

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	15
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	26
2.3 Моделирование системы массового обслуживания для анализа производительности IoT-приложений	30
2.4 Дисперсионный анализ в GPSS	37
3 Эксперименты с имитационными моделями	46
3.1 Основные понятия теории планирования экспериментов	46
3.2 Стратегическое планирование эксперимента	48
3.3 Tактическое планирование эксперимента	50
3.4 Дисперсионный анализ (отсеивающий эксперимент) на примере системы GPSS World	52
Заключение	63
Список литературы	64
Приложение А Результаты моделирования в среде GPSS	66
Приложение Б The run execution procedure in GPSS	69
Приложение В Справка Антиплагиата	74

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. Методология сетевого моделирования часто используется для верификации аналитических моделей, обобщения результатов измерений, оценки эффективности разрабатываемых новых протоколов, а также для сравнения 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

Телекоммуникационная отрасль в настоящее время переживает период динамичного роста. Наряду с традиционными видами телекоммуникационных услуг интенсивно развиваются новые: мобильная связь (включая передачу данных); электронный бизнес и торговля; интернет; мультимедийные технологии и др. В то же время каналы и линии связи, служащие для передачи сообщений в виде электрических и радиосигналов, все больше "сливаются" с компьютерной техникой, что приводит к развитию единых, разветвленных и достаточно сложных информационно-коммуникационных сетей как в нашей стране, так и за рубежом. Задача эффективного управления этими сетями становится все более актуальной и приводит к необходимости применения на практике таких разделов современной науки, как теория сложных систем, управление и организации; приведены результаты работы в области искусственного интеллекта.

Классические методы исследования связаны с экспериментами, требующими больших ресурсов, сил и средств, поскольку они "пассивны" - основаны на чередующемся изменении отдельных независимых переменных в условиях, когда остальные имеют тенденцию оставаться постоянными.

Использование планирования эксперимента делает поведение экспериментатора более целенаправленным и организованным, повышает производительность труда и достоверность полученных результатов. Важным преимуществом является его универсальность и пригодность в значительном большинстве областей исследований.

Эксперимент-целенаправленное воздействие на объект исследования с целью получения достоверной информации. Большинство научных исследований было проведено с помощью экспериментов. Она осуществляется на производстве, в лабораториях, на опытных полях и площадках, в клиниках и т.д. Эксперимент может быть физическим, психологическим или основанным на модели. Она может быть выполнена непосредственно на объекте или на его модели. Модель обычно отличается от объекта масштабом, а иногда и характером. Главным требованием к модели является достаточно точное описание объекта.

В последнее время наряду с физическими моделями все чаще используются абстрактные математические модели. Кроме того, планирование эксперимента непосредственно связано с разработкой и исследованием математической модели объекта исследования [1].

Планирование эксперимента - это процедура выбора количества и условий проведения экспериментов, необходимых и достаточных для решения поставленной задачи с требуемой точностью. Здесь значительную роль играет успокоение: стремление свести к минимуму общее количество попыток; одновременное варьирование всех переменных, определяющих

процесс, по специальным законам-алгоритмам; использование математической аппаратуры, формализующей многие действия экспериментатора; выбор четкой методологии, позволяющей принимать аргументированные решения после каждой серии экспериментов. Задачи, для решения которых может быть применено экспериментальное планирование, чрезвычайно разнообразны. К ним относятся следующие задачи: поиск оптимальных условий, построение интерполяционных формул, выбор существенных факторов, оценка и уточнение констант теоретических моделей, выбор наиболее подходящих из большего числа гипотез о механизме событий, исследование диаграмм состав-свойства и т. д.

Чтобы проводить исследования, нужно уметь влиять на поведение "черного ящика". Все методы такого воздействия мы обозначаем буквой (x) и называем факторами. Их еще называют входами в "черный ящик".

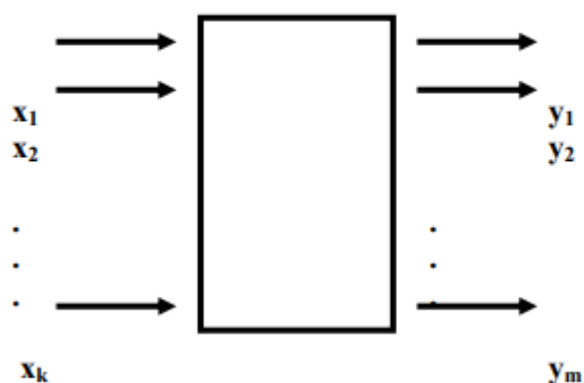


Рисунок 1.1 – Модель объекта исследования

При решении задачи будем применять математические модели исследования. Под математической моделью мы имеем ввиду уравнение, соединяющее параметр оптимизации с факторами. Это уравнение в общем случае имеет вид:

