

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ
имени Гумарбека Даукеева

Кафедра «Телекоммуникационные сети и системы»

Специальность: 6М071900 «Радиотехника, электроника и телекоммуникации»

ДОПУЩЕН К ЗАЩИТЕ
Зав. кафедрой
PhD, доцент Темырканова Э.К.
(ученая степень, звание, ФИО)

(подпись)

« _____ » _____ 2020 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
пояснительная записка

на тему: «Планирование экспериментов с имитационными моделями»

Магистрант: Сисенова А.С. _____ группа МРЭТн 18-2
(Ф.И.О.) (подпись)

Руководитель: к.т.н., профессор АУЭС _____ Туманбаева К.Х.
(ученая степень, звание) (подпись) (Ф.И.О.)

Рецензент _____ _____
(ученая степень, звание) (подпись) (Ф.И.О.)

Консультант по ВТ к.т.н., профессор АУЭС _____ Туманбаева К.Х.
(ученая степень, звание) (подпись) (Ф.И.О.)

Нормоконтроль: к.т.н., профессор АУЭС _____ Туманбаева К.Х.
(ученая степень, звание) (подпись) (Ф.И.О.)

Алматы 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ
имени Гумарбека Даукеева

Институт Космической Инженерии и Телекоммуникаций

Специальность: 6М071900 «Радиотехника, электроника и телекоммуникации»

Кафедра: «Телекоммуникационные сети и системы»

ЗАДАНИЕ

на выполнение магистерской диссертации

Магистранту Сисеновой Айнамкөз Сейілханқызы

Тема диссертации «Планирование экспериментов с имитационными моделями»

Утверждена Ученым советом университета №«122»от «25» октября _____
Срок сдачи законченной диссертации «25»_мая 2020г.

Цель исследования состоит в исследовании возможности планирования экспериментов с имитационными моделями и проведении дисперсионного анализа в среде GPSS World.

Перечень подлежащих разработке в магистерской диссертации вопросов или краткое содержание магистерской диссертации:

1. Моделирование телекоммуникационных сетей и систем
2. Имитационное моделирование в управлении сложными телекоммуникационными системами
3. Имитационное моделирование системы массового обслуживания М/М/5/0 в системе GPSS Studio
4. Эксперименты с имитационными моделями

Перечень графического материала (с точным указанием обязательных чертежей)

1. Элементы системы массового обслуживания
2. Структурная модель М/М/5/0
3. Настройка группы смешивания и стандартной процедуры
4. Результаты дисперсионного анализа

Рекомендуемая основная литература

1. Боев В.Д. Моделирование систем. Инструментальные средства GPSS World: Учеб. пособие. – СПб.: БХВ-Петербург, 2004.

2. A. M. Law and W. D. Kelton, Simulation modelling and analysis, third ed. New York: McGraw-Hill, 2017.

Г Р А Ф И К
подготовки магистерской диссертации

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
1. Информационный обзор согласно теме	01.10.2018	
2. Анализ современного состояния планирования экспериментов	30.03.2019	
3. Моделирование систем телекоммуникаций как систем массового обслуживания	17.10.2019	
4. Эксперименты с имитационными моделями	07.01.2020	
5. Анализ полученных экспериментальных и расчетных данных	10.05.2020	

Дата выдачи задания_30 сентября 2018г._____

Заведующий кафедрой _____ (Темырканова Э.К.)
(подпись) (Ф.И.О.)

Научный
руководитель диссертации _____ (Туманбаева К.Ч.)
(подпись) (Ф.И.О.)

Задание принял к исполнению
магистрант _____ (Сисенова А.С.)
(подпись) (Ф.И.О.)

Аннотация

В диссертационной работе рассматривается вопрос планирование экспериментов с имитационными моделями. В работе исследовано возможности планирования экспериментов с имитационными моделями в среде GPSS World. Разработана модель телекоммуникационной системы как системы массового обслуживания (СМО). Проведен анализ современного состояния планирования экспериментов. Приведен дисперсионный анализ имитационной модели в среде GPSS World. В ходе исследования построена имитационная модель эксперимента в программе GPSS WORLD.

Аңдатпа

Бұл диссертациялық жұмыста имитациялық модельдермен эксперименттерді жоспарлау мәселесі қарастырылды. Жұмыста GPSS World ортасында имитациялық модельдермен эксперименттерді жоспарлау мүмкіндіктері зерттелді. Телекоммуникациялық жүйенің жаппай қызмет көрсету жүйесінің моделі әзірленді. Эксперименттерді жоспарлаудың қазіргі жай-күйіне талдау жүргізілді. GPSS World ортасында имитациялық модельдің дисперсиялық талдауы келтірілді. Зерттеу барысында GPSS WORLD бағдарламасында эксперименттің имитациялық моделі құрылды.

Abstract

In the dissertation work, the issue of planning experiments with simulation models is considered. The paper investigates the possibilities of planning experiments with simulation models in the GPSS World environment. A model of a telecommunications system as a Queuing system (QMS) has been developed. The analysis of the current state of experiment planning is carried out. The variance analysis of the simulation model in the GPSS World environment is presented. In the course of the study, a simulation model of the M2M network was built in the GPSS WORLD program.

Содержание

Введение	6
1 Анализ современного состояния планирования экспериментов	7
1.1 Планирование экспериментов в ТКС	7
1.2 Моделирование телекоммуникационных сетей и систем.....	9
1.3 Имитационное моделирование в управлении сложными телекоммуникационными системами	14
2 Моделирование систем телекоммуникаций как систем массового обслуживания	21
2.1 Основные понятия систем массового обслуживания.....	21
2.2 Имитационное моделирование системы массового обслуживания М/М/5/0 в системе GPSS Studio	24
2.3 Моделирование системы массового обслуживания для анализа производительности IoT-приложений	29
2.4 Дисперсионный анализ в GPSS	36
3 Эксперименты с имитационными моделями.....	43
3.1 Стратегическое планирование эксперимента	43
3.2 Тактическое планирование эксперимента	46
3.3 Дисперсионный анализ (отсеивающий эксперимент) на примере системы GPSS World	48
Заключение	59
Список литературы	60
Приложение А Результаты моделирования в среде GPSS.....	62
Приложение Б The run execution procedure in GPSS	65
Приложение В Справка Антиплагиата	72

Введение

Огромный рост использования телекоммуникационных сетей для предоставления жизненно важных услуг заставил нас уделять больше внимания вопросам, связанным с планированием пропускной способности сетей.

Имитационное моделирование стала популярной среди исследователей и разработчиков компьютерных и телекоммуникационных сетей во всем мире. Эта популярность обусловлена наличием различных сложных и мощных пакетов моделирования, а также гибкостью в построении моделей и валидации, предлагаемых симуляцией. Методологий сетевого моделирования часто используются при верификации аналитических моделей, обобщении результатов измерений, оценки эффективности разрабатываемых новых протоколов и сравнении существующих протоколов. Обычные аналитические методы используют обширные математические модели, которые требуют допущений и ограничений, накладываемых на модель. Это может привести к неизбежной неточности в выходных данных. Моделирование позволяет избежать наложения ограничений на систему, а также учитывать случайные процессы; фактически в некоторых случаях моделирование является единственным применимым практическим методом моделирования. Все это определяет актуальность выбранной темы, которая позволила бы с наименьшими затратами получить имитационную модель сети и в то же время была бы простой в использовании и не требовала значительных вложений на этапе внедрения.

Целью работы является исследование возможности планирования экспериментов с имитационными моделями в среде GPSS World.

Для достижения поставленной цели необходимо решить следующие задачи:

- провести анализ современного состояния планирования экспериментов;
- разработать модель телекоммуникационной системы как системы массового обслуживания (СМО);
- провести дисперсионный анализ имитационной модели в среде GPSS World.

Практическая ценность диссертационной работы заключается в использовании разработанной методики для дисперсионного анализа имитационной модели.

1 Анализ современного состояния планирования экспериментов

1.1 Планирование экспериментов

Сфера телекоммуникации, как и все отрасли науки, в настоящий момент переживает период быстрого развития. Одновременно с классическими видами услуг телекоммуникации активно вырабатываются и другие сферы, такие как мобильная связь, электронный бизнес и торговля; интернет; мультимедийные технологии и др. Традиционные методы исследования связаны с экспериментами, требующими огромных ресурсов, сил и средств, поскольку они "пассивны" - основаны на чередующемся изменении отдельных независимых переменных в условиях, когда остальные имеют тенденцию оставаться неизменными.

Используя планирование эксперимента наблюдающий делает свое поведение более целенаправленным и организованным, повышает производительность труда и достоверность полученных результатов. Важные преимущества планирование эксперимента - это универсальность и пригодность в значительно больших области исследований.

Компьютерное моделирование наиболее полезно при использовании в сочетании с результатами аналитических и аппаратных измерений. Результат аналитических методов может быть уточнен с помощью компьютерного моделирования, где более детальная модель системы строится с использованием компьютерного программного обеспечения. На всех этапах проектирования системы моделирование может играть важную роль от первоначального концептуального проектирования до различных этапов технико-экономического обоснования, тематических исследований, детального проектирования, тестирования и внедрения [3].

Моделирование производительности с помощью имитационного моделирования обычно включает следующие этапы:

- а) получение действительной имитационной модели системы;
- б) разработка, тестирование и проверка программного обеспечения для моделирования;
- в) получение и анализ данных.

Моделирование систем связи первоначально использовалось в дорогостоящих системах, таких как геостационарные спутники, или в других зонах повышенного риска. Моделирование в настоящее время играет важную, гораздо более широкую роль в проектировании, внедрении и моделировании производительности систем связи. Моделирование использовалось для изучения различных аспектов телекоммуникационных систем (оценка протоколов, проектирование трафика, определение размеров, анализ BER, исследование частоты ошибок кадров, исследование соотношения S/N и др.). Он также был использован для изучения голоса, видео, данных и интегрированного трафика как в фиксированных, так и в беспроводных сетях.

К. Эрланг впервые заметил, что телефонная система может быть смоделирована с использованием пуассоновских прибытий и экспоненциального времени удержания вызова [4]. Поскольку существует определенное количество каналов и вызовы не стоят в очереди, голосовая сеть может быть смоделирована как система массового обслуживания М/М/К/К, где К-количество каналов (серверов), каждый из которых обслуживает вызов. Вероятность потерь по сравнению с трафиком Эрланга сведена в таблицы (таблицы трафика) и представлена в виде графиков в телекоммуникационных текстах. Моделирование с использованием системы массового обслуживания М/М/К/К дает те же результаты, что и Формула Эрланга-В.

Сеть передачи данных можно рассматривать как сеть очередей. Задержка и пропускная способность локальных сетей передачи данных анализируются с использованием модели системы массового обслуживания.

Чтобы проводить исследования, нужно уметь влиять на поведении исследуемого "черного ящика". Методы такого воздействия обозначаются буквой (x) и называются факторами или входами в "черный ящик".

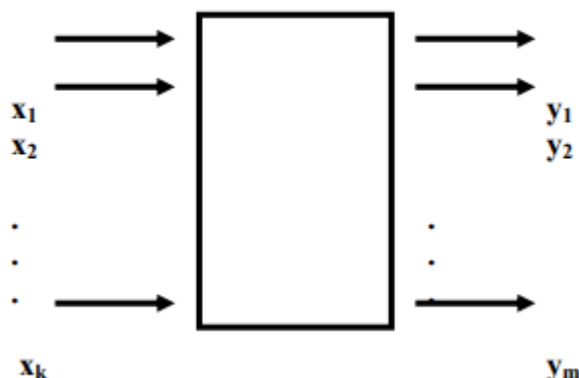


Рисунок 1.1 – Модель объекта исследования

Математические модели применяются при решении основных задач в телекоммуникации. Математическая модель - это уравнение, которое соединяет параметр оптимизации с значениями факторов. Данное уравнение в общем случае можно написать следующим образом:

$$y = \varphi(x_1, x_2, \dots, x_k),$$

где символ $\varphi()$ означает слова: «функция от». Это функция является функцией отклика.

В исследовании один фактор принимает только одно значение из нескольких. Эти значения называются уровнями. С целью упростить модели «черного ящика» и эксперимента фактор должен иметь определенное число дискретных уровней. Значение «черного ящика» определяется стационарным набором уровней. Одновременно это является условием проведения одного из

возможных опытов. Если анализировать все вероятные наборы состояний, то будем иметь множество многочисленных состояний «черного ящика». При этом это равняется числу возможных различных исследований [2].

Количество возможных исследований можно вычислить по формуле:

$$N=p^k,$$

где N – количество опытов;
 p – количество уровней;
 k – количество факторов.

При проведении опытов с планированием эксперимента требуется выполнение определенных условий. Главное условие – это воспроизводимость результата опытов и подконтрольность экспериментов. При повторном проведении некоторых опытов через неравные промежутки времени и сравнивать результаты, в данном случае – значения параметра оптимизации, то разброс их значений определяет воспроизводимость результатов. Объект считается удовлетворенным требованиям воспроизводимости результатов, если это значение не превышает некоторых заданных значений. В своей работе я рассматривала только те объекты, где это условие выполняется.

Для моделирования телекоммуникационных сетей обычно используются два вида моделирования, а именно: дискретное и непрерывное моделирование. Дискретное моделирование также известно как дискретное моделирование событий и представляет собой динамические стохастические системы, основанные на событиях. Другими словами, система содержит ряд состояний и моделируется с помощью набора переменных. Если значение переменной изменяется, это представляет собой событие и отражается в изменении состояния системы. Поскольку система динамична, она постоянно меняется, и поскольку она стохастична, в ней присутствует элемент случайности. Представление дискретного моделирования осуществляется с помощью уравнений состояния, содержащих все переменные, влияющие на систему.

Непрерывное моделирование также содержит переменные состояния; однако они непрерывно меняются со временем. Непрерывное моделирование обычно моделируется с помощью дифференциальных уравнений, которые отслеживают состояние системы относительно времени.

1.2 Моделирование телекоммуникационных сетей и систем

Моделирование является научным методом познания, где вместо исследуемого объекта используется более простой объект, называемый моделью. Есть 2 основные методы моделирования: математическое и физическое. При физическом (натурном) моделировании исследуемый объект заменяется лишь другим материальным объектом, который сохраняет свойства исследуемой системы, но при этом сохраняется их физическая природа. Как пример этого вида можно рассмотреть пилотную сеть, где изучаем принципиальную возможность построения сети на основе, которых имеем

компьютеры, коммуникационные средства, операционные системы и приложения.

Обычные аналитические методы используют обширные математические модели, которые требуют допущений и ограничений, накладываемых на модель. Это может привести к неизбежной неточности в выходных данных. Моделирование позволяет избежать наложения ограничений на систему, а также учитывать случайные процессы; фактически в некоторых случаях моделирование является единственным применимым практическим методом моделирования[1][2].

Аналитики могут детально изучить взаимосвязи между компонентами и смоделировать прогнозируемые последствия нескольких вариантов проектирования, прежде чем реализовать результат в реальном мире.[1][2]

Можно легко сравнить альтернативные конструкции, чтобы выбрать оптимальную систему.

Сам процесс разработки моделирования может дать ценную информацию о внутренней работе сети, которая, в свою очередь, может быть использована на более позднем этапе[1].

Недостатки имитационного моделирования:

- Точная разработка имитационной модели требует значительных ресурсов.[1][2]

- Результаты моделирования так же хороши, как и модель, и как таковые все еще являются только оценками / прогнозируемыми результатами.[1]

- Оптимизация может быть выполнена только с использованием нескольких альтернатив, поскольку модель обычно разрабатывается с использованием ограниченного числа переменных.[1][2]

- Моделирование стоит больших денег, чтобы построить и очень дорого сделать

- Статистические вопросы имитационного моделирования

Имитационные модели создаются на основе набора данных, полученных из стохастической системы. Необходимо проверить, что данные статистически достоверны, установив статистическое распределение, а затем проверить значимость такого соответствия. Кроме того, как и в любом процессе моделирования, точность входных данных должна быть проверена, а любые выбросы должны быть удалены[1].

После завершения моделирования необходимо проанализировать полученные данные. Выходные данные моделирования дадут только вероятную оценку реальных событий. Методы повышения точности выходных данных включают в себя: многократное выполнение моделирования и сравнение результатов, разделение событий на пакеты и обработку их по отдельности, а также проверку того, что результаты моделирования, проведенного в смежных временных периодах, “соединяются” для получения целостного целостного представления о системе[1][4].

Поскольку большинство систем связано со стохастическими процессами, моделирование часто использует генераторы случайных чисел для создания

входных данных, которые аппроксимируют случайную природу реальных событий. Сгенерированные компьютером [случайные числа] обычно не являются случайными в самом строгом смысле, поскольку они вычисляются с помощью набора уравнений. Такие числа называются псевдослучайными числами. При использовании псевдослучайных чисел аналитик должен убедиться, что истинная случайность чисел проверена. Если обнаружится, что числа ведут себя недостаточно случайным образом, необходимо найти другой метод генерации. Случайные числа для моделирования создаются генератором случайных чисел.

Функционирование сетей связи обычно оценивается с помощью методов моделирования, которые, в отличие от аналитических методов, используют меньше допущений. Сложность современных систем связи является еще одной движущей силой широкого использования методов имитационного моделирования [1, 2]. Моделирование позволяет избежать ошибок, связанных со сложным математическим анализом. В большинстве случаев они оценивают влияние случайных входных сигналов, таких как случайные задержки пути в беспроводных системах.

Альтернативой для математического моделирования сложных систем может стать имитационное моделирование. В телекоммуникации этот вид моделирования считается самым лучшим или даже единственным методом исследования реальных систем.

Имитационные модели можно классифицировать исходя из следующих наиболее распространенных признаков:

- по способу взаимодействия с пользователем;
- способ изменения модельного времени;
- цель эксперимента.

Схема классификации имитационных моделей показана на рисунке 2.



Рисунок 1.2 - Классификация имитационных моделей

Достижения в области компьютерных систем позволили создавать и запускать сложные компьютерные имитационные модели. Однако важно построить имитационную модель, которая наилучшим образом описывает реальную систему. Имитационная модель может быть построена с различной степенью сложности. В зависимости от требуемой степени точности, времени, отведенного на построение модели, и принятого времени моделирования должно быть принято решение о том, насколько сложной должна быть модель [3]. Если модель слишком упрощена, она может отменить некоторые жизненно важные характеристики системы, которые могут привести к неверным результатам.

В имитационной модели задаются такие параметры моделирования, как начальное число генерации случайных чисел, требуемая точность (или длина моделирования), количество выборок за итерацию (для рекурсивного или пакетного моделирования) и параметры системы.

При моделировании необходимо генерировать случайные числа, обусловленные случайностью входных данных в реальной системе. Большинство алгоритмов создают случайные числа в формате [5]

$$X_n = (bX_{n-1} + c)m$$

где X_n число в последовательности, b , c и m -соответственно выбранные константы. Приведенное выше уравнение генерирует положительные случайные числа между нулем и m . для генерации случайных чисел между нулем и единицей следует использовать. То случайные числа, сгенерированные между нулем и единицей, распределены равномерно. Для получения

случайных чисел с другими распределениями, например экспоненциальными, эти однородные случайные числа преобразуются с помощью соответствующих формул.

Важность написания и тестирования фактического кода не следует воспринимать легкомысленно. Это обычно не является простым делом даже для опытных программистов [5]. Ошибка в программном коде может привести к неверным результатам моделирования. По этой причине и из-за времени, необходимого для создания хорошо написанной имитационной модели, могут быть использованы некоторые заранее написанные программы. Примеры моделирования являются: Я думаю, MATLAB и прокомментируешь ситуацию. Ithink-это предварительно написанное программное обеспечение для моделирования с графическим средством, которое может быть использовано для моделирования последующих событий. Программное обеспечение Matlab используется для моделирования коммуникационных систем и методов с использованием моделирования методом Монте-Карло. Commnet - это еще одно предварительно написанное программное обеспечение для моделирования с графическим средством, которое используется для моделирования телекоммуникационных систем. Он может быть использован для анализа трафика и определения размеров телекоммуникационных систем.

