 и после преобразований экспортируются в разработанное в рамках этого исследования программное средство. **Основные результаты.** Предложена организация имитационных экспериментов с накоплением и обработкой их результатов средствами выбранной системы управления базами данных и последующим применением для анализа получаемых зависимостей сторонних программных приложений, таких как Microsoft Excel или реализованного программного средства для определения области эффективности резервированных передач и автоматизации поиска оптимальной кратности резервирования. Представлены результаты имитационного моделирования, подтверждающие эффективность многопутевого распределения копий запросов. **Практическая значимость.** Предложенные модели и инструментальные средства позволяют обосновать выбор и оптимизировать проектные решения по повышению надежности функционирования распределенных компьютерных систем с резервированной передачей и обслуживанием запросов.

### Ключевые слова

средства поддержки проектирования, имитационное моделирование, многопутевая передача, резервирование, надежность

## ORGANIZATION OF SIMULATION EXPERIMENTS IN THE DESIGN OF DISTRIBUTED COMPUTER SYSTEMS WITH REDUNDANT DATA TRANSMISSION

S.A. Parshutina<sup>а</sup>

<sup>а</sup> ITMO University, Saint Petersburg, 197101, Russian Federation  
Corresponding author: svetlana.parshutina@gmail.com

### Article info

Received 26.06.17, accepted 25.07.17

doi: 10.17586/2226-1494-2017-17-5-841-849

Article in Russian

**For citation:** Parshutina S.A. Organization of simulation experiments in the design of distributed computer systems with redundant data transmission. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2017, vol. 17, no. 5, pp. 841–849 (in Russian). doi: 10.17586/2226-1494-2017-17-5-841-849

**Abstract**

**Subject of Research.** We study the ways of multipath distribution of copies of requests (packets) through the network and their redundant servicing by servers of a given cluster. The alternatives under consideration differ in the way of path selection for a packet in the network having routes with joint switching nodes (static or dynamic) and in the condition under which those copies exit early from the system as a result of their losses in the communication channels, the emergence of bit errors, the overflow of transmitting and receiving buffers of the network and computing equipment, or the combination of these reasons. **Method.** We consider the efficiency of redundant transmissions, defined as the probability of faultless and timely delivery of at least one redundant copy to the cluster and dependent on the intensity of the flow of incoming requests. Based on the alternatives described, a set of simulation models with parameters representing the requirements for the system under simulation is created with AnyLogic 7 simulation environment. The results of simulation experiments are being written to the Microsoft SQL Server 2008 database, undergo transformations, and then are exported into the application program developed in this study. **Main Results.** Organization of simulation experiments involving accumulating and processing of their results with the tools of the chosen database management system is proposed. After being handled, those results are used to form the basis for analysis of the obtained dependencies with third-party software applications, such as Microsoft Excel or the one implemented in this work, to determine the scope of efficiency of redundant transmissions and to automate the search for the optimal order of redundancy. The simulation results that confirm the efficiency of multipath distribution of copies of requests are presented. **Practical Relevance.** The proposed models and tools enable one to justify the choice and optimize project solutions to the problem of improving functional reliability of distributed computer systems with redundant transmission and servicing of requests.

**Keywords**

design support tools, simulation, multipath transfer, redundancy, reliability

**Введение**

Резервирование структуры [1–7] сети и средств хранения и обработки данных является необходимым условием построения высоконадежных распределенных вычислительных систем. Повысить своевременность получения результатов, надежность и устойчивость процессов передачи и обработки данных в условиях сбоев, отказов и временной перегруженности ресурсов в компьютерных системах [1–11] позволяет резервированное обслуживание запросов [8–10], резервирование передач и обработки данных [9–11], в том числе при динамическом распределении запросов [12, 13]. Резервированная передача данных, как показано в [14–17], эффективна при многопутевой маршрутизации и ее использовании для распределения запросов через сеть.

При выборе проектных решений построения инфокоммуникационных систем действенным является использование имитационного моделирования [18–21]. Цель данной работы – повышение эффективности проектирования распределенных вычислительных систем на основе имитационных моделей и инструментальных средств поддержки анализа, выбора и оптимизации проектных решений по организации резервированной многопутевой передачи данных.

**Имитационные модели резервированных многопутевых передач**

Цель проведения имитационного моделирования – выбор вариантов организации процесса создания копий запросов (пакетов) и их резервированного многопутевого распределения через сеть к объединенным в кластер серверам при требовании обеспечения своевременной и безошибочной доставки в кластер хотя бы одной из резервных копий.

Маршрут (путь) от источника запросов до одного из серверов характеризуется последовательностью переходов между коммутационными узлами, представленными одноканальными системами массового обслуживания, с учетом сбоев и отказов, а также ограничений приемо-передающих буферов сетевого оборудования и серверов [14–17].

В отличие от [22, 23], где решались аналогичные задачи, но исследовались заранее выделенные непересекающиеся пути следования пакетов, в данной работе учитывается возможность пересечения резервированных путей в общих коммутационных узлах (рис. 1). При этом выбор очередного участка пути может осуществляться статически – на основе предварительно заданного маршрута, или динамически – например, с использованием алгоритма циклического обслуживания.