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

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

Каждый фактор может принимать в опыте одно из нескольких значений. Эти значения называются уровнями. Для упрощения моделирования «черного ящика» и эксперимента фактор должен обладать определенным числом дискретных уровней. Стационарный набор уровней факторов выявляет одно из вероятных значений «черного ящика». В то же время это является условием проведения одного из вероятных исследований. Если рассмотреть все допустимые наборы состояний, то получается множество многочисленных

состояний «черного ящика». При этом это будет являться числом возможных различных исследований [2].

Число допустимых опытов определяется по формуле:

$$N=p^k,$$

где N – число опытов;

p – число уровней;

k – число факторов.

Проведение опытов используя планирование эксперимента запрашивает выполнение некоторых условий. Главными из них являются условия воспроизводимости результатов опытов и подконтрольность эксперимента. Если провести еще раз некоторые опыты через неравные промежутки времени и сравнить результаты, в данном случае – значения параметра оптимизации, то разброс их значений определяет воспроизводимость результатов. Если он не превышает некоторых заданных значений, то объект удовлетворяет требованию воспроизводимости результатов. В диссертационной работе я рассматриваю только тек объекты, где это условие выполняется.

Планирование эксперимента подразумевает активное вмешательство в процесс и возможность выбора в каждом исследовании тех уровней факторов, которые интересуют экспериментатора. По этой причине такой эксперимент называют активным, а объект, на котором возможно планировать активный эксперимент – управляемым. В реальной мире нет абсолютно управляемых объектов, т.к. на них действуют управляемые, так и неуправляемые разные факторы. Неуправляемые факторы оказывают влияния на воспроизводимость опыта и является причиной ее нарушения либо сбоя. При нарушении воспроизводимости эксперимента приходится переходить к другим методам исследования.

1.2 Моделирование телекоммуникационных сетей и систем

Моделирование представляет собой метод научного познания, при использовании которого исследуемый объект заменяется более простым объектом, называемым моделью. Основными видами процесса моделирования считаются математическое и физическое моделирование. При физическом (натурном) моделировании исследуемая система заменяется другой материальной системой, которая воспроизводит свойства изучаемой системы, но сохраняет при этом их физическую природу. Примером этого вида можно привести пилотную сеть, с помощью которой изучается принципиальная возможность построения сети на основе тех или иных компьютеров, коммуникационных средств, операционных систем и приложений.

Возможности физического моделирования значительно ограничены. Физическое моделирование позволяет решать отдельные задачи при задании низкого количества исследуемых параметров системы. И в правду, при физическом моделировании вычислительной сети практически невозможно

проверить ее поведения для вариантов с использованием различных типов коммуникационных устройств - маршрутизаторов, коммутаторов и т.д. Исследование на практике около десятки разных типов маршрутизаторов связано не только с огромными усилиями и временными затратами, но и с большими материальными затратами.

Но даже и в тех случаях, когда при оптимизации сети меняются не типы устройств и операционных систем, но только их параметры, проведение экспериментов в реальном масштабе времени для огромного количества параметров практически невозможно за короткое время. Даже простое изменение максимального размера пакета в каком-либо протоколе обязывает переконфигурирования операционной системы в сотнях компьютеров сети, что в последствии администратора сети проведет очень большую работу.

Следовательно, при процессе максимизации улучшения сетей во многих случаях наилучшим оказывается использование математического моделирования. Математическая модель является совокупность соотношений (формул, уравнений, неравенств, логических условий), определяющих процесс изменения состояния системы в зависимости от ее параметров, входных сигналов, начальных условий и времени.

Имитационные модели - это особый класс математических моделей. Такие модели представляют собой компьютерную программу, которая шаг за шагом воспроизводит события, происходящие в реальной системе. Для вычислительных сетей их имитационные модели воспроизводят процессы генерации сообщений приложениями, разбиения сообщений на пакеты и фреймы определенных протоколов, задержки, связанные с обработкой сообщений, пакетов и фреймов внутри операционной системы, процесс получения компьютером доступа к общей сетевой среде, процесс обработки входящих пакетов маршрутизатором и т. д. При моделировании сети вам не нужно приобретать дорогостоящее оборудование - ее работа моделируется программами, которые точно воспроизводят все основные характеристики и параметры такого оборудования.