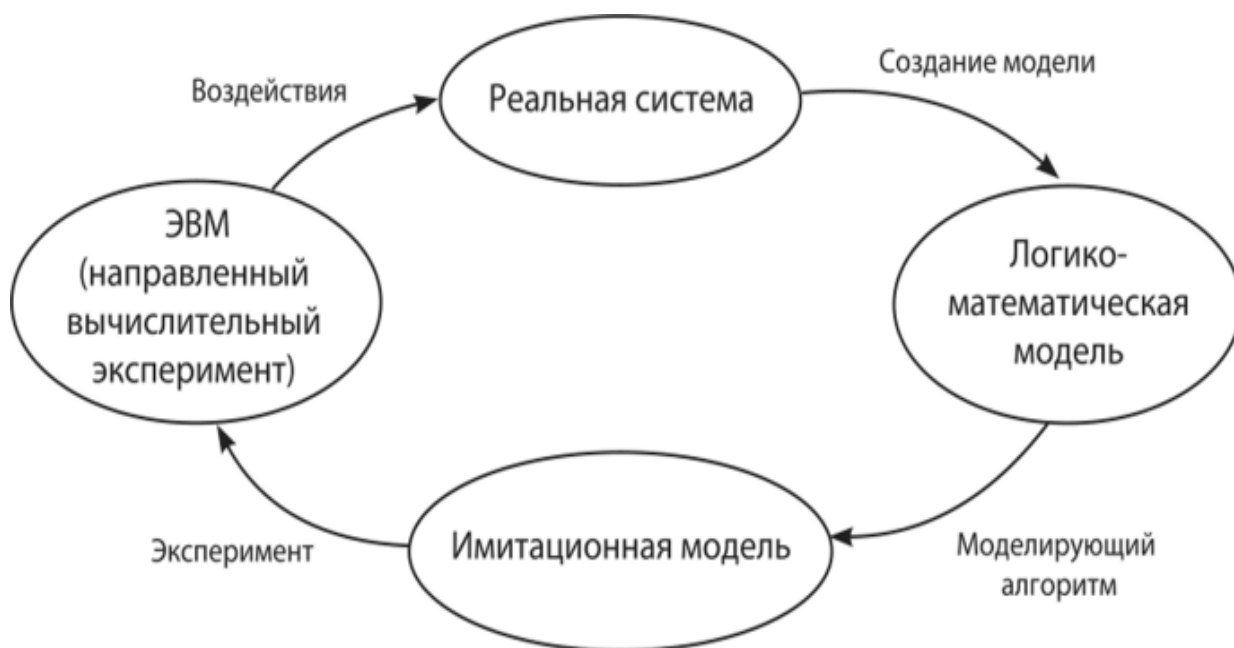


Рисунок 1.3 – Процесс имитационного моделирования

Моделирование для случая, когда входные данные фиксированы таким образом, что могут быть получены идентичные результаты, называется детерминированным моделированием [3]. Моделирование, в котором случайные величины изменяются со временем, называется стохастическим моделированием [1, 10]. Статистическое поведение случайной величины определяет точность полученных результатов [4]. Стохастический процесс -

это набор случайных величин, которые можно индексировать в зависимости от времени. $X(t)$ - состояние системы в момент времени t . например, количество вызовов в телефонной системе изменяется в зависимости от функции времени. Если время имеет дискретные значения, например, $0,1,2,\dots$, процесс классифицируется как стохастический дискретно-параметрический процесс.

С другой стороны, если время может принимать любое значение в интервале, то этот процесс называется стохастическим непрерывно параметрическим процессом [4]. Поскольку в стохастическом моделировании входные данные являются случайными величинами, выходные результаты могут варьироваться в зависимости от длины моделирования. Анализ полученных результатов необходимо проводить в рамках стохастического моделирования. Кроме того, имитационная модель является абстрактной моделью, и результаты не столь детерминированы, как результаты аппаратной модели *protуре*. Моделирование должно быть выполнено много раз, с различными случайными числами, чтобы обеспечить выборочные результаты для анализа [5]. Традиционный метод прямого стохастического моделирования не может быть эффективно использован, если требуемые события встречаются редко. Например, если вероятность возникновения события равна 10^{-10} (например, BER), то для получения нескольких событий потребуется много времени. В таких случаях более уместны редкие методы моделирования событий.

Метод прямого стохастического моделирования может быть использован для анализа ситуаций, которые приводят к вероятностям выше, чем 10^{-5} . Однако моделирование событий, которые не происходят, часто занимает очень много времени. Например, BER в системах связи может быть очень низким 10^{-10} [11].

Другим примером является вероятность потери сотовой связи в сетях банкоматов, которая также очень низка, в порядке $10^{-6} - 10^{-12}$, в зависимости от типа услуги (голос, видео или данные). Чтобы запустить традиционные методы моделирования для таких приложений, потребуется много времени. Можно ожидать, что число наблюдений должно быть, по крайней мере, в сто раз больше вероятности потерь, чтобы получить результаты с разумной уверенностью [12]. Это означает, что для получения достоверной оценки требуемой вероятности потерь для сетей АТМ необходимо смоделировать хотя бы статистически независимые ячейки.

1.3 Имитационное моделирование в управлении сложными телекоммуникационными системами

Существенные изменения в телекоммуникационных технологиях и компьютерных технологиях привели к развитию телекоммуникационных систем (ТКС). В последствии стремительного развития вычислительных возможностей, необходимости обеспечения мобильной связи, электронного бизнеса и торговли, мультимедийных технологий и др. привели к появлению

географически рассредоточенных и сложных информационно-коммуникационных сетей.

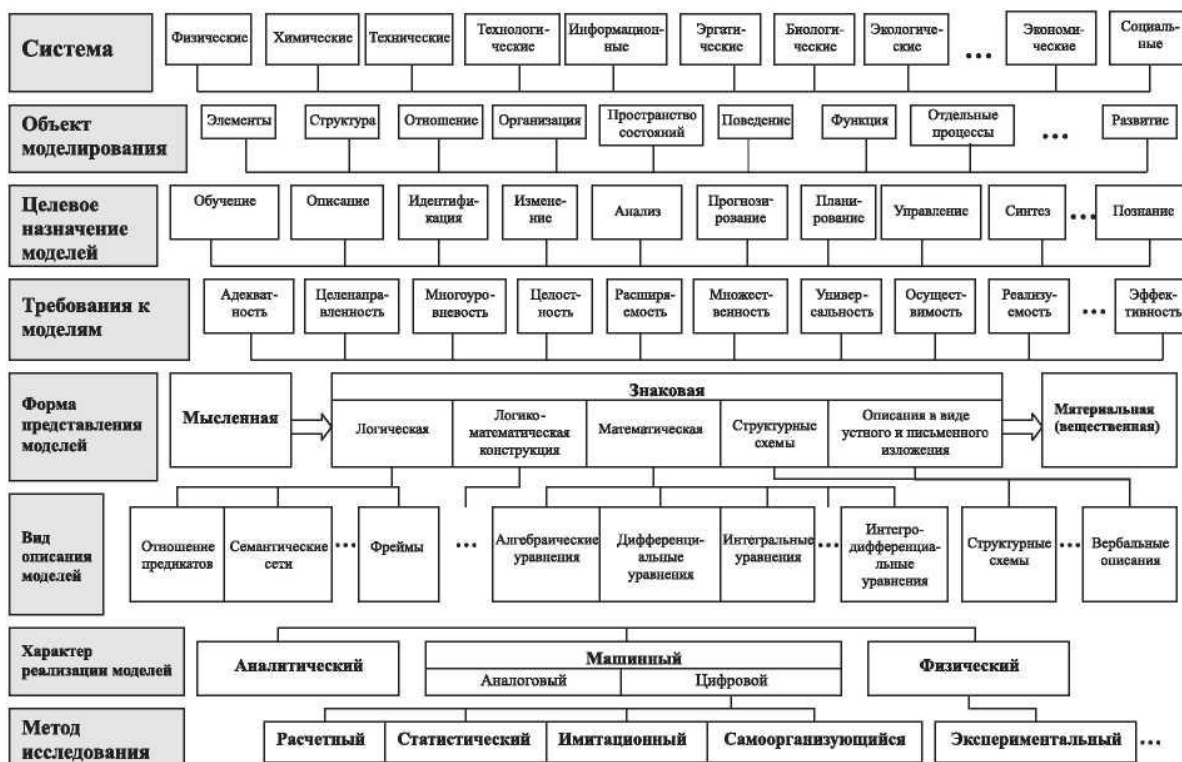


Рисунок 1.4 – Общая процедурно-технологическая схема построения и исследования сложных систем

В телекоммуникационных сетях, как и во многих других областях науки и техники, распространение компьютеров в качестве исследовательских инструментов привело к принятию компьютерного моделирования в качестве наиболее часто используемой парадигмы научных исследований. Это, вместе с множеством существующих языков моделирования и пакетов, создало популярное мнение, что моделирование-это в основном упражнение в компьютерном программировании. В новых вычислительных средах программирование может быть сведено к минимуму или даже полностью заменено манипуляцией значками (представляющими собой готовые программные объекты, содержащие основные функциональные блоки моделируемых систем) на мониторе компьютера. Можно сказать, что мы стали свидетелями еще одного успеха современной науки и техники: появления замечательных и мощных инструментов для исследования и прогнозирования поведения таких сложных, стохастических динамических систем, как телекоммуникационные сети.

Но этот энтузиазм разделяют далеко не все исследователи в этой области. Распространено мнение, что нельзя полагаться на большинство опубликованных результатов исследований оценки эффективности телекоммуникационных сетей, основанных на стохастическом моделировании,

поскольку они не заслуживают доверия. Действительно, распространение этого явления настолько широко, что можно говорить о глубоком кризисе доверия. В настоящей работе это утверждение подтверждается результатами опроса более чем 2200 публикаций по телекоммуникационным сетям в последних трудах IEEE INFOCOM и таких журналах, как IEEE Transactions on Communications, IEEE/ACM Transactions on Networking и The Performance Evaluation Journal.

Для того чтобы математически исследовать свойства потери пакетов и задержки телекоммуникационной сети или компонента такой сети, модель должна быть сформулирована система, в которой такие явления выражаются. Это можно сделать, используя раздел математики, известный как теория массового обслуживания. В теории массового обслуживания изучаются модели систем, в которых “клиенты” (случайным образом) прибывают на “станцию обслуживания”, чтобы быть “обслуженными”; поскольку впереди них могут быть другие клиенты, им может потребоваться ждать в буфере или “очереди”. Модели массового обслуживания характеризуются вероятностным распределением времени между прибытиями, вероятностным распределением времени, необходимого для обслуживания клиента, размером буферного пространства (если оно конечное), политикой массового обслуживания (например, первый пришел первым обслужен) и т. д. Одним из оригинальных мотивирующих приложений для развития теории массового обслуживания была фактически телефонная сеть в начале 20-го века: время ожидания до тех пор, пока вызов не сможет быть обработан оператором, и вероятность блокировки вызова из-за отсутствия свободных линий были одними из первых результатов А. К. Эрланг [ВНЖ60].

Уже в начале развития телекоммуникационных сетей с коммутацией пакетов в 1960-х годах теория массового обслуживания была применена к моделям таких сетей [Кле64]. Модель массового обслуживания коммуникационной сети обычно включает один или несколько источников, которые отправляют пакеты (см. клиенты) в сеть (ср. станция технического обслуживания); затем эти пакеты передаются (ср. обслуживается) по сетевому каналу связи, если канал свободен, или временно хранится в буферной памяти (ср. очередь), если канал занят передачей другого пакета.

Учитывая такую модель массового обслуживания, вопрос заключается в том, как оценить интересующие показатели производительности, такие как вероятность переполнения буфера (или потери пакета).

В идеале это можно было бы сделать, используя только аналитические средства, такие как теория вероятностей и математическое исчисление, чтобы получить замкнутые выражения показателя эффективности в терминах параметров модели. Действительно, для простых моделей это часто возможно;] для многих примеров. Такие модели обычно содержат только одну очередь, и/или распределение межприватного времени и времени обслуживания имеют хорошие свойства (например, распределение без памяти), и / или показатель производительности прост (например, среднее время ожидания). В несколько более сложных моделях явное выражение замкнутой формы может быть

недоступно, но результаты все же могут быть найдены численной оценкой (например, рекурсией, матричным уравнением или обратным преобразованием Лапласа). Кроме того, в тех случаях, когда решение невозможно, могут быть использованы аппроксимации для упрощения вычислений; например, может быть рассчитано предельное поведение для большого размера буфера.

Имитационное моделирование в настоящее время является важнейшим методом научного познания окружающего мира. Оно дало возможность проводить лабораторные исследования сложных систем и предсказывать их поведение, без осуществления сложных натуральных экспериментов, требующих больших затрат различных ресурсов.

Учитывая нелинейно возрастающую сложность изучаемых систем, имитационное моделирование с каждым годом приобретает все большее значение в качестве научного метода исследований.

Таким образом, анализ современных систем имитационного моделирования представляет собой основу для выбора инструмента научного познания как природных, так и технических систем.

Наиболее крупными группами классификации видов имитационного моделирования являются:

- дискретно-событийное моделирование – организация моделей как набор конечных состояний и событий, оперирование которыми и представляет поведение модели;

- агентное моделирование – вид имитационного моделирования, в котором модель представляется большим количеством самостоятельных элементов, имеющих собственное состояние и поведение, в совокупности образующих имитируемую систему;

- системная динамика - вид моделирования, в котором модель реализуется посредством описания (часто представленного математически) взаимосвязей между компонентами рассматриваемой системы.

Области применения имитационного моделирования, как правило, предполагают узкую специализацию определенной научной или промышленной сферы деятельности. Специализированные системы предполагают высокую когнитивную нагрузку на исследователя вследствие необходимости оперирования большим количеством специальных сведений по рассматриваемой тематике моделирования.

Следует учитывать, что эти два основных классификатора не учитывают прочие факторы, определяющих распространенность средств имитационного моделирования. Среди них имеются следующие показатели:

- Стоимость системы имитационного моделирования является прямым фактором распространения системы имитационного моделирования. Разные политики распространения (например, условно-бесплатное программное обеспечение) также в значительной степени влияют на распространение пакета программного обеспечения системы имитационного моделирования, как в Российской Федерации, так и за ее пределами.

- Удобство использования системы имитационного моделирования – необъективный косвенный показатель, характеризующий наличие дополнительных компонентов, интеграцию с другими программными продуктами, удобство использования интерфейса «человек-машина» и т.д.

- Наличие и объем доступной русскоязычной документации по системе имитационного моделирования – прямой показатель, увеличивающий распространение системы, как в студенческой, так и в научной среде за счет наличия информационного сопровождения системы.

Моделирование системы связи требует генерации дискретизированных значений всех входных сигналов, обработки этих выборок через модели функциональных блоков в системе и оценки производительности системы по выходным выборкам в различных точках модели.

Механические генераторы случайных чисел впервые были использованы для моделирования азартных игр, а методы моделирования с использованием генераторов случайных чисел носят название города Монте-Карло, где находится всемирно известное казино. Хотя существует множество вариаций метода Монте-Карло, он в основном включает моделирование случайного эксперимента с использованием искусственных средств, то есть без полного повторения основного физического эксперимента.

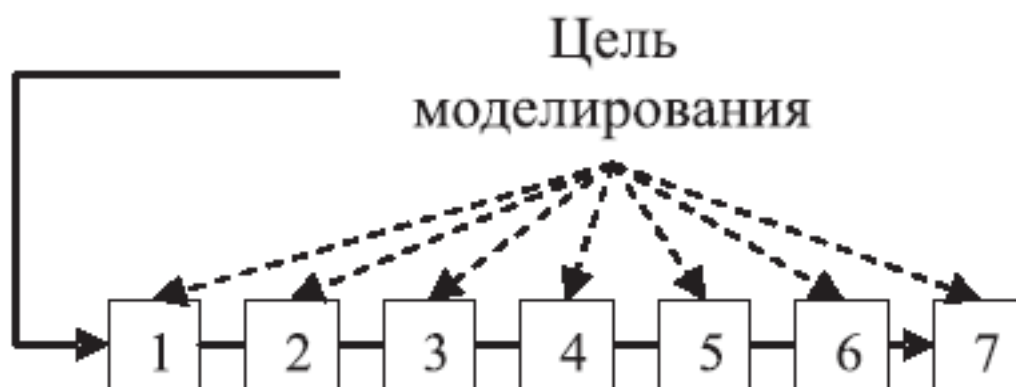


Рисунок 1.5 – Технологическая схема имитационного моделирования

- 1 - Реальная система;
- 2 - Построение логико-математической модели;
- 3 - Разработка моделирующего алгоритма;
- 4 - Построение имитационной (машинной) модели;
- 5 - Планирование и проведение имитационных экспериментов;
- 6 - Обработка и анализ результатов;
- 7 - Выводы о поведении реальной системы (принятие решений).

На рисунке 6 дана модель системы связи и описание входных сигналов $U(t)$, $V(t)$ и $W(t)$, которые считаются случайными процессами. Наша цель состоит в том, чтобы найти статистические свойства $Y(t)$ или ожидаемое значение некоторой функции $g(Y(t))$ от $Y(t)$. Если мы решим эту проблему,

эмулируя систему, включая эволюцию во времени всех осциллограмм в системе, то мы получим чистое моделирование по методу Монте-Карло. Это подразумевает генерацию выборочных значений всех входных процессов, позволяя моделям функциональных блоков в системе связи работать на них, и наблюдение выходных сигналов. В идеале моделирование методом Монте-Карло представляет собой однозначное соответствие реальной системе в рамках модельных допущений и приближений. Ожидаемое значение $E\{g(Y(t))\}$ оценивается на основе моделирования МК в соответствии с

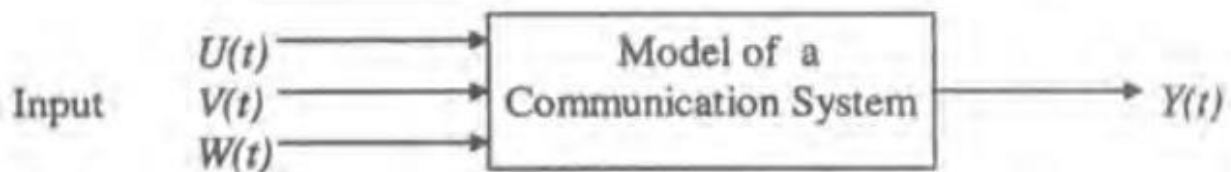


Рисунок 1.6 - Схема моделирования по методу Монте-Карло

Моделирование методом Монте-Карло для оценки частоты битовых ошибок в цифровой системе связи включает следующие этапы.