На основе рассматриваемой структуры распределенной компьютерной системы (рис. 1) и в продолжение исследований [22, 23] в системе имитационного моделирования AnyLogic 7 построены имитационные модели (рис. 2), различающиеся по способу выбора пути следования пакетов через сеть к кластеру (статический или динамический) и по условию досрочного выхода копий запросов из системы.

Предложенные имитационные модели отражают этапы:

- создания запросов в источнике (source) и их клонирования (req\_clone);
- прохождения запросами пути до сервера через промежуточные коммутационные узлы с возможностью выбора очередного участка пути (path $N$ , где  $N$  – идентификатор элемента модели заданного типа path);
- обслуживания запросов сервером;
- своевременного (успешного) выхода запросов из системы (serviced $N$ ).

В процессе моделирования учитываются: время ожидания запроса в буферах принимающих портов промежуточных узлов ( $r_{qN}$ ) и серверов ( $s_{qN}$ ), а также в буфере передающего порта источника ( $buffer$ ); время буферизации и распространения сигнала ( $delay$ ); время буферизации, распространения сигнала и задержка коммутации ( $rN$ ). Задаются: задержка в коммутаторе (маршрутизаторе) – фиксированная или распределенная согласно экспоненциальному закону, время буферизации и обслуживания доставленной на сервер копии запроса ( $sN$ ), включая проверку ее актуальности ( $relevantN$ ): не превышено ли допустимое время ее пребывания в системе и не была ли уже обслужена другая копия данного запроса.

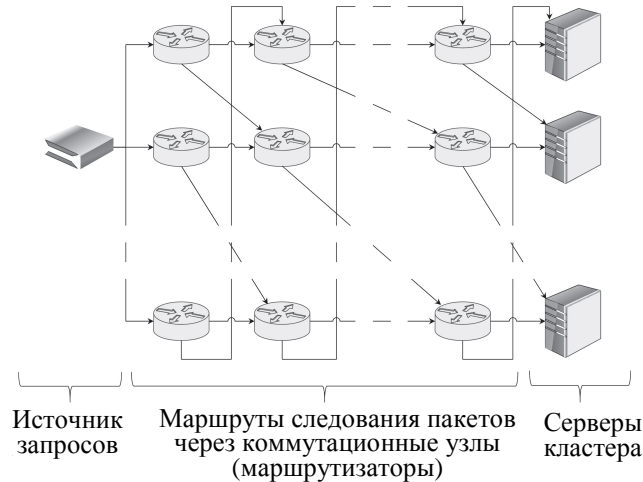


Рис. 1. Структура распределенной системы с многопутевой передачей резервных копий запросов

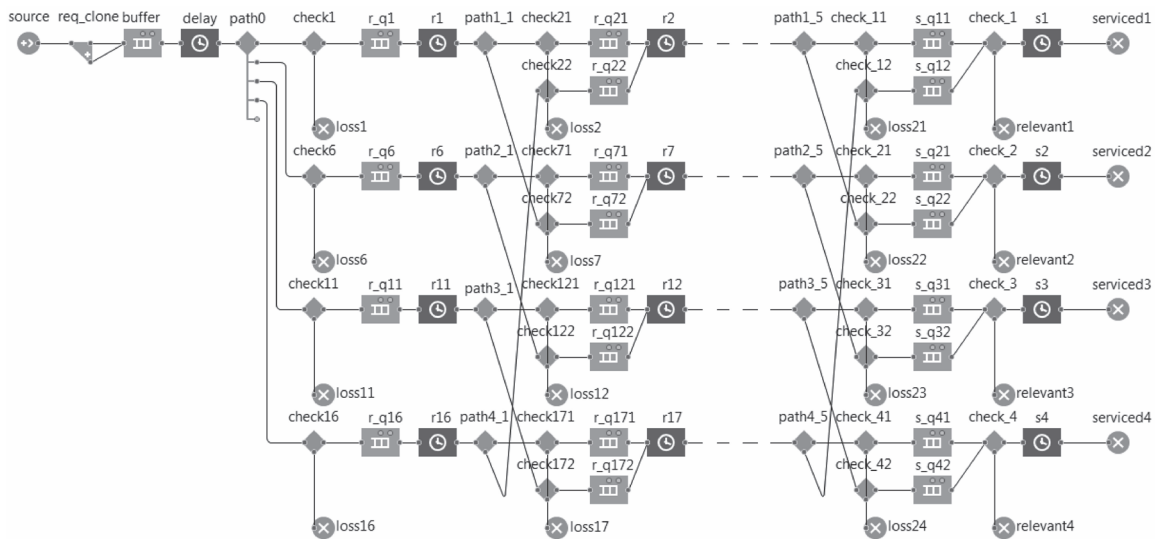


Рис. 2. Схема имитационных моделей резервированных многопутевых передач

При моделировании осуществляется проверка ( $checkN$ ) срабатывания условия досрочного выхода копий запросов из системы ( $lossN$ ) вследствие следующих причин:

- переполнения буферов коммутационных узлов и серверов;
- потерь копий в каналах связи;
- их отбрасывания из-за ошибок, выявленных при вычислении контрольной суммы для пакета;
- любой комбинации перечисленных факторов.