Преимуществом имитационных моделей является возможность замены процесса изменения событий в исследуемой системе в режиме реального времени ускоренным процессом изменения событий со скоростью выполнения программы. В результате можно воспроизвести работу сети в течение нескольких дней за несколько минут, что дает возможность оценить работу сети в широком диапазоне переменных параметров. Результатом работы имитационной модели являются статистические данные, собранные в ходе мониторинга событий о наиболее важных характеристиках сети: времени реакции, коэффициентах использования каналов и узлов, вероятности потери пакетов и др.

Существуют специальные языки моделирования, которые облегчают процесс создания программной модели по сравнению с использованием универсальных языков программирования. Примеры язык моделирования Симула сим, ГПСС, и имитированной дистилляции.

Существуют также имитационные системы, которые ориентированы на узкий класс изучаемых систем и позволяют строить модели без программирования.

Анализаторы протоколов необходимы для исследования реальных сетей, но они не позволяют получить количественные оценки производительности для несуществующих сетей, которые находятся в стадии проектирования. В этих случаях проектировщики могут использовать инструменты моделирования для разработки моделей, воссоздающих информационные процессы, происходящие в сетях.

Используя моделирование при проектировании или реинжиниринге локальной или глобальной сети, можно реализовать следующее [2,3]:

- оценить пропускную способность сети и ее компонентов;
- выявление "узких мест" в структуре компьютерной системы;
- сравнение различных вариантов организации компьютерной системы;
- реализация долгосрочного прогноза развития компьютерной системы;
- прогнозирование будущих требований к пропускной способности сети с использованием прогнозных данных;
- оценить необходимое количество и производительность серверов в сети;
- сравнить различные варианты модернизации компьютерной системы;
- оценить влияние на компьютерную систему обновлений программного обеспечения, мощности рабочих станций или серверов, изменения сетевых протоколов и многое другое.

Исследование параметров вычислительной системы с различными характеристиками отдельных компонентов позволяет выбрать сетевое и вычислительное оборудование исходя из производительности, качества обслуживания, надежности и стоимости.

Применение метода ИМ для решения задач оптимизации управления стимулируется также тем, что критериями оптимальности сложных систем часто выступают функционалы, трудно поддающиеся определению не только аналитическими, но и численными методами. Использование моделей, имитирующих процесс управления, не только упрощает постановку задачи (в том числе выбор параметров оптимизации и набор ограничений), но и дает возможность провести указанную оптимизацию - путем обработки множества реализаций модели при изменении исходных данных и факторов.

Несмотря на отличные результаты математического моделирования, многие реальные ситуации не могут быть адекватно представлены с помощью соответствующих математических моделей. В некоторых случаях этому препятствует определенная "жесткость" математики как языка описания и представления событий и явлений. Кроме того, даже если удастся формализовать рассматриваемую жизненную ситуацию путем построения математической модели, полученная из нее задача оптимизации может оказаться слишком сложной для современных алгоритмов решения задач этого класса.

Альтернативой математическому моделированию сложных систем может быть имитационное моделирование (им). Этот тип моделирования часто является лучшим (если не единственным) способом изучения реальных систем.

Термин "моделирование" означает, что мы имеем дело с такими моделями, которые не могут быть использованы для расчета или прогнозирования поведения системы заранее, а для прогнозирования поведения системы требуется вычислительный эксперимент (моделирование) на математической модели с заданными исходными данными.

Имитационные модели обычно классифицируются по следующим наиболее распространенным признакам:

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

Классификация имитационных моделей схематично показана на рисунке 2.