1. Генерируйте дискретизированные значения входной битовой последовательности $\{A(k)\}$, $k = 1, 2, \dots, N$, и шумовые выборки $\{N(j)\}$, $j = 1, 2, \dots, mN$ (частота дискретизации составляет m выборок на бит).
2. Обработайте эти образцы через модели функциональных блоков и сформируйте выходную последовательность $Y(k)$.
3. Оценка $E(g(Y(k)))$

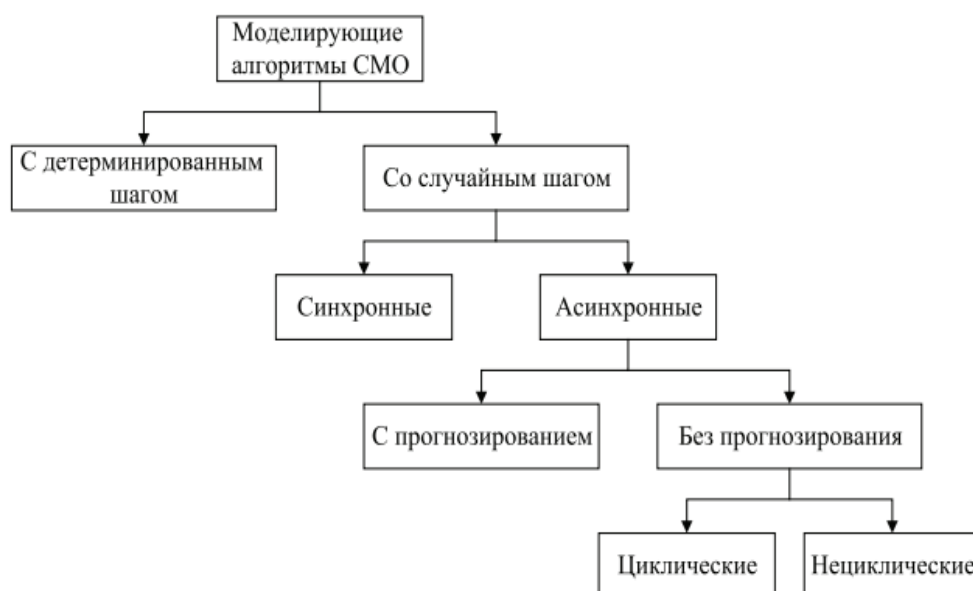


Рисунок 1.7 - Классификация моделирующих алгоритмов СМО

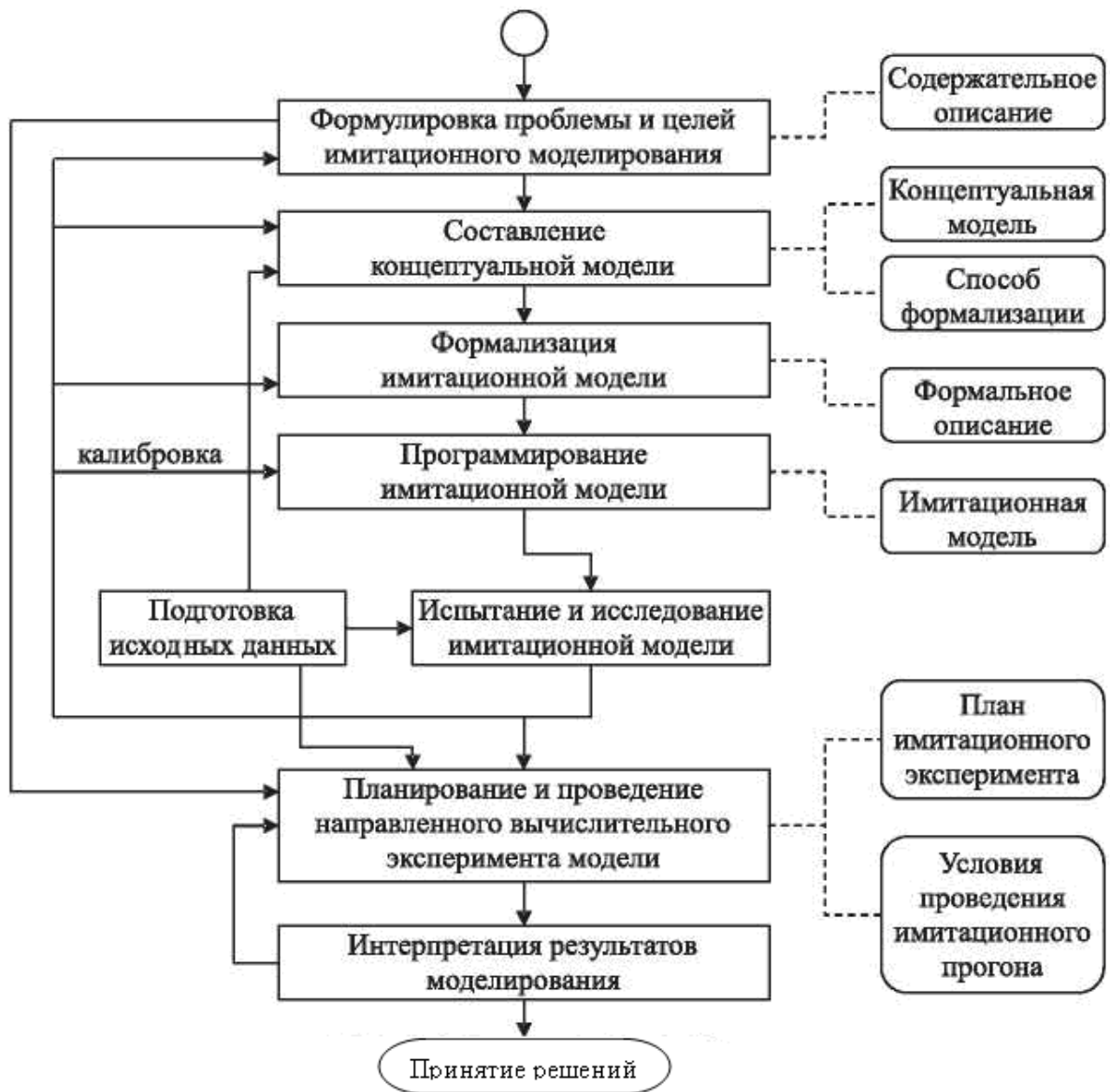


Рисунок 1.8 - Технологические этапы имитационного моделирования

Вариации моделирования Монте-Карло-Квазианалитическое моделирование Монте-Карло не все моделирования являются моделями Монте-Карло или чистыми моделями Монте-Карло. Монте-Карло подразумевает, что эмулируется по крайней мере один случайный процесс, а чистое Монте-Карло подразумевает, что эмулируются все входные процессы. Во многих приложениях моделирования нет необходимости моделировать все случайные процессы, присутствующие в системе. В примере, показанном на рисунке, частота ошибок системы будет зависеть от шума и кумулятивных искажений, вносимых нелинейностью и фильтрами. Поскольку шум является гауссовым и аддитивным на входе, его чистый эффект на выходе фильтра и, следовательно, на метрике решения также будет аддитивным и гауссовым. При отсутствии искажений эффект аддитивного Гауссова шума может быть обработан аналитически без моделирования. Даже с искажением, эффекты

AWGN все еще могут быть охарактеризованы аналитически для данного значения искажения. Однако распределение значений искажений, вносимых нелинейностью и фильтрами, может быть трудно охарактеризовать аналитически, но легко смоделировать. В таком случае нам нужно только имитировать кумулятивное влияние всех функциональных блоков на входную двоичную форму сигнала, и нет необходимости явно имитировать шумовую форму сигнала. Моделирование МК, в котором только некоторые (но не все) входные процессы в систему моделируются явно, в то время как эффекты других процессов обрабатываются с использованием аналитических методов, называется частичным МК или квазианалитическим (КА) моделированием. Главное преимущество моделирования КА заключается в том, что для получения оценок с той же точностью потребуется моделирование меньшего количества выборок, чем для чистого моделирования МК.

2 Моделирование систем телекоммуникаций как систем массового обслуживания

2.1 Основные понятия систем массового обслуживания

Система массового обслуживания является математическим (абстрактным) объектом, который состоит из нижеперечисленных элементов (рисунок 1):

- входной (входящий) поток заявок (требований) на обслуживание;
- приборы (каналы) обслуживания;
- очередь заявок, ожидающих обслуживания;
- выходной (выходящий) поток обслуженных заявок;
- поток заявок на дообслуживание после прерывания обслуживания;
- поток необслуженных заявок.

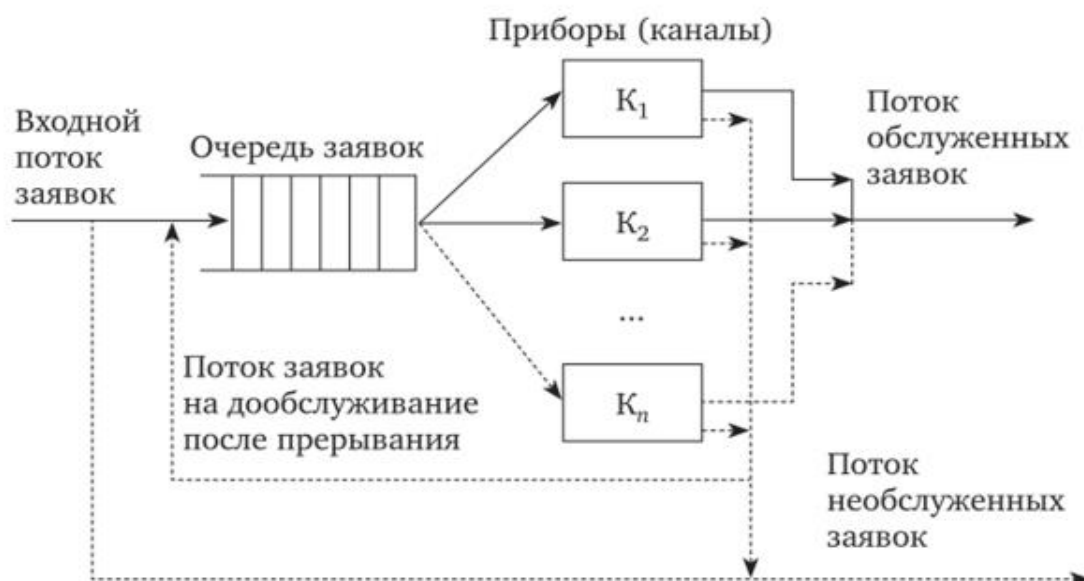


Рисунок 2.1 – Система массового обслуживания

Существует множество систем массового обслуживания, которые отличаются своей структурной и функциональной организацией. Однако разработка аналитических методов расчета показателей эффективности деятельности финансовых директоров во многих случаях сопряжена с рядом ограничений и допущений, которые сужают круг изучаемых финансовых директоров. Поэтому не существует универсальной аналитической модели для произвольной сложной структуры СМО.

Аналитическая модель СМО является совокупностью уравнений или формул, которые позволяют определить вероятности состояний системы в процессе ее функционирования и показатели эффективности на основе известных параметров входящего потока и каналов обслуживания, буферизации и сервисных дисциплин.

Аналитическое моделирование СМО значительно облегчается, если процессы, происходящие в СМО, являются Марковскими процессами (потоки приложений являются самыми простыми).

В системах обмена мгновенными сообщениями, следующие ограничения и допущения принимаются при реализации СМО:

- запрос, полученный в системе, немедленно отправляется на обслуживание, если в очереди нет запросов и устройство свободно;

- устройство может иметь только один запрос в обслуживании в любое время;

- после окончания обслуживания любого приложения в устройстве сразу же выбирается следующее приложение из очереди на обслуживание, т. е. устройство не простаивает, если в очереди есть хотя бы одно приложение;

- прием заявок в ЦФО и продолжительность их обслуживания не зависят ни от количества заявок, уже находящихся в системе, ни от каких-либо других факторов;

- продолжительность обслуживания заявок не зависит от интенсивности приема заявок в систему.

Давайте рассмотрим некоторые элементы СМО более детально.

Входной (входящий) поток запросов. Поток событий - это последовательность однородных событий, которые следуют друг за другом и происходят в определенные, вообще говоря, случайные моменты времени. Если событие состоит из появления запросов, то мы имеем поток запросов.

Важным методом аппроксимации является теория больших отклонений, которая рассматривает поведение системы, когда она отклоняется далеко от своего среднего поведения.

Обзор теории больших отклонений и ее приложений к телекоммуникационным системам представлен в работе .

Если аналитический или численный расчет невозможен и соответствующие аппроксимации отсутствуют, моделирование часто может быть полезным. Стохастическое моделирование - это использование компьютерной программы для выборки (псевдослучайных) значений случайных величин в модели, таким образом, создавая "путь выборки":

последовательность событий, которые могут произойти в модели. Повторяя это, можно получить множество различных и независимых выборочных траекторий. Затем можно получить оценку величины интереса, взяв среднее ее значение по всем моделируемым траекториям выборки. Очевидно, что из-за случайности это только приближение истинного значения величины. Поэтому вместе с оценкой вычисляется также дисперсия оценщика: мера того, насколько точной может быть оценка (для более точных определений см., например, чем больше траекторий выборки моделируется, тем меньше становится дисперсия (если только дисперсия не существует (бесконечна)). Большим преимуществом моделирования является его общность: в отличие от большинства аналитических и численных методов, оно не накладывает ограничений на используемые распределения вероятностей; кроме того, при наличии достаточной вычислительной мощности можно моделировать очень сложные модели и получать оценки с любой требуемой точностью.

Однако моделирование также имеет некоторые важные проблемы. Во-первых, это может быть очень трудоемко: чтобы точно оценить вероятность интересующего события, необходимо собрать множество (почти) независимых наблюдений за ним в ходе моделирования. Кроме того, один такой прогон моделирования в принципе дает оценку только для одного набора значений параметров модели (размер буфера, скорость поступления и т. д.). На практике часто требуются оценки при многих значениях параметров, например для выбора размера буфера таким образом, чтобы вероятность переполнения была достаточно низкой; в таких случаях моделирование необходимо повторять для многих значений параметра. (Однако следует отметить, что при использовании современных методик [RM98] одного прогона моделирования может быть достаточно для получения оценок диапазона значений некоторых параметров модели.)

Один класс задач, для которых (стандартное) моделирование является довольно непригодным, - это задачи, связанные с оценкой вероятностей редких событий, т. е. событий, которые имеют очень низкую вероятность возникновения (например, 10⁻⁶ или менее). Такие события представляют большой интерес в моделях массового обслуживания телекоммуникационных систем, поскольку они, как правило, рассчитаны на очень низкую вероятность потери пакетов, гарантирующую хорошее качество обслуживания. Эти низкие вероятности означают, что система может быть смоделирована в течение длительного времени без того, чтобы событие произошло хотя бы один раз. Как отмечалось выше, событие должно наблюдаться много раз во время выполнения моделирования, чтобы оценка его вероятности была точной; поэтому оценка редких событий требует непрактично длительных запусков моделирования, если не используются специальные методы.

Аналитическое исследование СМО обычно используется в предположениях о простейшем потоке запросов, что обуславливается рядом особенностей:

1. Суммирование (интегрирование) потоков. Простейший поток в теории CFR аналогичен нормальному закону распределения в теории вероятностей: предельный переход для потока, представляющего собой сумму потоков с произвольными характеристиками с бесконечным увеличением числа слагаемых и уменьшением их интенсивности, приводит к простейшему потоку.

2. Вероятностное разбавление потока. Вероятностное разбавление простейшего потока заявок, при котором любое приложение случайным образом исключается из потока с некоторой вероятностью P , независимо от того, исключаются ли другие приложения или нет, приводит к образованию простейшего потока с интенсивностью $X^* = pX$, где X -интенсивность исходного потока. Поток исключенных запросов с интенсивностью $X^{**} = (1-p)X$ также является простейшим потоком.

3. Эффективность. Если сервисные каналы (устройства) рассчитаны на простейший поток запросов с интенсивностью x , то обслуживание других типов потоков (с той же интенсивностью) будет обеспечиваться с не меньшей эффективностью.

4. Простота. Предположение о простейшем потоке приложений позволяет получить явные зависимости показателей CFR от параметров для многих математических моделей. Наибольшее количество аналитических результатов было получено для простейшего потока запросов.

2.2 Имитационное моделирование системы массового обслуживания М/М/5/0 в системе GPSS Studio

Система массового обслуживания используется при исследовании реальных вероятностных систем. М/М/м/Е является общим видом имитационной модели ОСМО согласно классификации Кендалла. Параметр m является количеством обслуживающих каналов, а параметр E – количественным ограничением мест в очереди. Модель М/М/5/0 является частным случаем модели М/М/м/Е при конкретных значениях параметров $m=5$ и $E=0$. В качестве программной среды для создания имитационной модели выбрана система структурного и имитационного моделирования (ССИМ) GPSS Studio [12]. На рисунке 2 мы увидим структурную модель ОСМО М/М/5/0 в GPSS Studio.

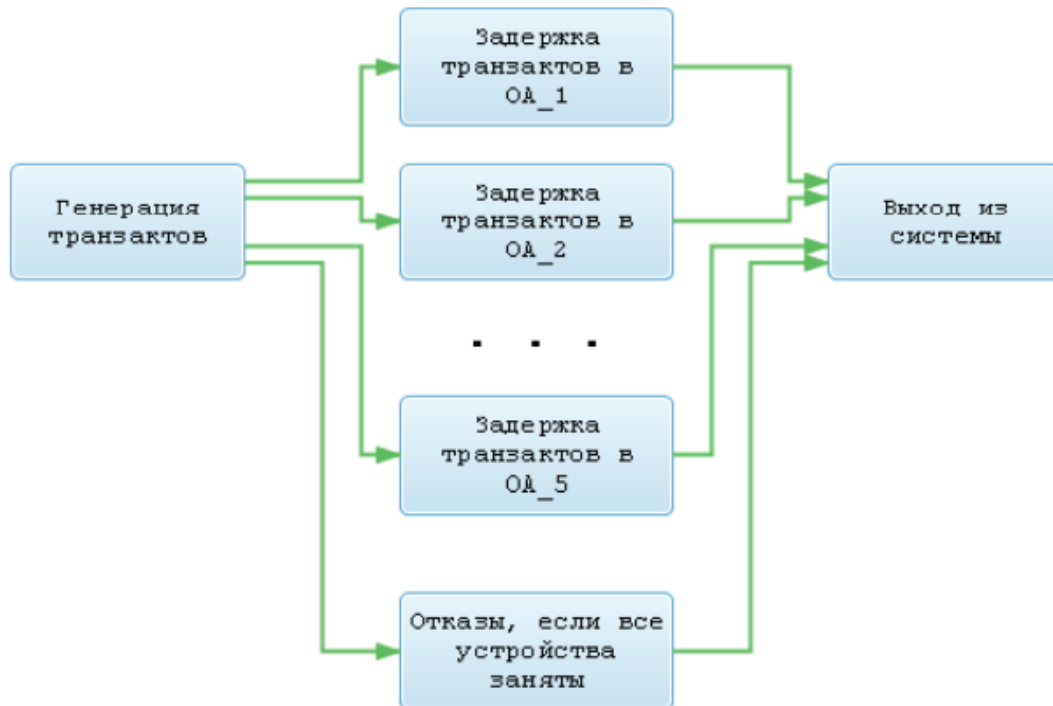


Рисунок 2.2 – Структурная модель ОСМО М/М/5/0 в ССИМ GPSS Studio

На рисунке 3 приведен стандартный отчет по ИМ ОСМО М/М/5/0 из системы GPSS Studio.

	Имя / номер	Кол-во тран. ожидающих выполнения	Значение сохраняемой величины
1. Среднее время между поступлением заявок.	TIMEOFARRIVAL	0	10.000
2. Среднее время обслуживания одной заявки.	TIMEOFSERVICE	0	30.000
3. Общее количество обслуживаемых заявок.	AMOUNTOFTRANSACTS	0	10000.000
4. Количество обслуживающих аппаратов (ОА).	DEVICES	0	5.000
5. Количество заявок, которым было отказано в обслуживании.	FAIL	0	1136.000
6. Вероятность отказа.	DEVIATIONFAIL	0	0.114
7. Среднее время задержки заявки в системе.	AVERAGETIMEOFSTAY	0	30.825
8. Среднее время обслуживания заявки в ОА.	AVERAGETIMEOFSERVICE	0	30.825
9. Среднее время ожидания заявки в очереди.	AVERAGEOFWAIT	0	0
10. Среднее количество заявок в ОА.	TRANSACTINOA	0	2.729
11. Среднее количество заявок в очереди.	TRANSACTINQUEUE	0	0
12. Среднее количество заявок в системе.	TRANSACTINSYSTEM	0	582.591
13. Стандартное отклонение времени пребывания заявки в системе.	DEVIATIONSTAY	0	30.999
14. Стандартное отклонения времени задержки заявки в ОА.	DEVIATIONSERVICE	0	30.999
15. Стандартное отклонения времени ожидания заявки в очереди.	DEVIATIONWAIT	0	0

Рисунок 2.3 – Основные результаты ИМ ОСМО М/М/5/0 в ССИМ GPSS

При сравнении результатов ИМ и АМ использовали параметрические и

непараметрические методы. В качестве параметрических были использованы следующие два метода [13]:

1. Оценка различия средних значений параметров СМО, полученных ИМ и АМ для i -го параметра СМО. Оценка производится по критерию Стьюдента и вычисляется по формуле:

$$t_i = \sqrt{n} \frac{|m_{1i}^* - m_{1i}|}{\sigma_i}, \quad (2.1)$$

где n – количество реализаций;

m_{1i}^* - оценка математического ожидания i -го параметра СМО по результатам ИМ;

m_{1i} - математическое ожидание i -го параметра СМО по результатам АМ;

σ_i - оценка стандартного отклонения i -го параметра СМО по результатам АМ.

Если вычисленное значение t_i не превышает критическое значение критерия Стьюдента $t_{krit}=1.960$, тогда можно считать, что полученные при ИМ результаты достоверны.

2. Проверка гипотезы о равенстве дисперсий результативных показателей СМО по критерию Пирсона производится по формуле:

$$x_{ij}^2 = (n - 1) \frac{\sigma_{ij}^{*2}}{\sigma_i^2} \text{ при } \sigma_{ij}^{*2} \geq \sigma_i^2, \quad X_{ij}^2 = (n - 1) \frac{\sigma_i^2}{\sigma_{ij}^{*2}} \text{ при } \sigma_{ij}^{*2} < \sigma_i^2, \quad (2.2)$$

где σ_{ij}^{*2} – оценка дисперсии i -го параметра для j -й ССИМ;

σ_i^2 – оценка дисперсии i -го параметра для системы аналитического моделирования.