Переполнение буферов определяется превышением максимальной емкости очередей, выраженной в битах; потери копий запросов – заданной вероятностью, а отбрасывание пакетов – вероятностью, вычисляемой на основе заданной вероятности битовых ошибок  $b$  и размера  $N$  в битах передаваемого пакета. Так, вероятность безошибочной передачи пакета  $B$  от одного узла к другому оценивается как  $B=(1-b)^N$ .

В ходе проведенных исследований [22, 23] установлена зависимость вероятности безошибочной и своевременной доставки запросов через сеть в кластер от кратности резервирования передач. Вместе с тем наибольший практический интерес представляет исследование влияния на вероятность успешной доставки интенсивности входного потока запросов – в зависимости от числа пересылаемых копий. Так,

знание оптимальной кратности резервирования передач в заданном диапазоне интенсивностей поступающих запросов позволяет динамически регулировать загрузку системы и повышать вероятность их безошибочной и своевременной доставки.

Для определения области эффективности резервирования при разных значениях интенсивности поступления запросов может понадобиться проведение сотен или тысяч имитационных экспериментов и, как результат, в процессе их многократного последовательного запуска в системе имитационного моделирования (многократных прогонов модели) – значительное количество вычислительных ресурсов.

Результаты промежуточных вычислений (отдельных экспериментов) целесообразно сохранять в базе данных, поскольку это позволяет:

- использовать их много раз, например, в сравнительном анализе с результатами последующих имитационных экспериментов, проводимых на базе измененной модели;
- использовать сохраненные в базе данные сторонними приложениями, независимо от возможностей выбранной системы имитационного моделирования, например, для визуализации обнаруженных зависимостей;
- восстанавливать или частично восстанавливать промежуточные результаты в случае сбоя процесса вычислений (серии прогонов модели), который может длиться часы или даже дни;
- сокращать время выполнения задачи, запуская проводимые в рамках ее решения эксперименты одновременно на разных вычислительных узлах и сохраняя результаты в общую базу данных.

### Проведение имитационных экспериментов с агрегированием результатов

По мере роста и расширения сфер применения имитационных моделей как эффективного инструмента анализа при проектировании сложных технических объектов и систем развивались и системы имитационного моделирования, становясь более удобными и наглядными для пользователя. Одна из таких систем имитационного моделирования – AnyLogic 7 – выбрана для разработки имитационных моделей в этом исследовании, поскольку обладает:

- развитыми средствами визуального проектирования – редактором моделей;
- расширенными возможностями проведения имитационных экспериментов с использованием модуля Custom Experiment («нестандартный эксперимент»);
- возможностью импорта и экспорта данных за счет встроенной поддержки интеграции с известными реляционными системами управления базами данных (СУБД), в том числе Microsoft SQL Server.

Следует отметить, что пользователи могут использовать предлагаемые в данной работе имитационные модели в качестве шаблонов для создания собственных моделей. Например, редактор AnyLogic 7 (рис. 2), поддерживающий функциональность перетаскивания (drag-and-drop), позволяет с легкостью добавлять, переставлять и удалять элементы модели (так называемые блоки) и связи между ними, изменять свойства блоков и, по сути, программную логику.

Применение имитационных моделей в рамках предлагаемого подхода как средства поддержки проектирования высоконадежных распределенных компьютерных систем предполагает выполнение следующих этапов:

- организацию автоматизированного многократного запуска имитационных экспериментов;
- выбор структуры и создание базы данных для сохранения результатов проведенных экспериментов;
- организацию обработки сохраненных результатов, в частности, их обобщение и подготовку для экспорта из базы данных во внешние приложения;
- выбор программных средств и анализ с их помощью экспортируемых результатов.

Автоматизировать проведение множественных имитационных экспериментов можно путем внедрения программного модуля с требуемой логикой в используемую систему имитационного моделирования. В AnyLogic 7 Professional реализован инструмент проведения «нестандартного эксперимента» в полном соответствии с пользовательским сценарием, написанным на языке программирования Java.

В данном исследовании этот сценарий включает в себя:

- создание, инициализацию и запуск движка (simulation engine) для управления прогоном модели, создание экземпляра класса модели и инициализацию ее параметров, как это описано в справочнике по прикладному программному интерфейсу AnyLogic 7 (<https://help.anylogic.com/>);
- сбор статистики, формирование на ее основе записей (кортежей) базы данных с результатами каждого отдельного эксперимента и сохранение этих записей путем обращения к целевой СУБД с помощью SQL-инструкций и с использованием драйвера JDBC;
- перебор значений интенсивности прибытий запросов и кратности резервированных передач в заданных пользователем диапазонах.

Для каждой имитационной модели с параметрами, отражающими требования к моделируемой системе (например, пропускной способности каналов связи или емкости приемо-передающих буферов обслуживания), запускается серия экспериментов.

Под «серией экспериментов» понимается набор последовательно проведенных испытаний с переменными (контролируемыми, наблюдаемыми) и фиксированными параметрами модели. В данном исследовании контролируются интенсивность входного потока запросов и кратность резервирования передач. Остальные параметры составляют конфигурацию системы для рассматриваемой серии экспериментов. Для заданной конфигурации рекомендуется проведение нескольких серий экспериментов с целью уточнения (усреднения, например, путем нахождения среднего арифметического) получаемых результатов – в случае присутствия в имитационной модели параметров со стохастическими характеристиками. Также может быть полезно сравнение результатов экспериментов, проведенных на базе исходной и измененной моделей при неизменных фиксированных параметрах.