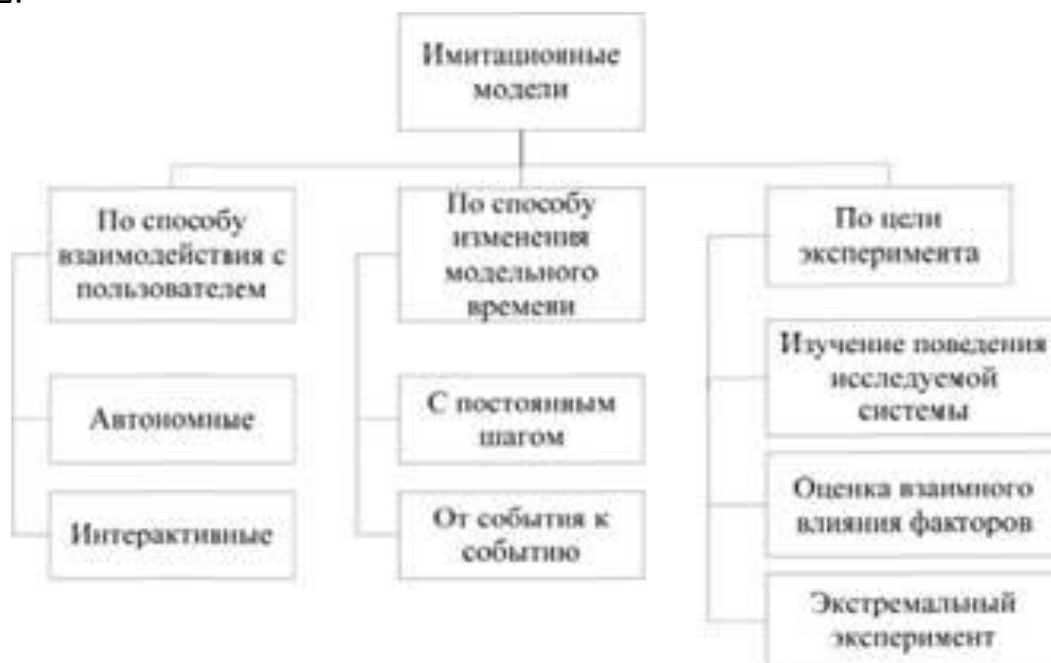


Рисунок 1.2 - Классификация имитационных моделей

Разница между математической и имитационной моделями заключается в том, что в последней вместо явного математического описания взаимосвязи между входными и выходными переменными реальная система разбивается на ряд достаточно мелких (в функциональном плане) элементов или модулей. Тогда поведение исходной системы имитируется как поведение совокупности этих элементов, которые определенным образом связаны (путем установления соответствующих отношений между ними) в единое целое [4]. Вычислительная реализация такой модели начинается с входного элемента, а затем проходит через все элементы до тех пор, пока не будет достигнут выходной элемент модели.

По способу взаимодействия с пользователем имитационные модели могут быть автономными и интерактивными [4]. Автономные модели не требуют вмешательства исследователя после определения режима моделирования и установки исходных данных. взаимодействие пользователя с такими моделями ограничивается только вводом исходной информации и управлением началом и концом работы модели. Интерактивные модели обеспечивают диалог с пользователем в том или ином режиме в соответствии со сценарием моделирования, позволяя пользователю приостановить сеанс моделирования/изменить значения параметров модели, скорректировать список зарегистрированных данных и так далее.

Процесс построения имитационных моделей представляет собой последовательное выполнение этапов моделирования. Вышеперечисленные этапы имитационного исследования редко выполняются в строго определенной последовательности, начиная с определения задачи и заканчивая документацией. В ходе имитационного исследования возможны сбои в работе модели, ошибочные допущения, от которых впоследствии придется отказаться, т. е. на каждом этапе можно вернуться к предыдущим этапам. Этот итерационный процесс позволит вам получить модель, которая позволит Вам принимать решения.

Вычислительные аспекты имитационных моделей обычно относительно просты, но обычно очень трудоемки. Поэтому реализация таких моделей предполагает использование компьютерных технологий.