Если вычисленное значение не превышает критическое значение критерия $X_{ikrit}^2=10905$, то результаты ИМ признаются достоверными.

Результаты сравнения по параметрическим методам приведены в таблице 1.

Таблица 1 - Сравнение СМОМ/М/5/0 параметрическими методами.

Сравнение АМ и ИМ по критерию Стьюдента для М/М/5/0				
№	Наименование	АМ	ИМ	Разница по Стьуденту
1	Среднее время пребывания заявки в системе	30.0	30.143	1.475
2	Среднее время задержки заявки в ОА	30.0	30.143	1.475
3	Среднее количество заявок в ОА	2.67	2.695	1.812
4	Среднее количество заявок в системе	2.67	2.695	1.812
Итого:		По 4 тестам: +4 - 0		
Сравнение АМ и ИМ по критерию Пирсона для М/М/5/0				
№	Наименование	АМ	ИМ	Разница по Стьуденту
1	Среднее время пребывания заявки в системе	1.90	1.884	10085.94 1
2	Среднее время задержки заявки в ОА	1.90	1.884	10085.94 1
3	Среднее количество заявок в ОА	900	945.62	10505.86 2
4	Среднее количество заявок в системе	900	945.62	10505.86 2
Итого:		По 4 тестам: +4 - 0		

В результатах, приведённых в таблице 1, мы видим, что вычисленные значения критериев Стьюдента и Пирсона не превышают критическое значение ни в одном из 4 случаев. В связи с этим результаты считаются удовлетворительными.

Непараметрические методы сравнения не требуют, чтобы были заданы параметры в явном виде, и используют для оценки достоверности интегрированные показатели. В данной работе использован непараметрический показатель коэффициент линейной корреляции.

Коэффициент линейной корреляции используется в случае нормальности распределений случайных величин x и y и наличия зависимости между ними близкой к линейной. Он вычисляется по n экспериментальным данным по следующей формуле:

$$r_{xy} = \frac{m_{1xy}^* - m_{1x}^* \cdot m_{1y}^*}{\sigma_x^* \cdot \sigma_y^*} \quad (2.3)$$

$$m_{1x}^* = \frac{1}{n} \sum_{i=1}^n x_i = \frac{450}{15} = 30,$$

$$m_{1y}^* = \frac{1}{n} \sum_{i=1}^n y_i = \frac{461.83}{15} = 30.789,$$

$$m_{1xy}^* = \frac{1}{n} \sum_{i=1}^n x_i \cdot y_i = \frac{16201.44}{15} = 1080.096.$$

Оценки вторых начальных моментов используются при вычислении средних квадратических отклонений. Для этого требуются следующие формулы:

$$m_{2x}^* = \frac{1}{n} \sum_{i=1}^n x_i^2 = \frac{15750}{15} = 1050,$$

$$m_{2y}^* = \frac{1}{n} \sum_{i=1}^n y_i^2 = \frac{16690.13}{15} = 1112.6753,$$

$$\sigma_x^* = \sqrt{m_{2x}^* - m_{1x}^{*2}} = 12.25,$$

$$\sigma_y^* = \sqrt{m_{2y}^* - m_{1y}^{*2}} = 12.83.$$

Коэффициент линейной корреляции вычисляется по формуле:

$$r_{xy} = \frac{m_{1xy}^* - m_{1x}^* \cdot m_{1y}^*}{\sigma_x^* \cdot \sigma_y^*} = \frac{156.426}{157.185} = 0.99517$$

По результатам проделанной работы можно сделать выводы, что исследуемую систему GPSS Studio можно использовать для проведения имитационного моделирования различных СМО, исходя из высоких результатов по всем методам. Провели сравнительный анализ результатов СМО М/М/5/0 по параметрическим методам: критериям Стьюдента и Пирсона. Удовлетворительные результаты сравнения говорят о том, что использование данных параметрических методов рационально для проведения оценки правильности имитационных моделей.

2.3 Имитационное моделирование системы массового обслуживания для анализа производительности IoT-приложений

Устройства интернета вещей могут быть мобильными, маломощными, недорогими, и в зависимости от условий окружающей среды результирующие приложения имеют различные Интернет вещей (IoT) обещает интеграцию физического мира в компьютерные системы. Устройства интернета вещей, обладающие сенсорными возможностями, развертываются в различных областях применения, таких как интеллектуальные здания, общественные пространства, интеллектуальный транспорт и многие другие. Используя их информацию, мы открываем новые возможности для повышения безопасности и качества жизни людей. Такое приложение генерирует критическую информацию и, как ожидается, будет функционировать правильно, своевременно и надежно. Таким образом, определение общей методологии моделирования производительности и анализа приложений Интернета вещей имеет первостепенное значение.

характеристики. В частности, можно выделить несколько ограничений качества обслуживания (QoS):

- прерывистое подключение мобильных устройств;
- ограниченная память недорогих устройств;
- ограниченная доступность данных в сценариях реагирования на чрезвычайные ситуации и т.д.

Кроме того, IoT-приложения построены поверх надежных / ненадежных протоколов и интерфейсов прикладного программирования (API), которые вводят дополнительные QoS ограничения [12]. Таким образом, эти ограничения создают важные проблемы для поддержки анализа производительности Интернета вещей.

Существующие усилия [11], касающиеся проектирования и оценки мобильных систем в условиях конкретных ограничений (например, прерывистая доступность, ограниченные ресурсы и т.д.) и для конкретных случаев использования, опираются на область теории массового обслуживания. Ключевые протоколы Интернета вещей были оценены с учетом таких показателей производительности, как время отклика и успешность доставки [9], [10]. Однако такие усилия носят протокольный характер, и поэтому исследовательское сообщество проанализировало эффективность хорошо известных парадигм взаимодействий, которые могут абстрагироваться от различных протоколов интернета вещей. Парадигма публикации / подписки была проанализирована с использованием формальных моделей [13] и оценена с использованием таких моделей, как сети Петри для массового обслуживания [14] или сети массового обслуживания.

Сети массового обслуживания широко применяются для представления и анализа коммуникационных и компьютерных систем. Сеть массового обслуживания - это сеть подключенных сервисных центров, которая предоставляет аналитические или имитационные решения для измерения

производительности (например, времени отклика). Объединение отдельных сервисных центров для формирования сети массового обслуживания позволяет нам моделировать и представлять взаимодействие Интернета вещей. Для этого мы определили методологию, в которой представлены узлы протокола промежуточного программного обеспечения (клиенты, серверы, брокеры и т. д.) и в качестве очередей и обмениваемых сообщений служили задания. Для моделирования параметров QoS, таких как прерывистое подключение мобильных устройств, мы исследовали отдельный центр массового обслуживания с помощью так называемой модели включения/выключения массового обслуживания. Затем мы продемонстрировали применимость нашего подхода, моделируя производительность протоколов публикации/подписки и потоковой передачи данных [13]. В работе мы применили наш подход к анализу производительности IoT-устройств с гетерогенными настройками QoS. Основные вклады этой работы заключаются в следующем:

1) Описание универсальной модели включения / выключения, расширенной вероятностями соединения сообщений и потерями сообщений.

2) введение дополнительных функций для моделей массового обслуживания, представляющих устройства с ограниченными ресурсами и доступность сообщений.

3) сравнение различных моделей массового обслуживания с точки зрения времени отклика и успешности сообщений с помощью имитационных экспериментов.

Рассмотрим отдельные модели массового обслуживания, которые используются в рамках имитационных сетей массового обслуживания.

А. Модель М/М / 1

Эта очередь моделирует непрерывное обслуживание (передачу, прием или обработку) сообщений в рамках сквозного взаимодействия Интернета вещей. Он соответствует наиболее распространенной очереди М/М/1 с Пуассоновским входным потоком и экспоненциальным временем обслуживания.

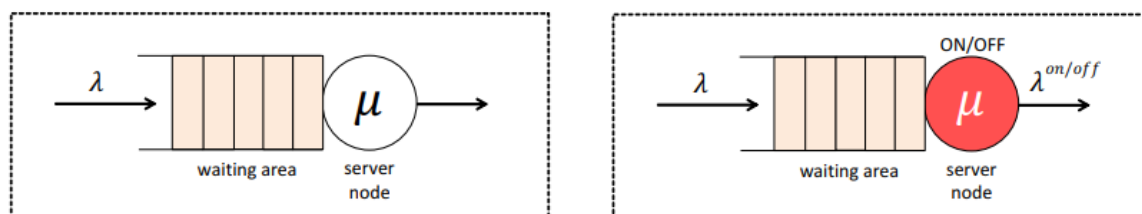


Рисунок 2.4 – Модели М/М/1 и ON/OFF

Очередь М/М/1 определяется следующей формулой:

$$q_{m/m/1} = (\lambda, \mu) \quad (2.4)$$

где λ - скорость ввода сообщений в очередь;

μ - скорость обслуживания для обработки сообщений.

Пусть D -потребность в обслуживании для задержки обработки каждого сообщения (т. е. $D = 1/\mu$). Основываясь на стандартных решениях для очереди $M/M/1$ [15], время, в течение которого сообщение остается в системе (соответствующее времени ожидания в очереди + времени обслуживания; также называют его средним временем отклика), задается следующим образом:

$$\Delta_{m/m/1} = \frac{D}{1-\lambda D} \quad (2.5)$$

В.Вкл-Выкл модель

Для работы с подключениями и отключениями мобильного однорангового узла введем прерывистую очередь (ВКЛ/ВЫКЛ), которая изображена на рисунке 1, сообщения поступают в соответствии с пуассоновским процессом со скоростью $\lambda > 0$ и помещаются в очередь, ожидающую «обслуживания». Сообщения подаются со скоростью $\mu > 0$, которая распределяется экспоненциально.

Мы предполагаем, что сервер подвергается процедуре включения-выключения[16]. Тем не менее, он остается в включенном состоянии в течение экспоненциально распределенного времени с параметром θ_{ON} , в течение которого он обслуживает сообщения (если таковые имеются). Пусть T_{ON} - это время, когда сервер включен ON - т. е., $T_{ON} = 1/\theta_{ON}$. По истечении этого времени сервер переходит в выключенное состояние, в течение которого он перестает работать (перестает обслуживать соответствующие сообщения) в течение экспоненциально распределенного периода времени со скоростью θ_{OFF} . Пусть T_{OFF} - это время, когда сервер выключен - т. е. $T_{OFF} = 1/\theta_{OFF}$. Соответственно, очередь ВКЛ/ВЫКЛ $q_{ON/OFF}$ определяется:

$$q_{ON/OFF} = (\lambda, \lambda^{ON/OFF}, \mu, T_{ON}, T_{OFF}) \quad (2.6)$$

где λ - скорость ввода сообщений в очередь,

$\lambda^{ON/OFF}$ - это скорость вывода сообщений,

μ - скорость обслуживания для обработки сообщений (если таковые имеются) во время T_{ON} .

Процесс вывода $\lambda^{ON/OFF}$ является прерывистым, поскольку никакие сообщения не выходят из очереди во время интервалов T_{OFF} . Не теряя общности, мы делаем следующее предположение: если T_{ON} истекает и есть сообщение, которое в данный момент подается, сервер прерывает его обработку и продолжит ее в следующем периоде T_{ON} .

Пусть $\Delta^{ON/OFF}$ - среднее время отклика (время, в течение которого сообщение остается в системе) для очереди включения/выключения:

$$\lambda^{ON/OFF} = \frac{E(n)_{ON/OFF}}{\lambda} \quad (2.7)$$

где $E(n)_{ON/OFF}$ - среднее количество сообщений в системе (сервер + очередь).

С. Вероятностная модель вкл/выкл

Чтобы уменьшить задержки массового обслуживания в модели включения/выключения, мы вводим вероятностную модель включения/выключения, которая изображена на рисунке 5. В такой модели сообщения поступают в систему согласно Пуассоновскому процессу со скоростью λ .

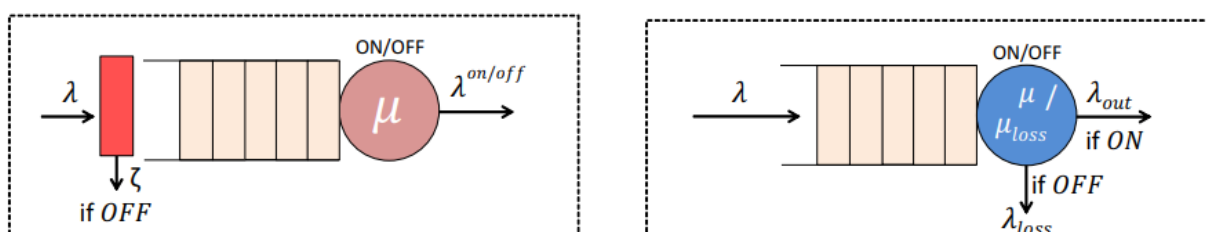


Рисунок 2.5 – Вероятностные и убыточные очереди ВКЛ /ВЫКЛ

Если сервер подключен (включен), то сообщения помещаются в очередь, ожидающую их обслуживания. В противном случае, если сервер находится в автономном режиме (выключен), сообщения решают присоединиться к очереди с вероятностью ζ или покинуть систему с вероятностью $1-\zeta$. Вероятностная очередь включения/выключения $q_{ON/OFF}^{prob}$ определяется:

$$q_{ON/OFF}^{prob} = (\lambda, \lambda^{ON/OFF}, \mu, \zeta, T_{ON}, T_{OFF}) \quad (2.8)$$

где λ -скорость ввода сообщений;

ζ -вероятность вступления в очередь;

$\lambda^{ON/OFF}$ - это скорость вывода сообщений;

μ - скорость обслуживания для передачи сообщений (если таковые имеются) во время T_{ON} .

Стоит отметить, что если $\zeta = 1$, то все сообщения присоединяются к системе в нерабочее время. Это эквивалентно модели ВКЛ/ВЫКЛ, следовательно, вероятностная модель ВКЛ /ВЫКЛ обобщает модель ВКЛ /ВЫКЛ.

$\Delta^{ON/OFF}$ - это среднее время отклика (время, когда сообщение остается в системе) для очереди $q_{ON/OFF}^{prob}$.

Используя вероятностную модель ВКЛ/ВЫКЛ, разработчики систем смогут соответствующим образом настроить и улучшить задержки компонентов, которые могут отключаться в течение среднего периода T_{OFF} и передавать сообщения во время T_{ON} по ограниченной полосе пропускания. Снижение скорости поступления сообщений во время T_{OFF} , уменьшает задержки в очереди и предотвращает затопление сети во время T_{ON} . Таким компонентом может быть ограниченный ресурсами (с точки зрения энергии и памяти) IoT-датчик, который передает сообщения по ограниченной полосе пропускания и может периодически находиться в спящем режиме (вкл / выкл). Это вероятностное поведение (т. е. вероятность передачи сообщений во время T_{OFF}) должно быть определено из соответствующего прикладного уровня устройства IoT.

Д. Модель потерь ВКЛ/ВЫКЛ

Модели ВКЛ/ВЫКЛ, определенные в вышеприведенных подразделах, не предполагают выпадения сообщений (или потери) из системы – любое сообщение либо подается, либо буферизуется, а затем подается.

Вероятностная модель ВКЛ/ВЫКЛ предполагает потерю сообщений, однако такие сообщения не попадают в очередь, так как они решают присоединиться к очереди с вероятностью ζ . Для моделирования потерь сообщений введем очередь потерь ВКЛ/ВЫКЛ. Аналогично вышеприведенным очередям включения/выключения, скорость поступления (λ) сообщений в систему является пуассоновской [19]. Если сервер включен, то сообщения помещаются в очередь, ожидающую «обслуживания» со скоростью обслуживания μ . В противном случае (т. е. сервер выключен) все сообщения, которые присоединяются к очереди ($\zeta = 1$), «обслуживаются» со скоростью обслуживания μ_{loss} , а затем они выходят из системы (т. е. мы потеряли сообщения). Мы предполагаем, что потерянные сообщения «обслуживаются» со скоростью обслуживания $\mu_{loss} > \mu$.

Последнее предположение вводится по следующей причине: мы определили очередь потерь ВКЛ/ВЫКЛ для моделирования производительности мобильного устройства IoT, использующего протокол промежуточного программного обеспечения. Такой протокол вводит потери сообщений, так как он строится поверх UDP и не настраивает никакой логической сеанс между отправителем и получателем IoT-устройств. Сообщения отправляются без каких-либо гарантий на другую конечную точку. Как уже было определено, скорость обслуживания μ может быть параметризована на основе задержки сетевой передачи данных. соответствующее сквозное взаимодействие (отправитель-получатель). С другой стороны, чтобы параметризовать μ_{loss} , мы должны определить, в какой момент сетевые сообщения теряются. Например, если предположить, что очередь потерь ВКЛ/ВЫКЛ моделирует беспроводной приемник и его прерывистое подключение, то μ_{loss} можно параметризовать через соответствующую сетевую задержку передачи. Эта задержка может быть

определена на основе взаимодействия между отправителем и точкой доступа, которую подключает и отключает получатель.

Очередь потерь ВКЛ/ВЫКЛ $q_{ON/OFF}^{loss}$ определяются [17]:

$$q_{ON/OFF}^{loss} = (\lambda, \lambda_{out}, \lambda_{loss}, \mu, \mu_{loss}, T_{ON}, T_{OFF}) \quad (2.9)$$

где λ -скорость ввода сообщений;

λ_{out} и μ - скорость вывода и обработки сообщений в периоды включения;

λ_{loss} и μ_{loss} - скорость потери и обработки сообщений в периоды выключения. Скорости $\lambda_{out}/\lambda_{loss}$ являются экспоненциальными, так как поступившие сообщения разделяются на основе T_{ON}/T_{OFF} интервалов.

$\lambda_{ON/OFF}^{loss}$ - это среднее время отклика для очереди $q_{ON/OFF}^{loss}$.

До сих пор мы определяем модели массового обслуживания, имеющие буферы с бесконечной емкостью и поступающие сообщения с бесконечным временем жизни (т. е. период времени до жизни, который определяет доступность сообщения внутри сети массового обслуживания). Это, безусловно, влияет на время отклика, но также и на скорость успешного обслуживания сообщений по сравнению с общим количеством поступающих сообщений. Однако моделирование взаимодействий Интернета вещей с такими характеристиками может оказаться нереалистичным. Например, при длительном периоде отключения (например, 30 минут) датчика IoT полученные данные/сообщения могут превысить буферную емкость датчика и/или некоторые из самых старых данных могут устареть для принимающего приложения/пользователя. Мы вводим соответствующие функции, учитывающие вышеперечисленные ограничения. Эти функции могут быть применены к предыдущим моделям массового обслуживания.

В первом случае, это сети массового обслуживания с пожизненными сообщениями. Как уже отмечалось, сеть массового обслуживания - это сеть подключенных очередей, которая может использоваться для моделирования производительности системы. Например, в контексте нашей работы мы моделируем взаимодействие Интернета вещей путем создания сетей массового обслуживания, которые состоят из моделей массового обслуживания, представленных в предыдущих подразделах. Сообщения поступают со скоростью λ для того, чтобы быть обработанными в первой очереди сети массового обслуживания.

Поступающее сообщение несет в себе период жизни, приписываемый ему при его создании, который представляет собой действительность сообщения внутри сети массового обслуживания. Таким образом, сообщение может войти в сеть массового обслуживания, и как только его срок службы истекает, сообщение покидает сеть и считается истекшим.

Чтобы считать сообщение просроченным, мы учитываем время, которое сообщение проводит как в очереди, так и на сервере в каждой очереди. Если

предположить, что сеть массового обслуживания состоит из одной очереди М/М/1, сообщение возобновляется, если его обслуживание не начинается с определенного периода времени жизни (который также включает ожидаемый спрос на обслуживание). Основываясь на литературе по теории массового обслуживания, такая модель изучается как очередь М/М/1 с отказывающимися или нетерпеливыми клиентами [14]. В этой работе мы предлагаем моделирование вышеупомянутой модели М/М/1 через наш симулятор. Кроме того, мы обогащаем модели ВКЛ/ВЫКЛ, чтобы поддержать отказывающихся или нетерпеливых клиентов. Таким образом, разработчики систем могут создавать сети массового обслуживания, которые могут включать различные модели массового обслуживания, обогащенные периодами жизни.