Результаты экспериментов, включая число успешно доставленных и обслуженных (*servicedN*) и досрочно покинувших систему (*lossN*, *relevantN*) запросов, в рамках этой работы сохранялись в базе данных Microsoft SQL Server 2008.

На рис. 3 представлены возможные схемы базы данных для хранения результатов имитационных испытаний: в одной таблице (рис. 3, а) и во многих таблицах (рис. 3, б) – по одной на каждую серию экспериментов.

Записи в таблице (таблицах) EXPERIMENT включают в себя информацию о:

- минимальном (*min\_time*), среднем (*mean\_time*) и максимальном (*max\_time*) времени пребывания запросов в системе в случае их успешной доставки и обслуживания;
- доле безошибочно и своевременно переданных через сеть и обработанных запросов (*delivered*);
- доле недоставленных запросов из-за их потерь в каналах связи (*lost*), повреждений пересылаемых данных (*discarded*), переполнения буферов сетевого и вычислительного оборудования (*rejected*), превышения предельного времени ожидания (*expired*);
- интенсивности входного потока запросов (*intensity\_rate*);
- кратности резервированных передач (*redund\_order*).

Таблица CONFIGURATION содержит записи со сведениями о параметрах модели, включая вероятности битовых ошибок (*bit\_err\_rate*), вероятности потери пакетов в каналах связи (*loss\_rate*) и предельное время ожидания запросов в системе (*max\_res\_time*). Время распространения сигнала в каналах связи определяется исходя из типа кабеля и расстояния между промежуточными узлами (*prop\_delay*). Скорость передачи данных соответствует пропускной способности каналов связи (*bandwidth*). Задержка в коммутационном устройстве (маршрутизаторе, *switch\_delay*) может быть фиксирована или задана согласно вероятностному распределению, в частности экспоненциальному; во втором случае необходимо ввести дополнительный атрибут сущности CONFIGURATION, описывающий это распределение. Размер пересылаемого пакета (кадра, *packet\_size*) может быть фиксированным или переменной величины – если он выбирается динамически в процессе выполнения программы, например, из заданного диапазона значений согласно равномерному дискретному распределению. При сохранении записи в базу данных для переменного размера пакета может быть использовано специальное (зарезервированное для этой цели) значение, например, 0. Следует отметить, что сочетания значений атрибутов сущности CONFIGURATION уникальны.

В случае использования единой таблицы для хранения результатов всех экспериментов (рис. 3, а) после проведения очередного испытания с новым сочетанием параметров модели в таблицу CONFIGURATION вносится новая запись, а ссылка на нее (*conf\_id*) – в запись о результатах данного испытания в таблице EXPERIMENT. При этом поле счетчика серий экспериментов (*exp\_num*) содержит 1 (рис. 3, а). При проведении последующих испытаний с тем же набором параметров записей в таблицу CONFIGURATION не делается, но счетчик (номер) серий экспериментов увеличивается. Следует отметить, что уникальность записи в таблице EXPERIMENT определяется комбинацией значений полей *conf\_id*, *exp\_num*, *intensity\_rate* и *redund\_rate*, однако из практических соображений в качестве первичного ключа выбран уникальный идентификатор записи *id*.

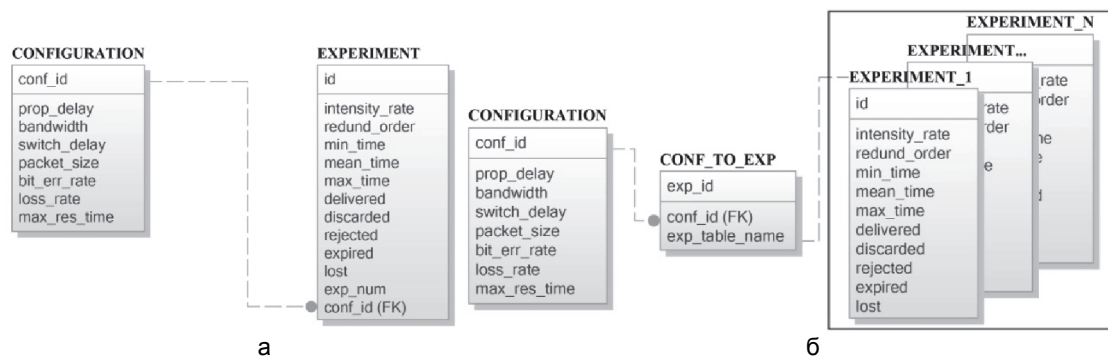


Рис. 3. Схемы базы данных результатов имитационных экспериментов

Работа с базой данных, содержащей сведения о результатах проведенных экспериментов, подразумевает извлечение этих сведений по заданной конфигурации моделируемой системы. Другими словами, данные из таблицы EXPERIMENT извлекаются в соответствии с выбранной записью (записями) из таблицы CONFIGURATION. В случае одновременной работы с таблицей EXPERIMENT многочисленных сессий – как на запись, так и на чтение – производительность СУБД может сильно снизиться, даже в случае использования блокировки отдельных строк («ROWLOCK»). Кроме того, при одновременном проведении испытаний с одной и той же конфигурацией возможно возникновение конфликтных ситуаций, например, при вычислении номера для серии эксперимента.

Альтернативную структуру базы данных с сохранением результатов каждой серии экспериментов в свою таблицу (рис. 3, б) целесообразно выбирать в случае большого числа испытаний в рамках одной серии, когда вероятность безошибочной и своевременной доставки и обслуживания запросов рассматривается в зависимости от широкого диапазона значений интенсивности их поступления.

На рис. 3, б, показана схема базы данных со вспомогательной таблицей CONF\_TO\_EXP, в которую делается запись о вновь созданной таблице EXPERIMENT\_1, EXPERIMENT\_..., EXPERIMENT\_N (и, соответственно, новой серии экспериментов). Эта запись содержит информацию об имени новой таблицы (exp\_table\_name), очевидно, уникальном в пределах CONF\_TO\_EXP, и существующей или вновь созданной конфигурации системы (conf\_id). При таком варианте организации хранения результатов имитационных испытаний может ускориться доступ к данным (по сравнению с вариантом на рис. 3, а) в многопользовательском режиме и упроститься работа с отдельно взятой серией экспериментов. Однако в случае создания множественных таблиц с результатами экспериментов необходимо обеспечить поддержание целостности и согласованности данных – посредством использования каскадных обновлений полей и удалений записей и триггерного механизма при удалении или переименовании этих таблиц, если такая возможность предусмотрена в выбранной СУБД.

Обработка результатов имитационных испытаний предполагает выполнение ряда шагов (рис. 4).

Шаг 1. Сохранение результатов проведения серии (серий) экспериментов в базе данных согласно одному из представленных на рис. 3 вариантов.

Шаг 2. Создание промежуточного представления («VIEW») EXP\_AVG\_V с записями об интенсивности потока запросов (intensity\_rate\_a), кратности резервированных передач (redund\_order\_a), (усредненной по нескольким сериям) доле успешно доставленных и обработанных запросов (delivered\_avg) и при необходимости (усредненном) среднем времени пребывания запросов в системе (mean\_time\_avg).

Шаг 3. Создание представления EXP\_REPRESENT\_V на основе записей из EXP\_AVG\_V для подготовки экспорта данных во внешние приложения путем соотношения интенсивности потока запросов (intensity\_rate\_0) при кратности резервирования от 1 до n с долей доставленных и обслуженных запросов – delivered и средним временем их пребывания в системе – mean\_time.

Шаг 4. Экспорт данных из представления EXP\_REPRESENT\_V в выбранное приложение:

Шаг 4а. Microsoft Excel со встроенной поддержкой работы с СУБД Microsoft SQL Server – для последующих преобразований и анализа,

Шаг 4б. Предлагаемое приложение «Резервирование передач» – для установления зависимости вероятности безошибочной и своевременной передачи копий запросов через сеть P от интенсивности потока запросов  $\Lambda$ ,  $c^{-1}$  при разной кратности резервирования.

Необходимо отметить, что шаги 2 и 3 могут быть совмещены, или шаг 2 может быть пропущен при внесении небольших технических изменений в процесс создания представления на шаге 3 в случае обработки результатов одной серии экспериментов.

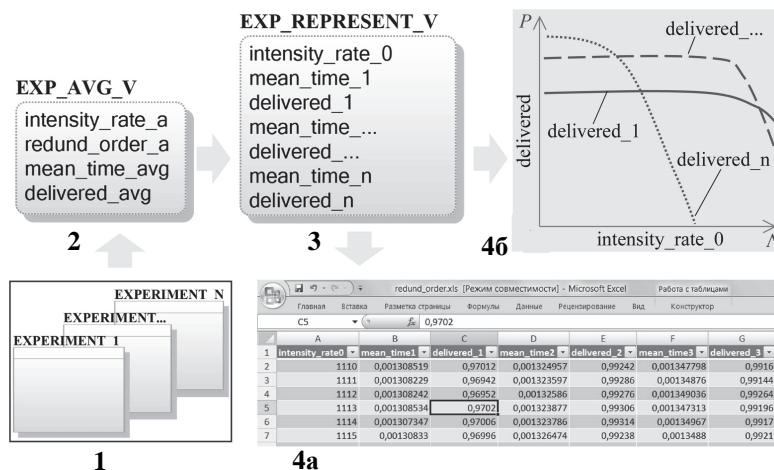


Рис. 4. Организация накопления, обработки и использования результатов имитационных экспериментов

### Автоматизированный поиск оптимальной кратности резервированных передач

Рассмотрим пример использования экспортированных из базы данных результатов имитационных экспериментов в приложении «Резервирование передач» (рис. 5), реализованном на языке программирования Java в рамках данного исследования.