Во втором случае имеем очереди с конечной емкостью: это хорошо известная функция модели массового обслуживания, где к очереди применяется определенный размер буфера, обеспечивающий наличие в системе максимально K сообщений (очередь + сервер). Это предотвращает хранение слишком большого количества сообщений в течение слишком долгого времени на устройствах с ограниченной аппаратной емкостью (память, жесткий диск). В частности, как показано на рисунке 6, сообщения поступают в очередь с λ_{in} . Перед тем как сообщение попадет в очередь проверяется следующее условие: новый размер очереди + сообщение в сервисе $> K$. Если условие истинно, сообщение отбрасывается. В противном случае сообщение попадает в очередь для обработки.

Согласно литературным данным, очередь М/М/1 с конечной емкостью обозначается как М/М/1/К [15]. В наших моделях мы представляем М/М/1 и очереди ВКЛ/ВЫКЛ, представленные в предыдущих подразделах, с конечной емкостью, добавляя размер системы (K) к соответствующему определению.

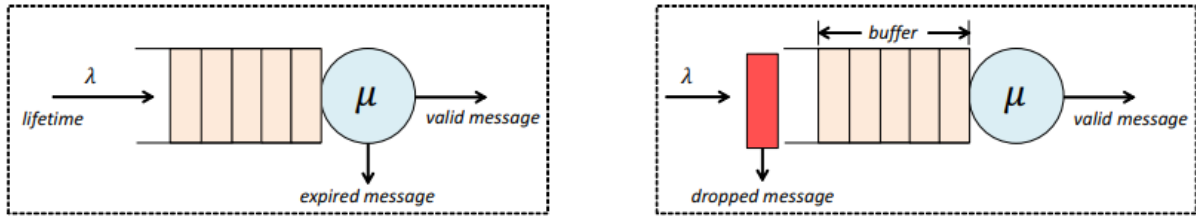


Рисунок 2.6 – Очереди с истечением срока действия событие и ограничение по мощности

Моделирование производительности приложений Интернета вещей-это утомительная задача. Обмен сообщениями в IoT может привести к высоким задержкам сообщений или потерям сообщений из-за нескольких параметров QoS, таких как доступность сообщений, прерывистое подключение, ограниченные ресурсы устройств и т.д. Используя разработанный имитатор, мы оцениваем компромисс между временем отклика и показателями успешности сообщений для различных моделей массового обслуживания. Опираясь на модели, разработчики систем могут создавать сети массового обслуживания, представляющие взаимодействие Интернета вещей, и анализировать их производительность или даже настраивать ее на основе наших экспериментальных результатов[20].

2.4 Дисперсионный анализ в GPSS

Дисперсионный анализ - это инструмент, впервые разработанный Рональдом Фишером, который способен извлечь большую часть информации, доступной в наборе измерений. В ней мы количественно оцениваем отклонение наблюдений от общего среднего, а затем разбиваем его на части, каждая из которых имеет свою отдельную причину. Если какой-либо экспериментальный фактор не может быть найден, чтобы вызвать изменчивость в измерении, мы говорим, что он не оказывает на него существенного влияния. С другой стороны, если какой-то фактор действительно вызывает изменчивость, мы сравниваем его величину с оценкой внутренней изменчивости наблюдения-стандартной ошибкой. Мы делаем это, чтобы исключить видимые эффекты, которые являются не более чем случайными флуктуациями. Наш стандарт сравнения заключается в том, что отклонение от любого источника должно быть намного больше стандартной ошибки, чтобы считаться значительным эффектом. Для этой цели используется тест F, названный в честь Фишера. Мы используем тест F в качестве критерия, с помощью которого мы объявляем эффекты экспериментальных факторов и их взаимодействия статистически значимыми.

Неявным в использовании ANOVA является существование аддитивной математической модели, используемой для объяснения компонентов вариации

в наблюдениях. Мы будем называть это "статистической моделью". Простейшая статистическая модель:

$$y_i = m + e_i \quad (2.10)$$

В этом примере каждое наблюдение разбивается только на две составляющие: великое среднее всех наблюдений, m , и случайную составляющую, e . Каждое наблюдение имеет начальную точку, великое среднее население, а затем подвергается случайному отклонению, ведущему к его конечному значению. Великое среднее значение не изменяется от наблюдения к наблюдению, тогда как случайная составляющая изменяется и подписывается соответственно. Хотя эта модель наводит на размышления, она допускает все вариации наблюдений к случайным источникам и ни одного к факторам, и поэтому не очень полезна при анализе результатов эксперимента.

Далее мы обратимся к статистической модели, используемой для анализа данных эксперимента с одним фактором, а именно фактором A .

$$y_{i,j} = m + a_i + e_{i,j} \quad (2.11)$$

На (2.11) обратите внимание на введение подписанного Альфа-термина, который обозначает влияние I -го уровня лечения одного и единственного фактора, рассматриваемого в модели. Эксперимент будет включать в себя один или несколько запусков на каждом из уровней лечения фактора A . Все наблюдения на данном уровне лечения анализируются с использованием одного и того же значения для a . Поскольку в этом эксперименте есть только один фактор, количество комбинаций лечения-это просто количество уровней лечения этого фактора. Дисперсионный анализ, основанный на этой статистической модели, приведет к созданию таблицы ANOVA, которая разделит вариацию наблюдений на ту, которая обусловлена обработкой a , и ту, которая обусловлена случайной вариацией.

$$y_{i,j,k} = m + \alpha_i + \beta_j + \iota(\alpha\beta)_{i,j} + e_{i,j,k} \quad (2.12)$$

На (2.12) мы переходим к модели, в которой рассматриваются два фактора. Обратите внимание, что теперь мы включаем термин для взаимодействия между фактором A и фактором B , обозначаемый $\iota(ab)$. Этот термин может отличаться в каждой комбинации лечения, и поэтому он выписывается дважды, так как есть два фактора. Вспомним выше, что когда действие двух факторов вместе не является суммой их отдельных эффектов, то между ними существует взаимодействие. Сила взаимодействия между факторами a и b обозначается "AB" и будет отражена в строке таблицы ANOVA.

В полный факторный эксперимент были бы включены все условия взаимодействия. Точно так же, как полная 2-факторная модель на 2.12 имеет 5

членов (RHS), полная 3-факторная модель имеет 8 членов, а полная 4-факторная модель имеет 16 членов. Вся эта информация обычно представлена в результирующей таблице ANOVA. GPSS World может обрабатывать до 6-факторных моделей с включением до 3-х стороннего взаимодействия.

Пользовательские эксперименты более гибки, но мы должны создавать и запускать их самостоятельно. Тем не менее, GPSS World предоставляет большую поддержку на этом пути. Конечная цель состоит в том, чтобы вы предоставили данные, необходимые для процедуры GPSS World ANOVA library, которая будет анализировать наши эксперименты с использованием до 6 факторов, включая 2-и 3-сторонние взаимодействия между факторами. Главное требование GPSS World ANOVA заключается в том, что мы должны передать ему название GPSS-матрицы, в которой были сохранены результаты нашего эксперимента.

Есть две важные вещи, которые нужно знать о матрицах результатов. Во-первых, прежде чем вы начнете свой эксперимент, вы должны инициализировать элементы вашей матрицы в неопределенное состояние. Вот типичное утверждение для этого:

`INITIAL MyResultMatrix,UNSPECIFIED`

Это делает его понятным для рутины ANOVA, когда запуск эксперимента не был завершен.

Во-вторых, если наш эксперимент имеет более одного фактора, нам нужно сделать его симметричным, чтобы программа GPSS World ANOVA могла четко анализировать дисперсию. Технический термин для этой особой симметрии - "ортогональность". Все, что нам нужно сделать, чтобы достичь ортогональности, - это посмотреть на все пары факторов в вашем эксперименте. Если в каждой паре каждый уровень обработки первого фактора появляется одинаковое количество раз в пределах каждого уровня обработки второго фактора, то эксперимент будет ортогональным. Если мы сделаем все симметричным, то оно будет ортогональным.

На рисунке 7 ниже представлена таблица ANOVA, представленная компанией GPSS World. Как было объяснено выше, отдельные заглавные буквы обозначают факторы, а комбинации заглавных букв - взаимодействия.

Во-первых, посмотрите на нижнюю часть стола. Общая сумма квадратов должна быть разделена, а компоненты должны быть связаны с эффектами факторов и их взаимодействиями. Все, что осталось, то есть остаточная сумма квадратов, отображается в предыдущей строке с надписью "ошибка". Для оценки стандартной погрешности эксперимента используется средняя сумма квадратов члена ошибки.

ANOVA					
Source of Variance	Sum of Squares	Degrees of Freedom	Mean Square	F	Critical Value of F (p=.05)
A	28.000	2	14.000	5.600	3.89
B	21.000	3	7.000	2.800	3.49
AB	11.000	6	1.833	0.733	3.00
Error	30.000	12	2.500		
Total	90.000	23			

Treatment Level A B	Count	Mean	Minimum	Maximum	95% C.I. (SE)
1 1	2	3.000	2.000	4.000	[0.564, 5.436]
1 2	2	1.000	1.000	1.000	[-1.436, 3.436]

01/27/01 13:26:44 1.5811388

Рисунок 2.7 – Таблица дисперсионного анализа

Иногда эксперимент не сможет обнаружить эффект, даже если он действительно существует. Одна из наших целей-сделать это как можно менее вероятным. Согласно таблице ANOVA, есть два способа сделать эксперимент более эффективным для обнаружения реальных эффектов. Для получения более положительных результатов нам потребуется либо большая статистика F, либо меньшее критическое значение F. В экспериментах в естественных науках это делается путем проведения сравнений в максимально однородной среде, что называется "блокировкой". Однако в имитационных исследованиях к этой цели, возможно, лучше всего подходит выявление дополнительных факторов, которые должны быть включены в эксперимент. Дисперсионный анализ - это статистический метод анализа результатов наблюдений, зависящих от различных одновременно действующих факторов, выбора наиболее важных факторов и оценки их влияния.

Одним из наиболее распространенных видов экспериментов является сравнение двух технологических методов или двух способов обработки. Существует несколько способов анализа такого эксперимента в зависимости от информации, полученной как от популяции, так и от выборки. Одним из наиболее простых методов оценки нового технологического метода является построение результатов на диаграмме SPC, которая также включает исторические данные из базового процесса с установленными контрольными пределами.

Затем примените стандартные правила для оценки неконтролируемых условий, чтобы увидеть, был ли процесс сдвинут. Возможно, Вам потребуется собрать несколько подгрупп данных, чтобы сделать определение, хотя одна подгруппа может выйти за пределы существующих контрольных пределов. Вы можете обратиться за помощью к модулю статистических диаграмм управления процессами.

Альтернативой подходу к контрольной диаграмме является использование F-критерия (F-ratio) для сравнения средств альтернативного лечения. Это делается автоматически с помощью функции ANOVA (дисперсионный анализ) статистического программного обеспечения.

Два других подхода направлены на увеличение степеней свободы термина ошибки. Первый - это просто увеличить количество повторов в эксперименте. Это обычно самый дорогой подход, но он может быть довольно эффективным. Вторая возможность связана с проектированием эксперимента и статистической моделью, лежащей в основе дисперсионного анализа. Средний квадрат ошибки-это фактически остаточный член, оставшийся после удаления всех остальных квадратов. Если мы сможем найти приемлемый способ оставить больше данных после устранения эффектов, мы получим оценку стандартной ошибки, которая имеет больше степеней свободы. Результирующее критическое значение F будет меньше, тем самым увеличивая мощность анализа. Именно это мы и пытаемся сделать, когда решаем игнорировать некоторые из взаимодействий.

Ни один из методов, которые мы рассматриваем, не будет искажать случайность, введенную для имитации случайности реального мира. Методы уменьшения дисперсии, которые делают это, не должны использоваться здесь. Намеренное уменьшение внутренней изменчивости наблюдений привело бы к завышению статистики F.

Во многих многофакторных экспериментах мы будем выбирать удаление взаимодействий высшего порядка, чтобы улучшить доступную информацию об основных факторах и взаимодействиях низкого порядка. Вы можете сделать это, используя третий аргумент процедуры ANOVA. Если есть разумная вероятность того, что он есть, нам было бы лучше добавить реплики для улучшения статистики, а не ограничивать анализ взаимодействиями низкого порядка. Реплики становятся известными процедуре ANOVA в ее втором аргументе.

$$y_{i,j,k} = m + \alpha_i + \beta_j + e_{i,j,k} \quad (2.12)$$

На 2.12 показана статистическая модель, в которой был удален термин 2-стороннего взаимодействия. Если верно, что эффект взаимодействия не существует, то мы улучшили четкость дисперсионного анализа, предоставив больше степеней свободы для среднего квадрата ошибки. Когда вы решите применить этот метод, вы просто измените 3-й аргумент вызова процедуры библиотеки GPSS World ANOVA, чтобы удалить взаимодействия более высокого порядка из анализа. Эта встроенная процедура может обрабатывать до 3-х сторонние взаимодействия и позволяет отказаться от 2-х или 3-х стороннего взаимодействия и выше, чтобы улучшить проверку значимости других эффектов.

При использовании процедуры библиотеки ANOVA необходимо определить матричную сущность GPSS, называемую матрицей результатов, для хранения отдельных результатов каждого запуска. Если бы вы анализировали более одной метрики, то получили бы более одной матрицы результатов. Поскольку объекты матрицы мира GPSS могут иметь до 6 измерений, каждое из которых имеет любой размер, результирующая матрица также ограничена 6 измерениями.

Когда вы пишете свой собственный эксперимент GPSS World, самое важное решение, которое вы примете, - это выбрать факторы эксперимента. Это те количества, которые вы можете контролировать, чтобы оптимизировать свою систему. Каждый фактор должен быть представлен пользовательской переменной, которая принимает уровни обработки в качестве значений. После выполнения каждого моделирования в эксперименте необходимо поместить полученное значение в матрицу результатов, чтобы данные можно было проанализировать с помощью процедуры ANOVA.

Положение в Матрице результатов для каждого результата определяется комбинацией методов лечения. Например, если в эксперименте рассматриваются 4 типа машин и две скорости каждой из них, то результат моделирования третьей машины на высокой скорости будет входить в матрицу результатов в положении [16]. На самом деле, поскольку реплики обычно помечаются на конце, первый запуск в этой комбинации лечения будет идти в элемент [15] матрицы результатов. Каждый размер матрицы результат анализируется как фактор, с помощью процедуры дисперсионного анализа библиотеки.

Мы можем исследовать любой эффект, задав ему его собственное измерение в Матрице результатов. При желании мы могли бы даже разделить потоки случайных чисел и рассматривать их по отдельности как отдельные факторы. Каждое случайное число семян будет представлять собой отдельный уровень обработки. Это позволило бы получить информацию об относительной важности каждого случайного процесса ввода. Обычно, однако, мы используем одно измерение для реплик и меняем все семена от одного реплицирования к другому в пределах одной комбинации обработки. Затем мы делаем известна

процедура дисперсионного анализа, какое измерение используется для повторностей. Это делается во втором аргументе.

Не все эксперименты нуждаются в повторном измерении. ANOVA часто может извлечь оценку стандартной ошибки без нее. Тогда дисперсионный анализ не сможет вычислить статистику F для таблицы ANOVA. В таких случаях вам придется либо добавить реплицирующие прогоны в эксперимент, либо удалить высокоуровневые взаимодействия из статистической модели, чтобы получить достаточно остаточных данных для оценки стандартной ошибки. Оба этих метода требуют изменения вызова процедуры библиотеки ANOVA.

Руководящий принцип, часто используемый при разработке экспериментов, таков: контролируйте все факторы, которые вы можете; то, что вы не можете контролировать, блокируйте; то, что вы не можете блокировать, рандомизируйте. Это относится и к использованию случайных потоков при моделировании дискретных событий.

Рассматривать потоки случайных чисел как факторы - значит выделять измерение для каждого потока и использовать каждое отдельное семя в качестве уровня обработки. Хотя таким образом можно рассматривать потоки случайных чисел как факторы и изучать их относительные эффекты, лучше всего это сделать в отдельном исследовании. Для основного исследования основных эффектов и взаимодействий мы больше заинтересованы в использовании наших потоков случайных чисел для представления источников естественной и неизбежной изменчивости, оставляя размеры матрицы результатов для изучаемых факторов.

Наиболее распространенный подход объединяет все потоки случайных чисел в один псевдофактор, где каждый уровень обработки представляет собой набор различных семян для всех потоков случайных чисел. Затем для этой цели из Матрицы результатов выделяется одно измерение-измерение репликации, которое становится известным процедуре библиотеки ANOVA. Это имеет некоторое сходство с блокировкой, где все сравнения выполняются в каждой однородной среде. Затем библиотечная процедура ANOVA может обработать это измерение репликации отдельно, чтобы лучше оценить стандартную ошибку эксперимента. Затем эксперимент состоит из экспериментальной "клетки" для каждой комбинации лечения, с несколькими повторными запусками внутри каждой клетки.

Если эти подходы слишком дороги или в Матрице результатов нет лишнего измерения, то запуски должны быть "полностью рандомизированы". Это означает, что рандомизация никоим образом не должна ограничиваться и что семена случайных чисел не должны повторяться в одном потоке случайных чисел. Если эта процедура не обеспечивает достаточных степеней свободы для анализа, возможно, Вам также потребуется ограничить уровень взаимодействий, которые будут включены в статистическую модель.

3 Эксперименты с имитационными моделями

3.1 Стратегическое планирование экспериментов

"Черный ящик" является абстрактной схемой исследуемой системы (рис.1). Подразумевается, что исследователь имеет возможность наблюдать входы и выходы "черного ящика" (имитационная модель).

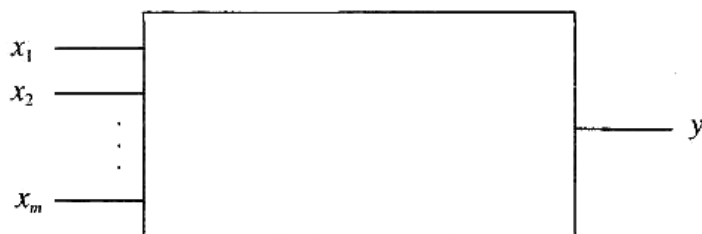


Рисунок 3.1 - Абстрактная схема системы

Термин эксперимент определяется как систематическая процедура, выполняемая в контролируемых условиях с целью обнаружения неизвестного эффекта, проверки или установления гипотезы или иллюстрации известного эффекта. При анализе процесса часто используются эксперименты для оценки того, какие входные данные процесса оказывают существенное влияние на выходные данные процесса и каким должен быть целевой уровень этих входных данных для достижения желаемого результата (выхода). Эксперименты могут быть разработаны различными способами для сбора этой информации.

Цель эксперимента-найти функцию y , где значение отклика определяется следующим образом

$$y = f(x_1, x_2, \dots, x_m) + e(x_1, x_2, \dots, x_T), \quad (3.1)$$

где $f(x_1, x_2, \dots, x_T)$ - функция отклика (неслучайная функция факторов);

$e(x_1, x_2, \dots, x_T)$ - ошибка эксперимента (случайная величина);

x_1, x_2, \dots, x_T - определенное сочетание уровней факторов из факторного пространства.

Очевидно, что y является случайной переменной, так как зависит от случайной величины $e(x_1, x_2, \dots, x_T)$. Дисперсия $D[y]$, которая характеризует точность измерений, равна дисперсии ошибки опыта: $D[y] = D[e]$.

Главная цель стратегического планирования эксперимента – это определение число наблюдений и комбинаций факторных уровней в них для получения более полной и достоверной информации о поведении исследуемой системы.

Как правило, экспериментатор сталкивается с целым списком факторов и должен решить, какие из них важны. Мы бы хотели это сделать:

- 1) определить основные факторы.

2) определить область интереса для основных факторов.

3) разработать полезные модели в области оптимальных настроек факторов.

4) подтвердить результаты.

Все эксперименты имеют проекты, но работа, связанная с фактическим планированием проекта, хорошая работа, обычно намного меньше, чем работа по фактическому выполнению эксперимента.

Возможно, самый распространенный экспериментальный метод заключается в изменении одного фактора за раз. Например, экспериментатор может зафиксировать все факторы, кроме фактора А, найти значение для этого фактора, которое максимизирует отклик, а затем повторить процесс для фактора В и т. д. Позже мы увидим, почему это не очень хороший план.

Напомним, что факторный эксперимент рассматривает все возможные методы планирования, состоящие из комбинаций факторов. Например, если есть два фактора с четырьмя и тремя уровнями соответственно, эксперимент будет иметь в общей сложности $4 \times 3 = 12$ обработок. Нахождение компромиссного решения, отвечающего этим требованиям, является задачей стратегического планирования эксперимента.