Для определения области эффективности резервированных передач пользователь выбирает интересующие его значения параметров модели, получаемые из таблицы CONFIGURATION, из предложенных вариантов в выпадающих списках. В случае выбора опции «Усредненный результат» используются средние (или преобразованные другим способом) значения данных; иначе шаг 2 опускается, и данные выбираются из произвольной таблицы EXPERIMENT. При нажатии на кнопку «Построить график», если результаты экспериментов для выбранной комбинации параметров не найдены, выдается соответствующее информационное сообщение.

На рис. 5 показаны графики зависимостей вероятности безошибочной и своевременной доставки и обслуживания запросов от интенсивности их поступления в случае без резервирования («Кратность=1») и в случае создания и многопутевой передачи от 2 до 4 резервных копий запросов. Представленные графики построены на основе модели с динамическим распределением запросов через сеть, не ограниченными по размеру буферами коммутационного и вычислительного оборудования и варьирующимся размером пересылаемых кадров – от 64 до 1518 Мбит.

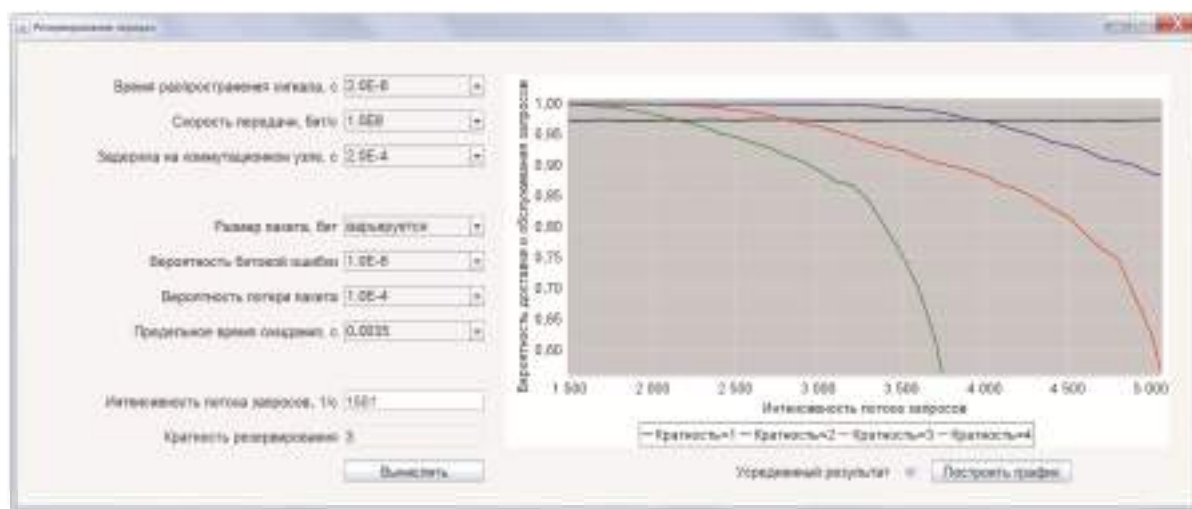


Рис. 5. Графический пользовательский интерфейс разработанного программного средства

При масштабировании можно графически определить область эффективности резервированных передач – наилучшие значения кратности резервирования в заданном диапазоне интенсивностей потока запросов. Вместе с тем может возникнуть задача нахождения оптимальной кратности резервирования для заданной интенсивности поступления запросов, что удобнее сделать программным способом.

В данной работе реализован автоматизированный поиск оптимальной кратности резервирования передач посредством выполнения SQL-запроса к представлению EXP\_AVG\_V на основе алгоритма, включающего в себя нахождение:

- максимальной доли доставленных и обслуженных запросов для заданной интенсивности;
- набора кратностей резервированных передач, соответствующих найденной доле запросов;
- минимальной кратности резервирования из полученного набора.

В поле, обозначенное как «Интенсивность потока запросов, 1/с» (рис. 5), пользователь вводит любое целочисленное значение и нажимает на кнопку «Вычислить». Если заданного значения интенсивности не существует, выбирается ближайшее к нему найденное значение.

### Заключение

Предложен набор имитационных моделей многопутевого распределения копий запросов (пакетов) через сеть и их резервированного обслуживания серверами кластера при пересекающихся в общих коммутационных узлах маршрутах и статическом или динамическом выборе пути следования пакетов.

Рассмотрена зависимость вероятности безошибочной и своевременной передачи и обслуживания хотя бы одной из резервных копий от интенсивности поступления запросов в систему – в условиях их негарантированной доставки из-за потерь в каналах связи, ошибок передач, перегруженности ресурсов или комбинации этих причин.

Разработано программное средство, позволяющее определять область эффективности резервированных передач и осуществлять автоматизированный поиск оптимальной кратности резервирования на основе сохраненных в базе данных результатов имитационных экспериментов.

Представлена организация проведения экспериментов с сохранением в базе данных, обработкой и использованием их результатов во внешних приложениях с целью оценки и обоснования выбора проектных решений по повышению надежности функционирования распределенных компьютерных систем с резервированными передачами и обслуживанием запросов.