В двухуровневом факторном эксперименте каждый фактор исследуется на двух уровнях. Эти конструкции позволяют исследовать относительно большое количество факторов при относительно небольшом количестве экспериментальных запусков. Почему два уровня? Нам нужно, по крайней мере, два, чтобы оценить размер эффекта, но использование только двух экономит ресурсы.

Двухуровневые факторные конструкции:

- может выявить взаимодействие факторов.
- обеспечьте превосходную точность для оценок
- дать некоторое представление о диапазоне допустимости выводов о каждом факторе;
- ограничены с точки зрения размера проектного пространства, которое может быть исследовано.
- может быть модифицирован для уменьшения количества запусков, а также увеличения размер проектного пространства.
- являются отличными строительными блоками.

В основе наших методов лежат три принципа. Это иерархический принцип упорядочения:

- эффекты более низкого порядка, скорее всего, будут важны, чем эффекты более высокого порядка.
- эффект разреженности, или Принцип Парето: число важных эффектов, как правило, невелико.
- принцип наследственности эффекта: для того, чтобы эффект взаимодействия был значимым, (по крайней мере, один из) его родительских эффектов должен быть значимым.

Давайте рассмотрим двухфакторный факториал, чтобы мы могли установить некоторые базовые обозначения, а также ввести некоторые фундаментальные идеи.

Условно конструкции составляют 2^K

K = количество факторов,

где 2^K - количество сеансов комбинированного планирования.

Например, если $K = 2$, то для проектирования требуется $2^2 = 4$ прогона.

Теперь остается проблема, как определить, какие из этих факторов важны, "значимы" в научном, а не статистическом смысле.

Если у нас есть повторяющиеся наблюдения в каждой ячейке, то мы можем выполнить наш обычный тест значимости. Много раз в каждой клетке собирается только одно наблюдение. Как мы можем справиться с этой проблемой?

Один из подходов заключается в том, чтобы заранее дать обет, что нас интересуют только, скажем, основные эффекты и двустороннее взаимодействие. Суммы квадратов в оставшихся терминах можно было бы объединить в термин ошибки, а затем мы могли бы использовать проверку значимости.

Этот метод допустим, если члены более высокого порядка фактически равны нулю (что маловероятно, когда мы переходим к дробным факториальным конструкциям), но может не иметь большой силы в небольших экспериментах.

Альтернативным подходом является использование нормального вероятностного графика. Чтобы сделать это, сначала оцените эффекты. Затем вычислите совокупные проценты. Существует множество способов сделать это. Первая колонка ниже, вероятно, лучше всего, но вторая и третья легче, если вы делаете это вручную или с помощью калькулятора. Выводы, которые вы делаете не будут заметно отличаться от высокого порядка и делим 32 возможные комбинации на четыре блока одинакового размера (по восемь комбинаций в каждом блоке). Затем выберите один из этих блоков для эксперимента. Это означает, что мы должны смешивать три эффекта в процессе объединения наблюдений в блоки.

В работе были выбраны двух - и трехфакторные взаимодействия с общим фактором. В нашем случае я выбрала BCE и ADE с общим фактором E. При смешивании этих факторов автоматически были смешаны и ABCD. В результате мы имеем уравнение или определяющее разбиение, $I = BCE - ADE - ABCD$, которое используется при 32 наблюдениях для деления на четыре блока по восемь наблюдений в каждом [21]. Это значит, что мы имеем доступ к распознаванию таких факторов. Используя это уравнение как основу для смешивания в нашей работе, мы должны проделать то же и со следующими факторами:

$A=DE=BCD=ABCE,$
 $B=CE=ACD=ABDE,$
 $C=BE=ABD=ACDE,$
 $D=AE=ABC=BCDE,$
 $E=BC=AD=ABCDE,$
 $AB=CD=ACE=BDE,$
 $AC=BD=ABE=CDE.$

Как видите, имеем 7 групп смешивания. Это означает, что если взаимодействий между факторами нет или их можно пропустить, то мы можем по восьми наблюдениям получить информацию о каждом из пяти основных факторов (А, В, С, D и E) и использовать данные эффектов взаимодействий АВ и ВС при оценке об ошибках дисперсионного анализа.

3.2 Тактическое планирование эксперимента

Цель планирования тактического эксперимента -это определение требуемого количества пробегов модели в каждом наблюдении.

Так как моделирование считается статистическим экспериментом, то требуется получения достоверного результата.

В общем случае число пробегов наблюдений, требуемое для получения оценок наблюдаемого фактора с меньшей ошибкой, зависит от следующих факторов[24]:

- тип распределения наблюдаемой переменной;
- корреляция между элементами выборки;
- наличие и длительность переходного режима работы моделируемой системы.

Если отсутствует информация о необходимых факторах количество пробегов для каждого наблюдения, т. е. для каждой комбинации уровней факторов, выбранных на этапе стратегического планирования эксперимента, увеличивается для повышения точности оценок истинного значения наблюдаемой переменной.

Если случайные значения наблюдаемой переменной не коррелируют и их распределение не изменяется от запуска к запуску, то среднее значение выборки можно считать нормально распределенным. Для случая, когда целью моделирования является определение среднего значения некоторого случайного параметра, необходимое число запусков супермодели в каждом наблюдении определяется по формуле:

$$N_T = \frac{\sigma_\alpha^2}{\varepsilon^2} \cdot t_\alpha^2 \quad (3.1)$$

где ε - точность оценки;

σ_α - среднеквадратическое отклонение;

t_α - аргумент функции Лапласа, при заданном уровне значимости $\alpha/2$ находится по таблице 3.2, в которой даны наиболее актуальные пары α и t_α .

Таблица 3.2 - Аргумент функции Лапласа, при заданном уровне значимости $\alpha/2$

α	0,8	0,85	0,9	0,95	0,99	0,995	0,999
t_α	1,28	1,44	1,65	1,96	2,58	2,81	3,30

Если требуемое значение среднеекватрического отклонения σ_α до начала эксперимента неизвестно хотя бы ориентировочно, целесообразно выполнить пробное количество прогонов N_0 и вычислить на их основе выборочное среднеекватрическое отклонение или дисперсию, значение которой подставить в (3.1) и получить предварительную оценку числа прогонов N_T . Затем выполнить $N_T - N_0$ оставшиеся прогонов, периодически уточняя оценку и число прогонов N_T .

Для случая, когда целью моделирования является нахождение вероятности P исхода какого-либо события, требуемое число прогонов N_T модели в каждом наблюдении определяется по формуле

$$N_T = \frac{P \cdot (1 - P)}{\varepsilon^2} \cdot t_\alpha^2 \quad (3.2)$$

Параметр t_α находится также по таблице значений функции Лапласа в соответствии с заданной величиной достоверности (таблица 3.2). При вычислении N_T по формуле (2.2) значение вероятности P до эксперимента также неизвестно. Поэтому для определения N_T нужно поступить так, как и при нахождении N_T по формуле (3.1).

В ряде случаев для определения необходимого количества N_T прогонов можно использовать упрощенную формулу:

$$N_T = \frac{t_\alpha^2}{4\varepsilon^2} \quad (3.3)$$

Формула (3.3) ориентирована на «худший вариант», т. е. для $P = 0,5$. Следовательно, она, как правило, будет давать завышенные значения для N_T . Но при не очень высоких требованиях к точности применение этой формулы вполне оправдано, так как некоторая избыточность N_T будет скомпенсирована исключением процедуры определения ориентировочного значения P .

3.3 Дисперсионный анализ (отсеивающий эксперимент) на примере системы GPSS World

Двухпроцессорный вычислительный комплекс (ВК) состоит из трех автоматизированных рабочих мест (АРМ). Первое АРМ имеет высокий приоритет по сравнению с остальными, а второе — более высокий приоритет по отношению к третьему АРМ. В первый АРМ поступают запросы на обслуживание, интервалы между которыми распределены равномерно $[t_1, t_2]$. Интервалы времени поступления запросов от второго и третьего АРМ экспоненциально распределены с параметрами t_3, t_4 соответственно. Время обслуживания одного запроса 1-й категории, экспоненциально распределенное с параметром t_5 с, 2-й категории — t_6 с, 3-й категории — t_7 с. Вычислительный комплекс имеет накопитель емкостью L запросов и находится она в оперативной памяти.

Каждый процессор имеет конечную надежность. Длительность интервалов между отказами распределена экспоненциально с параметром t_8 с., а длительность восстановления — по нормальному закону с параметрами t_9 с и t_{10} с. В случае отказа хоть одного процессора обслуживающие и прерванные запросы будут потеряны.

Задача заключается в разработке имитационной модели с целью исследования работы двухпроцессорного ВК в течение 2 часов зависимости следующих факторов:

- вероятности потерь запросов от емкости накопителя,
- времени обработки запросов, интервалов
- времени выхода из строя;
- времени восстановления процессоров.

Необходимо провести дисперсионный анализ результатов моделирования.

Программа модели приведена ниже.

Listing:

```
VrMod EQU 3600
L_ EQU 1
T1_ EQU 11
T2_ EQU 3
T3_ EQU 10
T4_ EQU 12
T5_ EQU 5
T6_ EQU 7
T7_ EQU 9
T8_ EQU 10000
T9_ EQU 15
T10_ EQU 2
Koef EQU 1
Ver VARIABLE N$Term1/N$Met0
GENERATE T1_,T2_,2
ASSIGN 1,1
ASSIGN 2,(Exponential(11,0,T5_))
TRANSFER ,Met0
GENERATE (Exponential(2,0,T3_)),,1
ASSIGN 1,2
```


Продолжение Listing

```
ASSIGN      2,(Exponential(13,0,T6_))
TRANSFER   ,Met0
GENERATE    (Exponential(2,0,T4_))
ASSIGN      1,3
ASSIGN      2,(Exponential(15,0,T7_))
Met0 TEST E  P1,1,Met5
GATE FV     Cpu1,Term1
GATE NU     Cpu1,Met1
Met3 PREEMPT Cpu1,PR
ADVANCE     P2
RETURN      Cpu1
UNLINK      Nak,Met3,1
TERMINATE
Met1 GATE FV  Cpu2,TERM1
GATE NU     Cpu2,Met2
Met4 PREEMPT Cpu2,PR
ADVANCE     P2
RETURN      Cpu2
UNLINK      Nak,Met4,1
TERMINATE
Met5 GATE FV  Cpu1,Term1
GATE NU     Cpu1,Met6
Met7 SEIZE   Cpu1
ADVANCE     P2
RELEASE     Cpu1
UNLINK      Nak,Met7,1
TERMINATE
Met6 GATE FV  Cpu2,TERM1
GATE NU     Cpu2,Met2
Met8 SEIZE   Cpu2
ADVANCE     P2
RELEASE     Cpu2
UNLINK      Nak,Met8,1
TERMINATE
Met2 TEST L  CH$Nak,L_,Term1
LINK        Nak,P1
Term1      TERMINATE
GENERATE    ,,1
Met10 ADVANCE (Exponential(13,0,T8_))
FUNAVAIL   Cpu1,RE,Term1,,RE,Term1,RE,Term1
ADVANCE    (Normal(4,T9_#Kcoef,T10_#Kcoef))
FAVAIL     Cpu1
TRANSFER   ,Met10
GENERATE    ,,1
Met9 ADVANCE (Exponential(13,0,T8_))
FUNAVAIL   Cpu2,RE,Term1,,RE,Term1,RE,Term1
ADVANCE    (Normal(4,T9_#Kcoef,T10_#Kcoef))
FAVAIL     Cpu2
TRANSFER   ,Met9
GENERATE    VrMod
SAVEVALUE  Ver,V$Ver
TERMINATE  1
```

Для проведения дисперсионного анализа необходимо создать новую модель в системе GPSS

Для этого выбрала Edit/InsertExperiment/ Screening ...

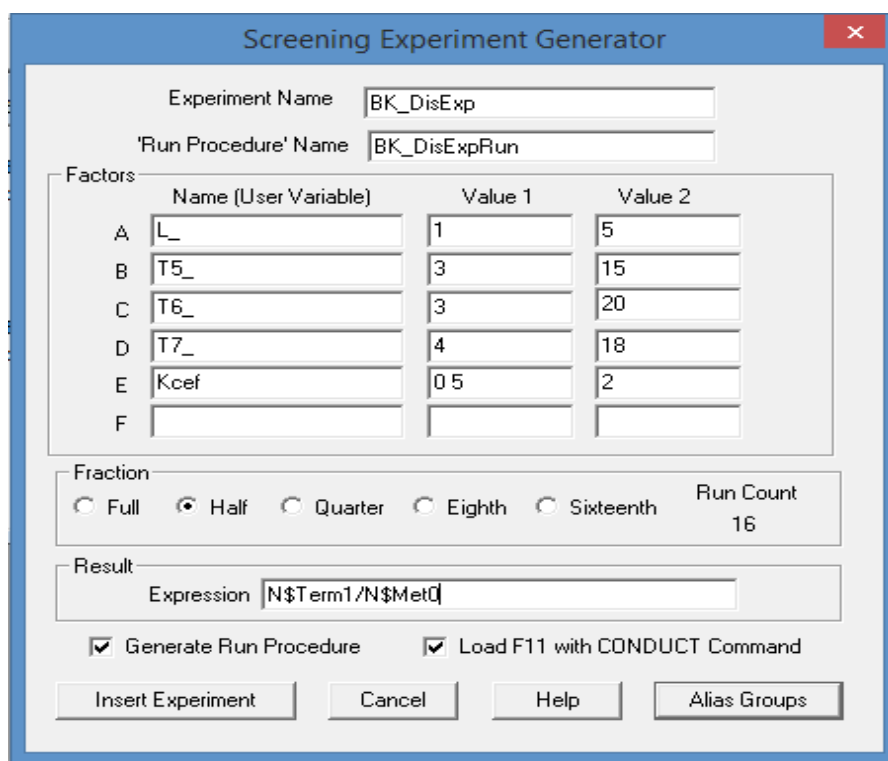


Рисунок 3.2 - Диалоговое окно Screening Experiment Generator (Генератор отсеивающего эксперимента)

В поле Experiment Name (Имя эксперимента) и Run Procedure Name (Имя процедуры запуска) пишем BK_DisExp и BK_DisExpRun соответственно.

Далее требуется настроить группу полей Factors (Факторы) в примере Listing для определения вероятности потерь запросов, которые поступают с АРМ. Имеем следующие факторы, влияющие на вероятность потери запросов в системе:

L - емкость накопителя;

T5 - среднее время обработки запросов от АРМ 1;

T6 - среднее время обработки запросов от АРМ2;

T7 - среднее время обработки запросов от АРМ 3;

Kcoef - коэффициент изменения среднего времени восстановления T9_ и среднеквадратического отклонения T10_ времени восстановления.

Название каждого фактора должно соответствовать именем переменной.. Это я учла при разработке Listing для данного эксперимента. Для того чтобы изменить среднее время восстановления T9_ и среднеквадратическое отклонение T10_ времени восстановления добавили переменную Kcoef. Следовательно это уменьшает количество фактора от 2 до 1 (Kcoef)[25].

В среде GPSS World имеем возможность делать эксперименты с помощью дисперсионного анализа 6 разных факторов, которые влияют на работу функции. При этом каждый фактор должен иметь 2 уровня: нижний и верхний. В источниках рекомендуют выбирать уровни, значительно

отдаленного друг от друга. Это гарантирует получения значительно отличающихся откликов в конечном результате.

В своем эксперименте я выбрала 5 факторов, каждый с 2 значениями.

Следующим шагом я настроила группу Fraction (Часть дробного эксперимента). В GPSS World эксперимент можно проводить с ПФЭ или ДФЭ. С помощью Группы Fraction мы можем задать и выбрать один из двух вышеописанных. Справа под Run Count появится число 16 - количество наблюдений(прогонов), которые необходимо сделать в эксперименте.

В поле Expression (Выражение) группы Result (Результат) вводим следующее выражение по которому вычисляется вероятность потерь запросов с АРМ:

$$N\$Term1/N\$Met0$$

После группы Result (Результат) идут два флажка, которые позволяют функцию выбора опции. С помощью опции Generate Run Procedure создается вместе с экспериментом стандартная процедура запуска и мы можем откорректировать ее исходя из своих данных.

Для стратегического планирования эксперимента заранее до создания эксперимента изучила основы группы смешивания. [26]. Для этого требуется выбрать Alias Groups (Группы смешивания). Появится диалоговое окно Alias Groups (Группы смешивания) (рисунок 3). Сперва мы находим отсутствующие факторы в списке, только потом неразличимые факторы, потому что они расположены в одной группе смешивания. Факторы обозначаются одной буквой. Например, 5 фактор обозначается буквой E, а взаимное взаимодействие 1 и 5 факторов - AE.

Из рисунка 3 мы видим, что отсутствует фактор E, соответственно в нем не будет никакой информации. Факторы C и D являются неразличимыми, так как находятся в одной группе смешивания.

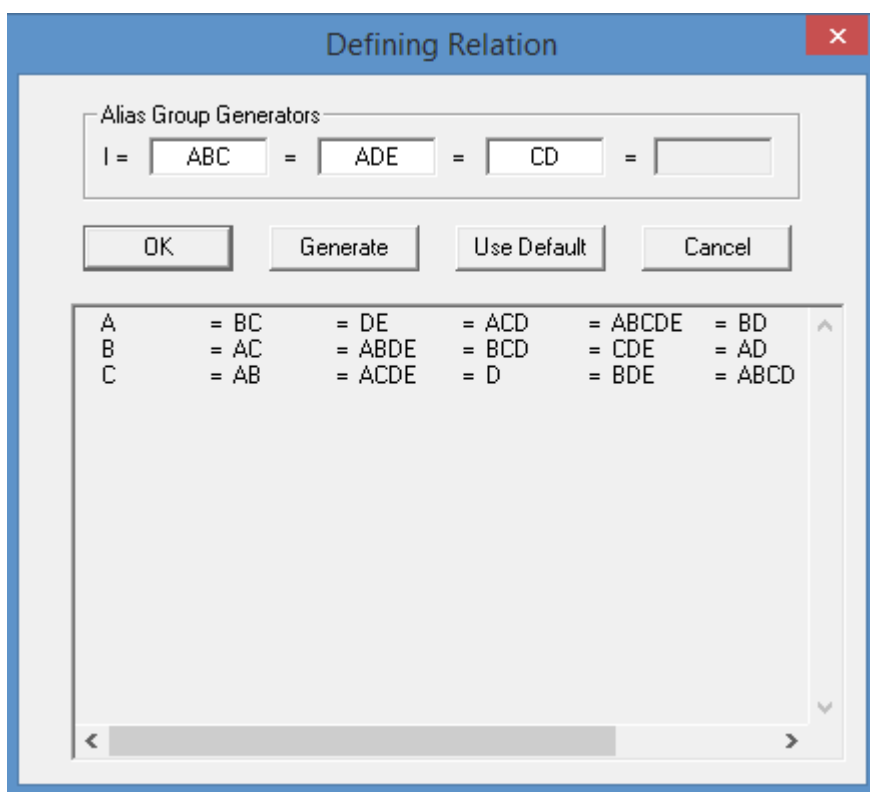


Рисунок 3.3 - Группы смешивания

В поле Alias Generators мы можем изменить распределение факторов по группам смешивания, а значит, и получаемых эффектов. Для этого нужно изменить порождающие функции и нажать кнопку Generate. В качестве порождающих функций я использовала наборы взаимодействий высших порядков. При создании группы смешивания в GPSS World сперва программа расширяет множество порождающих функций и формирует все возможные комбинации при помощи деления по модулю два и добавляя их к расширенному множеству порождающих функций. Например, если ABD и ADE - порождающие функции, их результат - BE будет нанесен в расширенное множество ($ABD * ADE = AABDDE = (AA)B(DD)E = BE$). Эффекты расширенного множества порождающих функций не включены в заключительный отчет. Это значит, что нельзя выбирать порождающие функции, которые приведут к появлению фактора в расширенном множестве порождающих функций. Нельзя также использовать порождающую функцию тождества I. Если не выполнено хоть одно условие из вышеописанных, программа выдаст системную ошибку.

Чтобы вернуться к исходным порождающим функциям, предложенным GPSS World, нужно нажать кнопку По умолчанию.

В нашем примере имеются только три группы смешивания, поэтому невозможно различить эффекты каждого из пяти факторов. Закроем диалоговое окно Группы смешивания, нажав кнопку Отмена.

В диалоговом окне Генератора отсеивающего эксперимента в группе Часть дробного эксперимента установила Quarter (1/4). Снова открываем диалоговое окно Группы смешивания.