### Литература

### References

1. Aysan H. *Fault-Tolerance Strategies and Probabilistic Guarantees for Real-Time Systems*. Malardalen University, Vasteras, Sweden, 2012. 190 p.
2. Kopetz H. *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Springer, 2011. 396 p. doi: 10.1007/978-1-4419-8237-7
3. Koren I., Krishna C.M. *Fault-Tolerant Systems*. San Francisco: Morgan Kaufmann Publishers, 2009. 378 p.
4. Polovko A.M., Гуров С.В. Основы теории надежности. СПб.: БХВ-Петербург, 2006. 704 с.
5. Черкесов Г.Н. Надежность аппаратно-программных комплексов. СПб.: Питер, 2005. 479 с.
6. Bogatyrev V.A. An interval signal method of dynamic interrupt handling with load balancing // *Automatic Control and Computer Sciences*. 2000. V. 34. N 6. P. 51–57.
7. Bogatyrev V.A. Protocols for dynamic distribution of requests through a bus with variable logic ring for reception authority transfer // *Automatic Control and Computer Sciences*. 1999. V. 33. N 1. P. 57–63.
8. Lee M.H., Dudin A.N., Klimenok V.I. The SM/V/N queueing system with broadcasting service // *Mathematical Problem in Engineering*. 2006. V. 2006. Art. 98171. doi: 10.1155/MPE/2006/98171
9. Богатырев В.А., Богатырев А.В. Модель резервированного обслуживания запросов реального времени в компьютерном кластере // *Информационные технологии*. 2016. Т. 22. № 5. С. 348–355.
10. Богатырев А.В., Богатырев В.А. Надежность функционирования кластерных систем реального времени с фрагментацией и резервированным обслуживанием запросов // *Информационные технологии*. 2016. Т. 22. № 6. С. 409–416.
11. Богатырев В.А., Богатырев С.В. Резервированная передача данных через агрегированные каналы в сети реального времени // *Известия высших учебных заведений. Приборостроение*. 2016. Т. 59. № 9. С. 735–740. doi: 10.17586/0021-3454-2016-59-9-735-740
12. Богатырев В.А., Богатырев А.В., Голубев И.Ю., Богатырев С.В. Оптимизация распределения запросов между кластерами отказоустойчивой вычислительной системы // *Научно-технический вестник информационных технологий, механики и оптики*. 2013. № 3 (85). С. 77–82.
13. Богатырев В.А., Богатырев С.В. Надежность мультикластерных систем с перераспределением потоков запросов // *Известия высших учебных заведений. Приборостроение*. 2017. Т. 60. № 2. С. 171–177. doi: 10.17586/0021-3454-2017-60-2-171-177
14. Богатырев В.А., Паршутина С.А. Модели многопутевой отказоустойчивой маршрутизации при распределении запросов через сеть // *Вестник компьютерных и информационных технологий*. 2015. № 12. С. 23–28.
15. Bogatyrev V.A., Parshutina S.A. Redundant distribution of requests through the network by transferring them over multiple paths // *Communications in Computer and Information Science*. 2016. V. 601. P. 199–207. doi: 10.1007/978-3-319-30843-2\_21
16. Богатырев В.А., Паршутина С.А. Многопутевое резервированное распределение через сеть критичных к задержкам запросов // *Вестник компьютерных и информационных технологий*. 2016. № 10 (148). С. 41–46. doi: 10.14489/vkit.2016.10.pp.041-046
17. Bogatyrev V.A., Parshutina S.A. Efficiency of redundant multipath transmission of requests through the network to destination servers // *Communications in Computer and*
1. Aysan H. *Fault-Tolerance Strategies and Probabilistic Guarantees for Real-Time Systems*. Malardalen University, Vasteras, Sweden, 2012, 190 p.
2. Kopetz H. *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Springer, 2011, 396 p. doi: 10.1007/978-1-4419-8237-7
3. Koren I., Krishna C.M. *Fault Tolerant Systems*. San Francisco, Morgan Kaufmann Publishers, 2009, 378 p.
4. Polovko A.M. Gurov S.V. *Osnovy Teorii Nadezhnosti [Fundamentals of Reliability Theory]*. St. Petersburg, BHV-Peterburg Publ., 2006, 704 p. (In Russian)
5. Cherkesov G.N. *Nadezhnost Apparatno-Programmykh Kompleksov [Reliability of Hardware and Software Systems]*. St. Petersburg, Piter Publ., 2005, 479 p. (in Russian)
6. Bogatyrev V.A. An interval signal method of dynamic interrupt handling with load balancing. *Automatic Control and Computer Sciences*, 2000, vol. 34, no. 6, pp. 51–57.
7. Bogatyrev V.A. Protocols for dynamic distribution of requests through a bus with variable logic ring for reception authority transfer. *Automatic Control and Computer Sciences*, 1999, vol. 33, no. 1, pp. 57–63.
8. Lee M.H., Dudin A.N., Klimenok V.I. The SM/V/N queueing system with broadcasting service. *Mathematical Problem in Engineering*, 2006, vol. 2006, art. 98171. doi: 10.1155/MPE/2006/98171
9. Bogatyrev V.A., Bogatyrev A.V. The model of redundant service requests real-time in a computer cluster. *Informacionnye Tehnologii*, 2016, vol. 22, no. 5, pp. 348–355. (In Russian)
10. Bogatyrev V.A., Bogatyrev A.V. The reliability of the cluster real-time systems with fragmentation and redundant service requests. *Informacionnye Tehnologii*, 2016, vol. 22, no. 6, pp. 409–416. (In Russian)
11. Bogatyrev V.A., Bogatyrev S.V. Redundant data transmission using aggregated channels in real-time network. *Journal of Instrument Engineering*, 2016, vol. 59, no. 9, pp. 735–740. (in Russian). doi: 10.17586/0021-3454-2016-59-9-735-740
12. Bogatyrev V.A., Bogatyrev A.V., Golubev I.Yu., Bogatyrev S.V. Queries distribution optimization between clusters of fault-tolerant computing system. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2013, no. 3, pp. 77–82.
13. Bogatyrev V.A., Bogatyrev S.V. Reliability of multi-cluster systems with redistribution of the flow of requests. *Journal of Instrument Engineering*, 2017, vol. 60, no. 2, pp. 171–177. (In Russian). doi: 10.17586/0021-3454-2017-60-2-171-177
14. Bogatyrev V.A., Parshutina S.A. Multipath fault-tolerant routing models for distributing queries through the network. *Herald of Computer and Information Technologies*, 2015, no. 12, pp. 23–28.
15. Bogatyrev V.A., Parshutina S.A. Redundant distribution of requests through the network by transferring them over multiple paths. *Communications in Computer and Information Science*, 2016, vol. 601, pp. 199–207. doi: 10.1007/978-3-319-30843-2\_21
16. Bogatyrev V.A., Parshutina S.A. Multipath redundant distribution of delay-sensitive queries through the network. *Herald of Computer and Information Technologies*, 2016, no. 10, pp. 41–46. (In Russian) doi: 10.14489/vkit.2016.10.pp.041-046
17. Bogatyrev V.A., Parshutina S.A. Efficiency of redundant multipath transmission of requests through the network to destination servers. *Communications in Computer and*



- Information Science, 2016, V. 678, P. 290–301. doi: 10.1007/978-3-319-51917-3\_26
18. Кутузов О.И., Татарникова Т.М. Инфокоммуникационные сети. Моделирование и оценка вероятностно-временных характеристик. СПб.: ГУАП, 2015. 381 с.
  19. Кутузов О.И., Татарникова Т.М. К анализу парадигм имитационного моделирования // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17. № 3. С. 552–558. doi: 10.17586/2226-1494-2017-17-3-552-558
  20. Карпов Ю.Г. Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. СПб.: БХВ-Петербург, 2005. 403 с.
  21. Алиев Т.И. Основы моделирования дискретных систем. СПб.: СПбГУ ИТМО, 2009. 363 с.
  22. Богатырев В.А., Кармановский Н.С., Попцова Н.А., Паршутина С.А., Воронина Д.А., Богатырев С.В. Имитационная модель поддержки проектирования инфокоммуникационных резервированных систем // Научно-технический вестник информационных технологий, механики и оптики. 2016. Т. 16. № 5. С. 831–838. doi: 10.17586/2226-1494-2016-16-5-831-838
  23. Bogatyrev V.A., Parshutina S.A., Poptcova N.A., Bogatyrev A.V. Efficiency of redundant service with destruction of expired and irrelevant request copies in real-time clusters // Communications in Computer and Information Science. 2016, V. 678, P. 337–348. doi: 10.1007/978-3-319-51917-3\_30
  - Information Science, 2016, vol. 678, pp. 290–301. doi: 10.1007/978-3-319-51917-3\_26
  18. Kutuzov O.I., Tatarnikova T.M. *Infocommunication Networks. Simulation and Evaluation of Probability-Time Characteristics*. St. Petersburg, SUAI Publ., 2015, 381 p. (In Russian)
  19. Kutuzov O.I., Tatarnikova T.M. On the simulation paradigm analysis. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2017, vol. 17, no. 3, pp. 552–558. (In Russian) doi: 10.17586/2226-1494-2017-17-3-552-558
  20. Karpov Yu.G. *Simulation Modeling of Systems. Introduction to Modeling with AnyLogic 5*. St. Petersburg, BKhV-Peterburg Publ., 2005, 403 p. (In Russian)
  21. Aliev T.I. *Osnovy Modelirovaniya Diskretnykh Sistem* [Basics of Discrete Systems Modeling]. St. Petersburg, NRU ITMO Publ., 2009, 363 p. (In Russian)
  22. Bogatyrev V.A., Karmanovsky N.S., Poptcova N.A., Parshutin S.A., Voronina D.A., Bogatyrev S.V. Simulation model for design support of infocomm redundant systems. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2016, vol. 16, no. 5, pp. 831–838. (In Russian). doi: 10.17586/2226-1494-2016-16-5-831-838
  23. Bogatyrev V.A., Parshutina S.A., Poptcova N.A., Bogatyrev A.V. Efficiency of redundant service with destruction of expired and irrelevant request copies in real-time clusters. *Communications in Computer and Information Science*, 2016, vol. 678, pp. 337–348. doi: 10.1007/978-3-319-51917-3\_30

#### Авторы

**Паршутина Светлана Александровна** – аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, svetlana.parshutina@gmail.com

#### Authors

**Svetlana A. Parshutina** – postgraduate, ITMO University, Saint Petersburg, 197101, Russian Federation, svetlana.parshutina@gmail.com