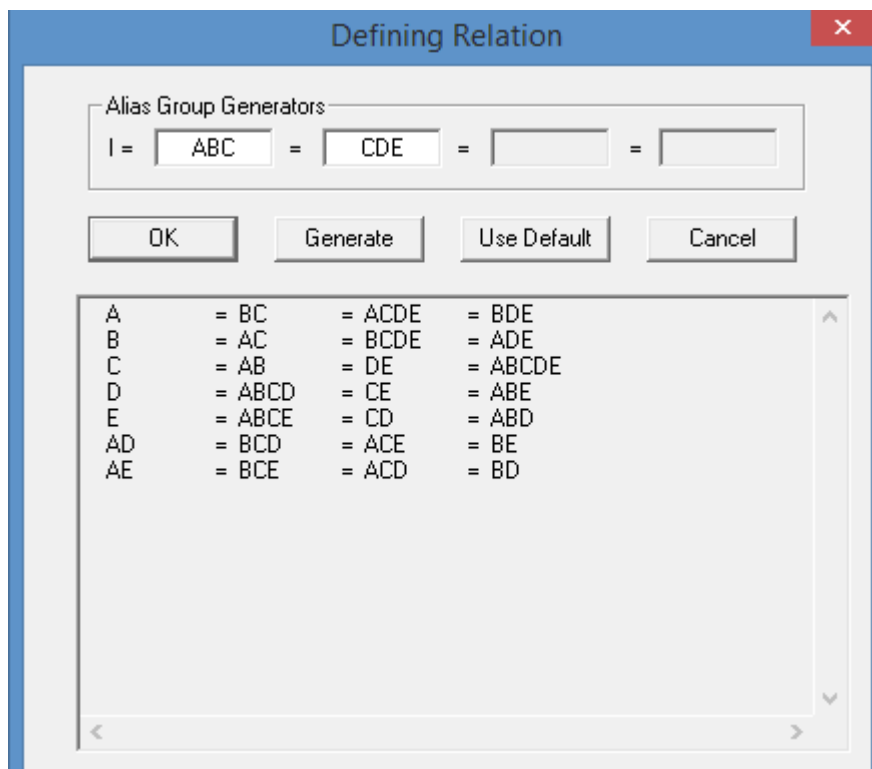


Рисунок 3.4 - Группы смешивания $\frac{1}{4}$

Как видно на рисунке факторы находятся в отдельных группах. Однако невозможно будет судить о влиянии на отклик взаимодействий двух факторов. В некоторых случаях этого будет достаточно. Иначе можно попробовать различные порождающие функции для выделения одного или двух двухфакторных взаимодействий. Так как в нашем эксперименте нету необходимости этого и мы закроем диалоговое окно Группы смешивания, нажав кнопку Отмена.

В диалоговом окне Screening Experiment Generator в группе Fraction установила Half (1/2).

Имеющий сейчас GPSS World план эксперимента большое подходит, чем ранее показанные группы смешивания. Вместо 32 нужно будет выполнить 16 наблюдений. В плане выделены все факторы и двухфакторные взаимодействия нескольких факторов. Создаем эксперимент с этим планом, нажав кнопку Ok.

Следующим шагом работы является создания Plus-операторов. В левой нижней части диалогового окна Screening Experiment Generator (нажимаем кнопку Вставить эксперимент.

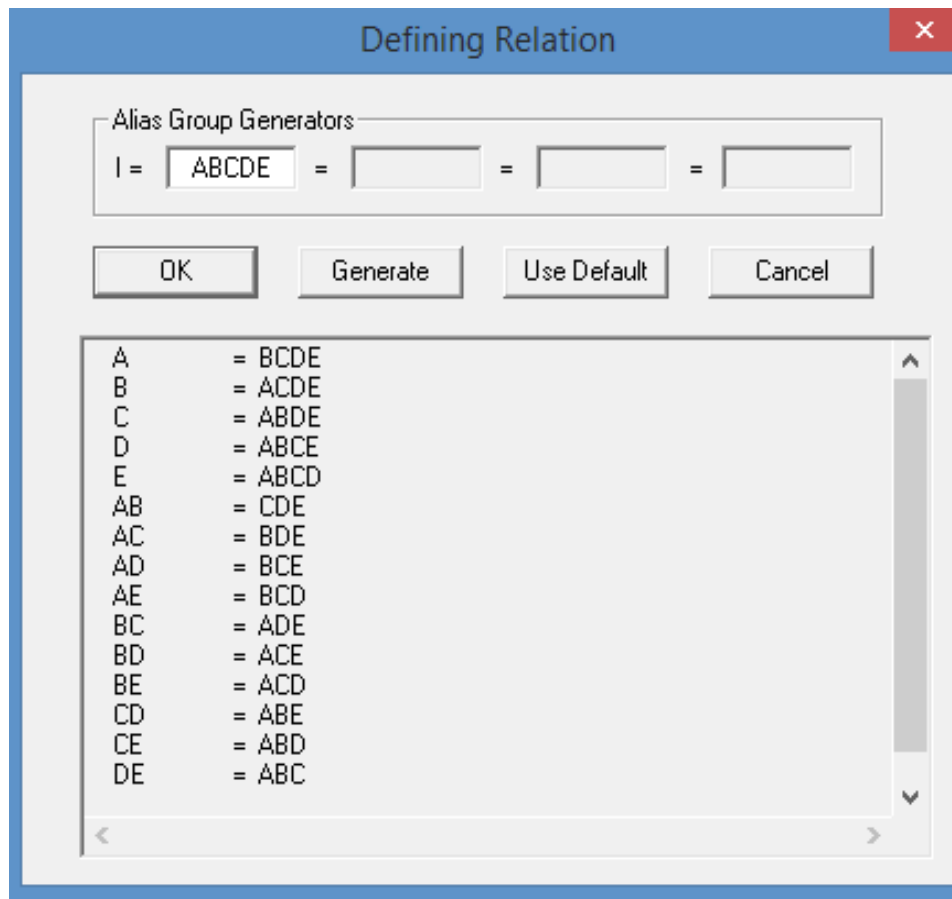


Рисунок 3.5 - Группы смешивания $\frac{1}{2}$

Так как была выбрана опция Generate Run Procedure, то создана стандартная процедура запуска. Появится диалоговое окно (рисунок 6), дающее возможность изменить ее согласно своим требованиям.

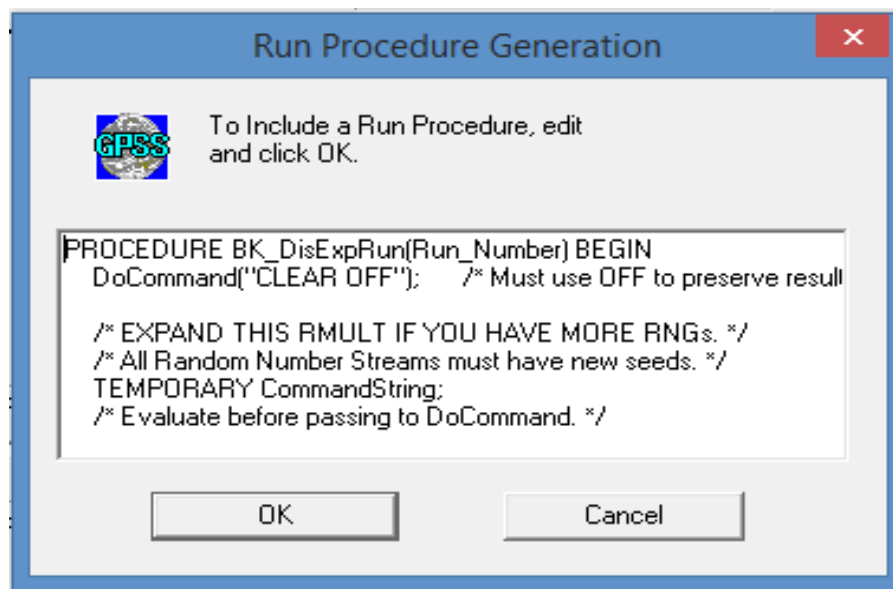


Рисунок 3.6 - Диалоговое окно стандартной процедуры Запуска

В конце запуска стандартной процедуры мы видим условие моделирования в каждом прогоне. (рисунок 7). Откорректировала их так, как указано на рисунке 8.

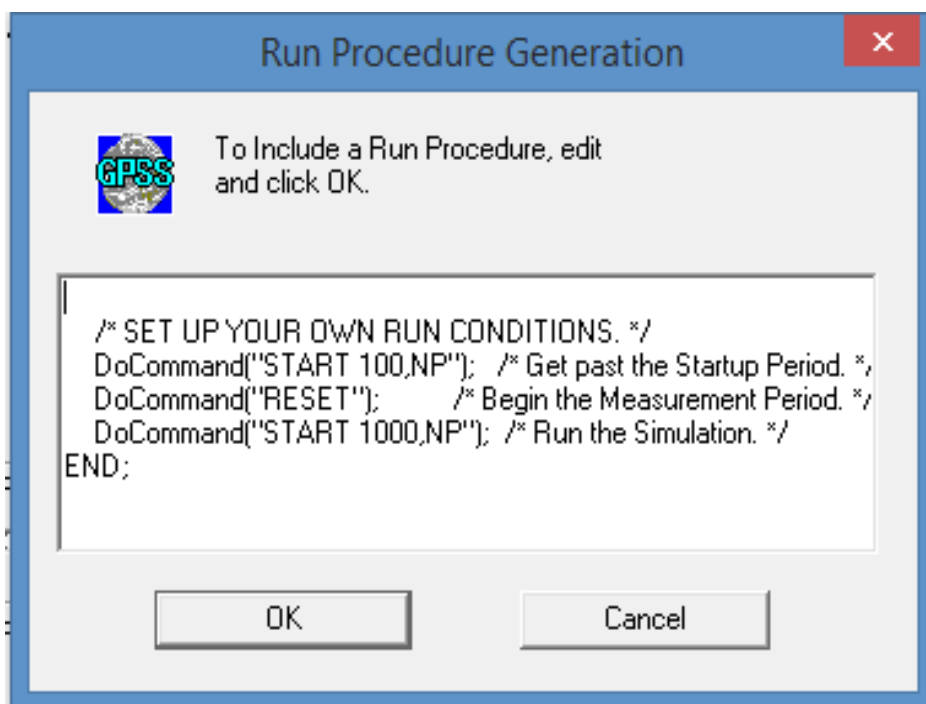


Рисунок 3.7 - Условия стандартной процедуры Запуска по умолчанию

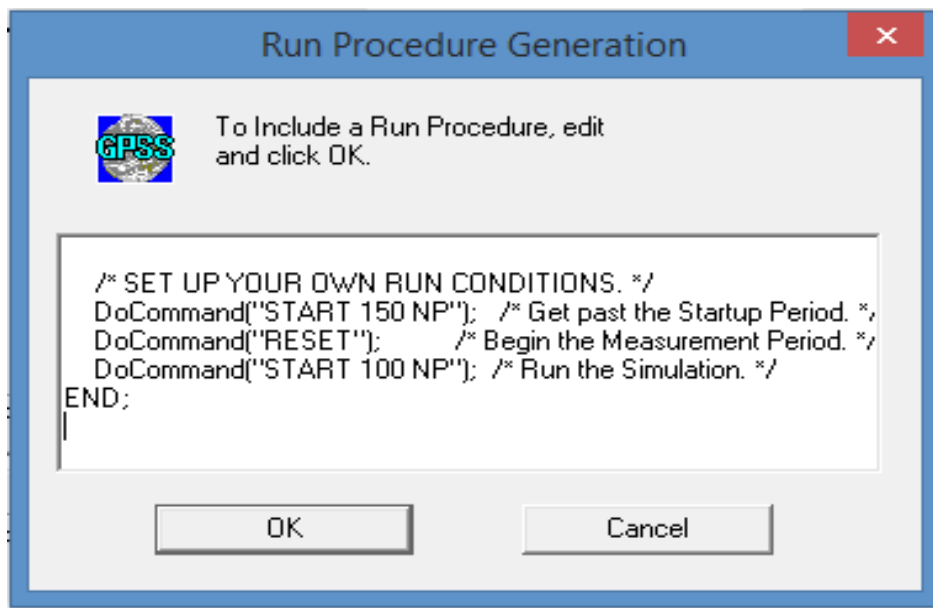


Рисунок 3.8 -Условия стандартной процедуры после корректировки

После всей корректировки нажала ОК. Сгенерированный Plus-эксперимент представлен ниже. В начале автоматически сгенерированного эксперимента определяется и инициализируется в неопределенное состояние матрица результатов. Далее имеем Plus-операторы, которые для каждого из наблюдений определяют сочетания уровней факторов. В рассматриваемом примере таких сочетаний шестнадцать (1/2 часть ПФЭ).

Проведем эксперимент. Для вызова эксперимента предназначена команда CONDUCT [28].

Провели трансляцию, т.е. создали объект «Процесс моделирования. После этого нажимаем [F11]. Эксперимент начинает работать. В ходе работы эксперимента программа выдает отчет, который после готовности записывается в окно Journal (Журнал) созданного объекта «Процесс моделирования» (рисунок 9).


```

GPSS World - [Untitled]
File Edit Search View Command Window Help
05/25/20 22:18:34 Model Translation Begun.
05/25/20 22:18:34 Ready.
05/25/20 22:18:34 BK_DISEXP Procedure registered.
05/25/20 22:18:34 BK_DISEXP_GETRESULT Procedure registered.
05/25/20 22:18:34 BK_DISEXP_RUN Procedure registered.
05/25/20 22:23:24 CONDUCT BK_DisExp]
05/25/20 22:23:24 **** Experiment in Progress. ****
05/25/20 22:23:24 Simulation in Progress.
05/25/20 22:23:26 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:26 Simulation in Progress.
05/25/20 22:23:27 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:27 "Run 1. Yield=.06064299862867591. L_=1; T5_=3; T6_=3; T7_=4; Koef=.5;"
05/25/20 22:23:27 Simulation in Progress.
05/25/20 22:23:29 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:29 Simulation in Progress.
05/25/20 22:23:30 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:30 "Run 2. Yield=.305708168376828. L_=1; T5_=3; T6_=3; T7_=18; Koef=2;"
05/25/20 22:23:30 Simulation in Progress.
05/25/20 22:23:32 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:32 Simulation in Progress.
05/25/20 22:23:33 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:33 "Run 3. Yield=.3922406839812873. L_=1; T5_=3; T6_=20; T7_=4; Koef=2;"
05/25/20 22:23:33 Simulation in Progress.
05/25/20 22:23:35 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:35 Simulation in Progress.
05/25/20 22:23:36 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:36 "Run 4. Yield=.5187137105747638. L_=1; T5_=3; T6_=20; T7_=18; Koef=.5;"
05/25/20 22:23:36 Simulation in Progress.
05/25/20 22:23:38 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:38 Simulation in Progress.
05/25/20 22:23:39 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:39 "Run 5. Yield=.2828873602230371. L_=1; T5_=15; T6_=3; T7_=4; Koef=2;"
05/25/20 22:23:39 Simulation in Progress.
05/25/20 22:23:41 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:41 Simulation in Progress.
05/25/20 22:23:42 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:42 "Run 6. Yield=.4633186974021222. L_=1; T5_=15; T6_=3; T7_=18; Koef=.5;"
05/25/20 22:23:42 Simulation in Progress.
05/25/20 22:23:45 "Run 7. Yield=.5185912953640774. L_=1; T5_=15; T6_=20; T7_=4; Koef=.5;"
05/25/20 22:23:45 Simulation in Progress.
05/25/20 22:23:47 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:47 Simulation in Progress.
05/25/20 22:23:48 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:48 "Run 8. Yield=.614588680536683. L_=1; T5_=15; T6_=20; T7_=18; Koef=2;"
05/25/20 22:23:48 Simulation in Progress.
05/25/20 22:23:50 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:50 Simulation in Progress.
05/25/20 22:23:51 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:51 "Run 9. Yield=.005709484425457822. L_=5; T5_=3; T6_=3; T7_=4; Koef=2;"
05/25/20 22:23:51 Simulation in Progress.
05/25/20 22:23:53 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:53 Simulation in Progress.
05/25/20 22:23:55 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:55 "Run 10. Yield=.1518815433418804. L_=5; T5_=3; T6_=3; T7_=18; Koef=.5;"
05/25/20 22:23:55 Simulation in Progress.
05/25/20 22:23:57 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:57 Simulation in Progress.
05/25/20 22:23:58 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:23:58 "Run 11. Yield=.2852265013120966. L_=5; T5_=3; T6_=20; T7_=4; Koef=.5;"
05/25/20 22:23:58 Simulation in Progress.
05/25/20 22:24:00 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:24:00 Simulation in Progress.
05/25/20 22:24:01 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:24:01 "Run 12. Yield=.4565019426906265. L_=5; T5_=3; T6_=20; T7_=18; Koef=2;"
05/25/20 22:24:01 Simulation in Progress.
05/25/20 22:24:03 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:24:03 Simulation in Progress.
05/25/20 22:24:05 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:24:05 "Run 13. Yield=.1499519741165765. L_=5; T5_=15; T6_=3; T7_=4; Koef=.5;"
05/25/20 22:24:05 Simulation in Progress.
05/25/20 22:24:07 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:24:07 Simulation in Progress.
05/25/20 22:24:08 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:24:08 "Run 14. Yield=.3848057863396127. L_=5; T5_=15; T6_=3; T7_=18; Koef=2;"
05/25/20 22:24:08 Simulation in Progress.
05/25/20 22:24:11 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:24:11 "Run 15. Yield=.4707014824046772. L_=5; T5_=15; T6_=20; T7_=4; Koef=2;"
05/25/20 22:24:11 Simulation in Progress.
05/25/20 22:24:13 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:24:13 Simulation in Progress.
05/25/20 22:24:14 A Simulation in an Experiment has ended. Clock is 900000.000000.
05/25/20 22:24:14 "Run 16. Yield=.5893712000323187. L_=5; T5_=15; T6_=20; T7_=18; Koef=.5;"
05/25/20 22:24:14

```

Рисунок 3.9 - Окно Journal (Журнал) с отчетами по каждому опыту

Так как эксперимент включает 16 наблюдений по 100 прогонов в каждом из них, будет выдано 16 отчетов. После завершения всех наблюдений окончательные результаты будут выведены в виде таблицы (рисунок 10).

05/26/20 01:07:41 05/26/20 01:07:41 05/26/20 01:07:41	Alias Group	Effect	Sum of Squares	Degrees of Freedom	F - for Only Main Effects	Critical Value of F (p=.05)
05/26/20 01:07:41	A = BCDE	-0.083	0.027	1	19.193	4.96
05/26/20 01:07:41	B = ACDE	0.162	0.105	1	73.618	4.96
05/26/20 01:07:41	AB = CDE	0.012				
05/26/20 01:07:41	C = ABDE	0.255	0.260	1	182.140	4.96
05/26/20 01:07:41	AC = BDE	0.022				
05/26/20 01:07:41	BC = ADE	-0.027				
05/26/20 01:07:41	DE = ABC	0.012				
05/26/20 01:07:42	D = ABCE	0.165	0.109	1	76.060	4.96
05/26/20 01:07:42	AD = BCE	0.003				
05/26/20 01:07:42	BD = ACE	-0.007				
05/26/20 01:07:42	CE = ABD	0.016				
05/26/20 01:07:42	CD = ABE	-0.037				
05/26/20 01:07:42	BE = ACD	0.014				
05/26/20 01:07:42	AE = BCD	-0.013				
05/26/20 01:07:42	E = ABCD	-0.022	0.002	1	1.346	4.96
05/26/20 01:07:42	Error		0.014	10		
05/26/20 01:07:42	Total		0.518	15		
05/26/20 01:07:42	Grand Mean	0.353				
05/26/20 01:07:42 Experiment ended.						

Рисунок 3.10 - Результаты дисперсионного анализа

В конечной таблице каждый фактор и взаимодействие факторов записаны на отдельные строки. В каждой строке для данного Effect имеем значения основных вычислений и F-статистика. Чем больше значение F-статистики фактор имеет, тем сильнее влияет на работу системы[27].

Эффект, а, следовательно, и фактор, считается значимым, если его значения больше чем критическое ($p=0.05$). В нашем примере факторы А, В, С и D являются значимыми, так как их F-статистики больше критического значения и равняется 4,96. Наибольший эффект на вероятность потери запросов оказывает фактор С, так как он имеет самую большую статистику (182,140).

Исходя их дисперсионного анализа, можно сделать вывод, что при данных потоках и характеристиках процессоров вероятность потерь запросов в среднем составит 0,353. С целью уменьшения потерь запросов нужно продолжить исследование каждого значимого фактора и в первую очередь фактора С.

Заключение

Подводя итоги диссертационной работы, можно сделать следующее заключение: для уменьшения потерь запросов нужно исследовать влияния каждого значимого фактора и в первую очередь фактора, который имеет наибольшую F-статистику. Также по полученным результатам дисперсионного анализа было исследовано влияние не только одного фактора, но и смешанных групп.

В первой главе представлен общий анализ современного состояния планирования экспериментов. В теоретической части работы было рассмотрено имитационное моделирование в управлении сложными телекоммуникационными системами.

Изучению практических аспектов посвящена вторая глава – рассмотрено моделирование систем телекоммуникаций как систем массового обслуживания. Во второй части работы представлена сравнительная таблица частного случая СМО М/М/5/0.

В третьей главе проведен дисперсионный анализ на примере системы GPSS World. В результате отсеивающего эксперимента был получен отчет с коэффициентом влияния на потери запросов для каждого фактора и смешанных групп. Исходя из таблицы экспериментатор может повторить опыт с уменьшением значения значимых факторов для минимизации потери запросов, что облегчает работу группы и увеличивает возможность точного описания объекта. С помощью полученных данных при повторе дисперсионного анализа мы имеем полную информацию о влиянии каждого фактора, тем самым делая объект более управляемым и целенаправленным на определенные результаты.

Список литературы

- 1 Акопов А. С. Имитационное моделирование. М.: Юрайт, 2015.
- 2 Kendall D. G., Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. Ann. Math. Statistics, 1953, 24, №3, 338—354 (РЖМат, 1956, 3218). /Перевод с английского/. - Математика, сб. перев. ин. ст.3, 6, 1959)
- 3 Асеев А. А., Боев В. Д., Кулешов И. А., Сеченев Д. Н. Основы моделирования систем связи и автоматизации на GPSS/PC: Учеб. пособие. — СПб.: ВУС, 2000. — 230 с.
- 4 Асеев А. А., Боев В. Д., Кулешов И. А. и др. Основы моделирования систем связи и автоматизации в Delphi 4: Учеб. пособие. — СПб.: ВУС, 2003. — 232 с.
- 5 Боев В. Д. Имитационное моделирование на GPSS/PC системе и обеспечения войск: Учеб. пособие. — СПб.: ВАУ, 1999. — 186 с.
- 6 J. S. Carson II, "Introduction to Modeling and Simulation," presented at 2018 Winter Simulation Conference, December, 2018, pp. 1283-1289.
- 7 Боев В. Д., Таран С. П. Основы моделирования систем и военного назначения в Delphi 4: Учеб. пособие. — СПб.: ВАУ, 2000. — 218 с.
- 8 Боев В. Д., Филюстин А. Е. Датчик равномерно распределенных случайных чисел: Авт. св. № 138499, бил. N-° 10, 1988.
- 9 Боев В. Д., Филюстин А. Е. Устройство для определения прогнозных оценок случайного процесса: Авт. св. N-° 1383406, бюл. N-° 11, 1988.
- 10 Девятков В.В., Девятков Т.В., Федоров М.В. /Имитационные исследования в среде моделирования GPSS Studio: учеб. пособие; под общ. ред. В.В. Девяткова. М.: Вузовский учебник: ИНФРА-М, 2018, 283 с.
- 11 Кирпичников А.П. /Методы прикладной теории массового обслуживания//Казань: Изд-во Казан. гос. ун-та. 2011. 200 с.
- 12 Советов Б.Я., Яковлев С.А. Моделирование систем. - М.: Высшая школа, 1998.
- 13 Шварц М. Сети связи: Протоколы, моделирование и анализ. - М.: Наука, 1992.
- 14 Боев В.Д. Моделирование систем. Инструментальные средства GPSSWorld: Учеб. пособие. – СПб.: БХВ-Петербург, 2004.
- 15 Кудрявцев Е.М. GPSSWorld. Основы имитационного моделирования различных систем. – М.: ДМК Пресс, 2004.
- 16 Томашевский В., Жданова Е. Имитационное моделирование в среде GPSS. - М.: Бестселлер, 2003.
- 17 Учебное пособие по GPSSWorld. /Перевод с английского /.- Казань: Изд-во «Мастер Лайн», 2002.
- 18 K. Pawlikowski, H.-D. J. Jeong, and J.-S. R. Lee, "On credibility of simulation studies of telecommunication networks," IEEE Communications Magazine, vol. 40, no. 1, pp. 132-139, 2002.

19 A. M. Law and W. D. Kelton, Simulation modelling and analysis, third ed. New York: McGraw-Hill, 2017.

20 B. Schmeiser, "Simulation output analysis: A tutorial based on one research thread," presented at the 2019 Winter Simulation Conference, December 5-8, 2019, pp. 162-170.

21 Боев В. Д., Сыпченко Р. П. Компьютерное моделирование: Руководство по проектированию. — СПб.: ВУС, 2002. — 96 с.

22 E. K. Bowdon, "Using simulation to evaluate system performance " presented at Proceedings of the 11th workshop on Design automation 2019, pp. 359-365.

23 Рыжиков Ю. И. Имитационное моделирование: Теория и технологии. — СПб.: Корона принт; N.: Альтекс — А, 2014. — 384 с.

24 Томашевский В. Н., Хаданова Е. Г. Имитационное моделирование в среде GPSS. — N.: Бестселлер, 2003. — 4т6 с.

25 Советов Б. Я., Яковлев С. А. Моделирование систем и: учебник для вузов. — N.: Высшая школа, 200t.

26 Учебное пособие по GPSS World. /Перевод с англи йского/. — Казань: Изд-во «Настер Лайн», 2002. — 270 с.

27 Шрайбер Т. Дж. Моделирование на GPSS. — N.: Машиностроение, t980. —592 с.

28 R. McHaney, Computer simulation: a practical perspective. San Diego: Academic Press, 2019.

Приложение А

Результаты моделирования в среде GPSS

06/18/20 17:42:02 Model Translation Begun.
06/18/20 17:42:02 Ready.
06/18/20 17:42:02 BK_DISEXP Procedure registered.
06/18/20 17:42:02 BK_DISEXP_GETRESULT Procedure registered.
06/18/20 17:42:02 BK_DISEXP_RUN Procedure registered.
06/18/20 17:42:21 CONDUCT BK_DisExp[]
06/18/20 17:42:21 **** Experiment in Progress. ****
06/18/20 17:42:22 Simulation in Progress.
06/18/20 17:42:31 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:42:31 Simulation in Progress.
06/18/20 17:42:37 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:42:37 "Run 1. Yield=.06064299862867591. L_=1; T5_=3; T6_=3; T7_=4; Koef=.5;"
06/18/20 17:42:38 Simulation in Progress.
06/18/20 17:42:47 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:42:47 Simulation in Progress.
06/18/20 17:42:53 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:42:53 "Run 2. Yield=.305708168376828. L_=1; T5_=3; T6_=3; T7_=18; Koef=2;"
06/18/20 17:42:53 Simulation in Progress.
06/18/20 17:43:03 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:43:03 Simulation in Progress.
06/18/20 17:43:09 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:43:09 "Run 3. Yield=.3922406839812873. L_=1; T5_=3; T6_=20; T7_=4; Koef=2;"
06/18/20 17:43:09 Simulation in Progress.
06/18/20 17:43:18 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:43:18 Simulation in Progress.
06/18/20 17:43:24 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:43:24 "Run 4. Yield=.5187137105747638. L_=1; T5_=3; T6_=20; T7_=18; Koef=.5;"
06/18/20 17:43:25 Simulation in Progress.
06/18/20 17:43:34 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:43:34 Simulation in Progress.
06/18/20 17:43:40 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:43:40 "Run 5. Yield=.2828873602230371. L_=1; T5_=15; T6_=3; T7_=4; Koef=2;"
06/18/20 17:43:41 Simulation in Progress.
06/18/20 17:43:50 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:43:50 Simulation in Progress.
06/18/20 17:43:56 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:43:56 "Run 6. Yield=.4633186974021222. L_=1; T5_=15; T6_=3; T7_=18; Koef=.5;"
06/18/20 17:43:56 Simulation in Progress.
06/18/20 17:44:05 A Simulation in an Experiment has ended. Clock is 540000.000000.

06/18/20 17:43:56 Simulation in Progress.
06/18/20 17:44:05 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:44:05 Simulation in Progress.
06/18/20 17:44:11 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:44:11 "Run 7. Yield=.5185912953640774. L_=1; T5_=15; T6_=20; T7_=4; Koef=.5;"
06/18/20 17:44:12 Simulation in Progress.
06/18/20 17:44:21 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:44:21 Simulation in Progress.
06/18/20 17:44:27 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:44:27 "Run 8. Yield=.614588680536683. L_=1; T5_=15; T6_=20; T7_=18; Koef=2;"
06/18/20 17:44:27 Simulation in Progress.
06/18/20 17:44:37 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:44:37 Simulation in Progress.
06/18/20 17:44:43 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:44:43 "Run 9. Yield=.005709484425457822. L_=5; T5_=3; T6_=3; T7_=4; Koef=2;"
06/18/20 17:44:43 Simulation in Progress.
06/18/20 17:44:54 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:44:54 Simulation in Progress.
06/18/20 17:45:01 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:45:01 "Run 10. Yield=.1518815433418804. L_=5; T5_=3; T6_=3; T7_=18; Koef=.5;"
06/18/20 17:45:02 Simulation in Progress.
06/18/20 17:45:12 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:45:12 Simulation in Progress.
06/18/20 17:45:15 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:45:15 "Run 11. Yield=.2852265013120966. L_=5; T5_=3; T6_=20; T7_=4; Koef=.5;"
06/18/20 17:45:15 Simulation in Progress.
06/18/20 17:45:18 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:45:18 Simulation in Progress.
06/18/20 17:45:20 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:45:20 "Run 12. Yield=.4565019426906265. L_=5; T5_=3; T6_=20; T7_=18; Koef=2;"
06/18/20 17:45:20 Simulation in Progress.
06/18/20 17:45:31 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:45:31 Simulation in Progress.
06/18/20 17:45:37 A Simulation in an Experiment has ended. Clock is 900000.000000.
06/18/20 17:45:37 "Run 13. Yield=.1499519741165765. L_=5; T5_=15; T6_=3; T7_=4; Koef=.5;"
06/18/20 17:45:37 Simulation in Progress.
06/18/20 17:45:48 A Simulation in an Experiment has ended. Clock is 540000.000000.
06/18/20 17:45:48 Simulation in Progress.
06/18/20 17:45:55 A Simulation in an Experiment has ended. Clock is 900000.000000.

06/18/20 17:45:55 Simulation in Progress.
 06/18/20 17:46:05 A Simulation in an Experiment has ended. Clock is 540000.000000.
 06/18/20 17:46:05 Simulation in Progress.
 06/18/20 17:46:08 A Simulation in an Experiment has ended. Clock is 900000.000000.
 06/18/20 17:46:08 "Run 15. Yield=.4707014824046772. L_=5; T5_=15; T6_=20; T7_=4; Koef=2;"
 06/18/20 17:46:08 Simulation in Progress.
 06/18/20 17:46:16 A Simulation in an Experiment has ended. Clock is 540000.000000.
 06/18/20 17:46:16 Simulation in Progress.
 06/18/20 17:46:24 A Simulation in an Experiment has ended. Clock is 900000.000000.
 06/18/20 17:46:24 "Run 16. Yield=.5893712000323187. L_=5; T5_=15; T6_=20; T7_=18; Koef=.5;"
 06/18/20 17:46:24

06/18/20 17:46:24	Alias Group	Effect	Sum of Squares	Degrees of Freedom	F - for Only Main Effects	Critical Value of F (p=.05)
06/18/20 17:46:24	A = BCDE	-0.083	0.027	1	19.193	4.96
06/18/20 17:46:24	B = ACDE	0.162	0.105	1	73.618	4.96
06/18/20 17:46:24	AB = CDE	0.012				
06/18/20 17:46:24	C = ABDE	0.255	0.260	1	182.140	4.96
06/18/20 17:46:24	AC = BDE	0.022				
06/18/20 17:46:24	BC = ADE	-0.027				
06/18/20 17:46:24	DE = ABC	0.012				
06/18/20 17:46:24	D = ABCE	0.165	0.109	1	76.060	4.96
06/18/20 17:46:25	AD = BCE	0.003				
06/18/20 17:46:25	BD = ACE	-0.007				
06/18/20 17:46:25	CE = ABD	0.016				
06/18/20 17:46:25	CD = ABE	-0.037				
06/18/20 17:46:25	BE = ACD	0.014				
06/18/20 17:46:25	AE = BCD	-0.013				
06/18/20 17:46:25	E = ABCD	-0.022	0.002	1	1.346	4.96
06/18/20 17:46:25	Error		0.014	10		
06/18/20 17:46:25	Total		0.518	15		
06/18/20 17:46:25	Grand Mean	0.353				

06/18/20 17:46:25 Experiment ended.

Приложение Б The run execution procedure in GPSS

```
*****
*
*          BK_DisExp
*   Fractional Factorial Screening Experiment
*
*****
BK_DisExp_Results MATRIX ,2,2,2,2,2
INITIAL BK_DisExp_Results,UNSPECIFIED
BK_DisExp_NextRunNumber EQU      0

EXPERIMENT BK_DisExp() BEGIN

    /* Run 1 */
    L_ = 1;
    T5_ = 3;
    T6_ = 3;
    T7_ = 4;
    Koef = 0.5;
    IF (StringCompare(DataType(BK_DisExp_Results[1,1,1,1,1]),"UNSPECIFIED")'E'0)
    THEN BEGIN
        /* Set the Run Number Variable at the beginning. */

        BK_DisExp_NextRunNumber = 1;
        /* Log the Run and Execute the Simulation */
        BK_DisExp_GetResult();
        BK_DisExp_Results[1,1,1,1,1] = N$Term1/N$Met0;
    END;

    /* Run 2 */
    L_ = 1;
    T5_ = 3;
    T6_ = 3;
    T7_ = 18;
    Koef = 2;
    IF (StringCompare(DataType(BK_DisExp_Results[1,1,1,2,2]),"UNSPECIFIED")'E'0)
    THEN BEGIN
        /* Log the Run and Execute the Simulation */
```

```

    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,1,1,2,2] = N$Term1/N$Met0;
END;

/* Run 3 */
L_ = 1;
T5_ = 3;
T6_ = 20;
T7_ = 4;
Koef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[1,1,2,1,2]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,1,2,1,2] = N$Term1/N$Met0;
END;

/* Run 4 */
L_ = 1;
T5_ = 3;
T6_ = 20;
T7_ = 18;
Koef = 0.5;
IF (StringCompare(DataType(BK_DisExp_Results[1,1,2,2,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,1,2,2,1] = N$Term1/N$Met0;
END;

/* Run 5 */
L_ = 1;
T5_ = 15;
T6_ = 3;
T7_ = 4;
Koef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[1,2,1,1,2]),"UNSPECIFIED")'E'0)
THEN BEGIN

```

```

    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,2,1,1,2] = N$Term1/N$Met0;
END;

/* Run 6 */
L_ = 1;
T5_ = 15;
T6_ = 3;
T7_ = 18;
Kcoef = 0.5;
IF (StringCompare(DataType(BK_DisExp_Results[1,2,1,2,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,2,1,2,1] = N$Term1/N$Met0;
END;

/* Run 7 */
L_ = 1;
T5_ = 15;
T6_ = 20;
T7_ = 4;
Kcoef = 0.5;
IF (StringCompare(DataType(BK_DisExp_Results[1,2,2,1,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,2,2,1,1] = N$Term1/N$Met0;
END;

/* Run 8 */
L_ = 1;
T5_ = 15;
T6_ = 20;
T7_ = 18;
Kcoef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[1,2,2,2,2]),"UNSPECIFIED")'E'0)

```

```

THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[1,2,2,2,2] = N$Term1/N$Met0;
END;

/* Run 9 */
L_ = 5;
T5_ = 3;
T6_ = 3;
T7_ = 4;
Koef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[2,1,1,1,2]), "UNSPECIFIED") 'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,1,1,1,2] = N$Term1/N$Met0;
END;

/* Run 10 */
L_ = 5;
T5_ = 3;
T6_ = 3;
T7_ = 18;
Koef = 0.5;
IF (StringCompare(DataType(BK_DisExp_Results[2,1,1,2,1]), "UNSPECIFIED") 'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,1,1,2,1] = N$Term1/N$Met0;
END;

/* Run 11 */
L_ = 5;
T5_ = 3;
T6_ = 20;
T7_ = 4;
Koef = 0.5;

```

```

IF (StringCompare(DataType(BK_DisExp_Results[2,1,2,1,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,1,2,1,1] = N$Term1/N$Met0;
END;

/* Run 12 */
L_ = 5;
T5_ = 3;
T6_ = 20;
T7_ = 18;
Koef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[2,1,2,2,2]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,1,2,2,2] = N$Term1/N$Met0;
END;

/* Run 13 */
L_ = 5;
T5_ = 15;
T6_ = 3;
T7_ = 4;
Koef = 0.5;
IF (StringCompare(DataType(BK_DisExp_Results[2,2,1,1,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,2,1,1,1] = N$Term1/N$Met0;
END;

/* Run 14 */
L_ = 5;
T5_ = 15;
T6_ = 3;
T7_ = 18;
Koef = 0.5;

```

```

Koef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[2,2,1,2,2]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,2,1,2,2] = N$Term1/N$Met0;
END;

/* Run 15 */
L_ = 5;
T5_ = 15;
T6_ = 20;
T7_ = 4;
Koef = 2;
IF (StringCompare(DataType(BK_DisExp_Results[2,2,2,1,2]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,2,2,1,2] = N$Term1/N$Met0;
END;

/* Run 16 */
L_ = 5;
T5_ = 15;
T6_ = 20;
T7_ = 18;
Koef = 0.5;
IF (StringCompare(DataType(BK_DisExp_Results[2,2,2,2,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
    /* Log the Run and Execute the Simulation */
    BK_DisExp_GetResult();
    BK_DisExp_Results[2,2,2,2,1] = N$Term1/N$Met0;
END;

/* Aliased Effects in Fractional Factorial Experiment */
SE_Effects(BK_DisExp_Results,"I=ABCDE");

END;

```

```

*           The Run Execution Procedure           *
*****

PROCEDURE BK_DisExp_GetResult() BEGIN

  /*   Run Simulation and Log Results.   */
  /*   Treatments have already been set for this run.   */

  TEMPORARY CurrentYield,ShowString,CommandString;

  /*   Run Procedure Call   */

  BK_DisExpRun(BK_DisExp_NextRunNumber);
  CurrentYield = N$Term1/N$Met0;
  ShowString = PolyCatenate("Run ",String(BK_DisExp_NextRunNumber),". ", "" );

  ShowString = PolyCatenate(ShowString," Yield=",String(CurrentYield),". ");
  ShowString = PolyCatenate(ShowString," L_=",String(L_), ";");
  ShowString = PolyCatenate(ShowString," T5_=",String(T5_), ";");
  ShowString = PolyCatenate(ShowString," T6_=",String(T6_), ";");
  ShowString = PolyCatenate(ShowString," T7_=",String(T7_), ";");
  ShowString = PolyCatenate(ShowString," Koef=",String(Koef), ";");
  CommandString = PolyCatenate("SHOW """,ShowString,"""", "" );
  DoCommand(CommandString);
  BK_DisExp_NextRunNumber = BK_DisExp_NextRunNumber + 1;
  RETURN CurrentYield;

END;

*****
*           Run Procedure           *
*****
PROCEDURE BK_DisExpRun(Run_Number) BEGIN
  DoCommand("CLEAR OFF");      /* Must use OFF to preserve results. */

  /* EXPAND THIS RMULT IF YOU HAVE MORE RNGs. */
  /* All Random Number Streams must have new seeds. */
  TEMPORARY CommandString;
  /* Evaluate before passing to DoCommand. */

  CommandString = Catenate("RMULT ",Run_Number#111);
  /* DoCommand compiles the string in Global Context. */
  DoCommand(CommandString);

  /* SET UP YOUR OWN RUN CONDITIONS. */
  DoCommand("START 150,NP");    /* Get past the Startup Period. */
  DoCommand("RESET");          /* Begin the Measurement Period. */
  DoCommand("START 100,NP");   /* Run the Simulation. */
END;
*****

```