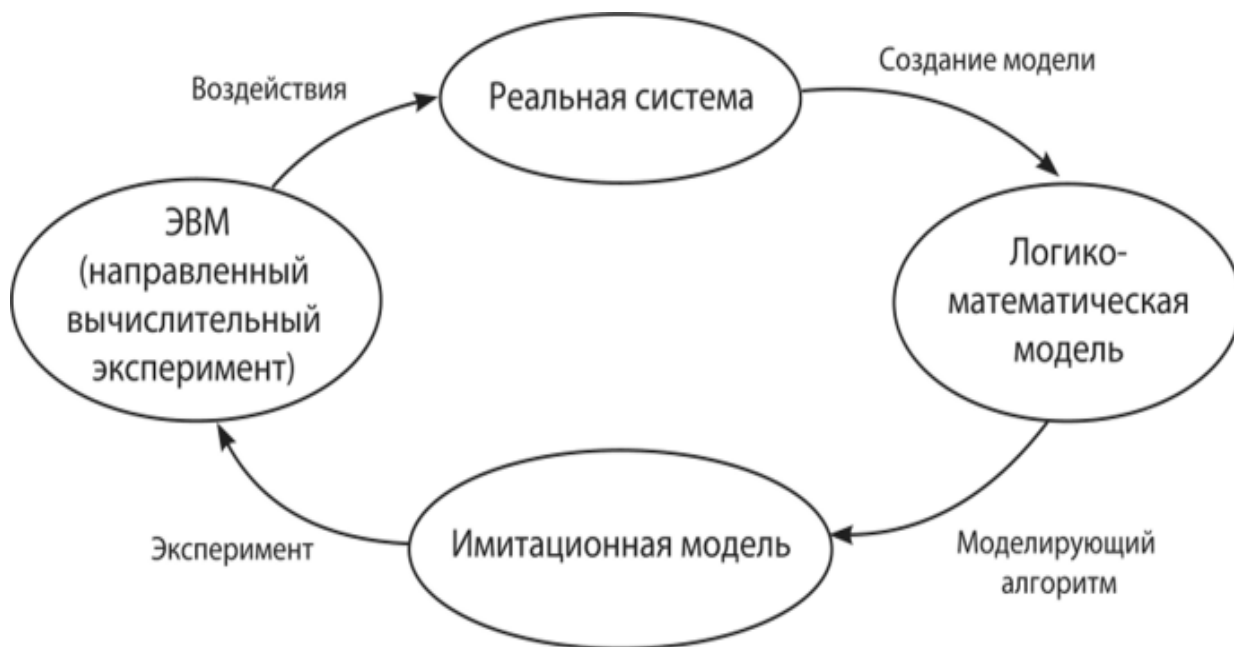


Рисунок 1.3 – Процесс имитационного моделирования

Таким образом, моделирование является мощным инструментом для изучения поведения реальных систем. Методы имитационного моделирования позволяют собрать необходимую информацию о поведении системы путем

создания ее компьютеризированной модели. Эта информация затем используется для проектирования системы. Главное преимущество чата:

- способность описывать поведение компонентов (элементов) процессов или систем на высоком уровне детализации;

- отсутствие ограничений между параметрами им и состоянием внешней среды;

- возможность изучения динамики взаимодействия компонентов во времени и пространстве параметров системы [6].

Эти преимущества обеспечивают широкое применение метода моделирования. Мы рекомендуем использовать моделирование в следующих случаях:

а) если нет полной постановки задачи исследования и процесс обучения объекта моделирования идет полным ходом. Имитационная модель служит средством изучения данного явления.

б) если имеются аналитические методы, но математические процессы сложны и требуют больших затрат времени, то моделирование обеспечивает более простой способ решения задачи.

в) когда, помимо оценки влияния параметров (переменных) процесса или системы, желательно наблюдать поведение компонентов (элементов) процесса или системы в течение определенного периода времени.

г) когда моделирование является единственным способом изучения сложной системы из-за невозможности наблюдать явления в реальных условиях (термоядерные реакции, освоение космоса).

д) когда необходимо управлять потоком процессов или поведением систем путем замедления или ускорения явлений во время моделирования.

д) при подготовке специалистов для нового оборудования, когда имитационные модели дают возможность приобрести навыки работы с новым оборудованием.

д) когда изучаются новые ситуации в поведении реальных процессов и систем. В этом случае имитационное моделирование используется для проверки новых стратегий и правил проведения полевых экспериментов.

ж) когда последовательность событий в проектируемых процессах и системах имеет особое значение и модель используется для прогнозирования узких мест в их функционировании [5].

Однако они, наряду с достоинствами, имеют и недостатки:

- разработка хорошей модели часто обходится дороже, чем создание аналитической модели, и требует много времени;

- может оказаться, что им неточен (что часто бывает), и исследователь не в состоянии измерить степень этой неточности;

- часто исследователи обращаются к ним, не представляя трудностей, с которыми они столкнутся, и допускают ряд методологических ошибок [4]. И тем не менее, ИМ является одним из наиболее широко используемых методов при решении задач синтеза и анализа сложных процессов и систем.

В настоящее время имитационное моделирование является общепризнанным методом исследования сложных динамических систем. Оно широко применяется в различных областях науки, бизнеса и производства.

1.3 Имитационное моделирование в управлении сложными телекоммуникационными системами

Изменения в телекоммуникационных технологиях и интенсивное развитие компьютерных технологий привели к развитию телекоммуникационных систем (ТКС). Стремительное развитие вычислительных возможностей, необходимость обеспечения мобильной связи, электронного бизнеса и торговли, мультимедийных технологий и др. это привело к появлению географически рассредоточенных и сложных информационно-коммуникационных сетей.

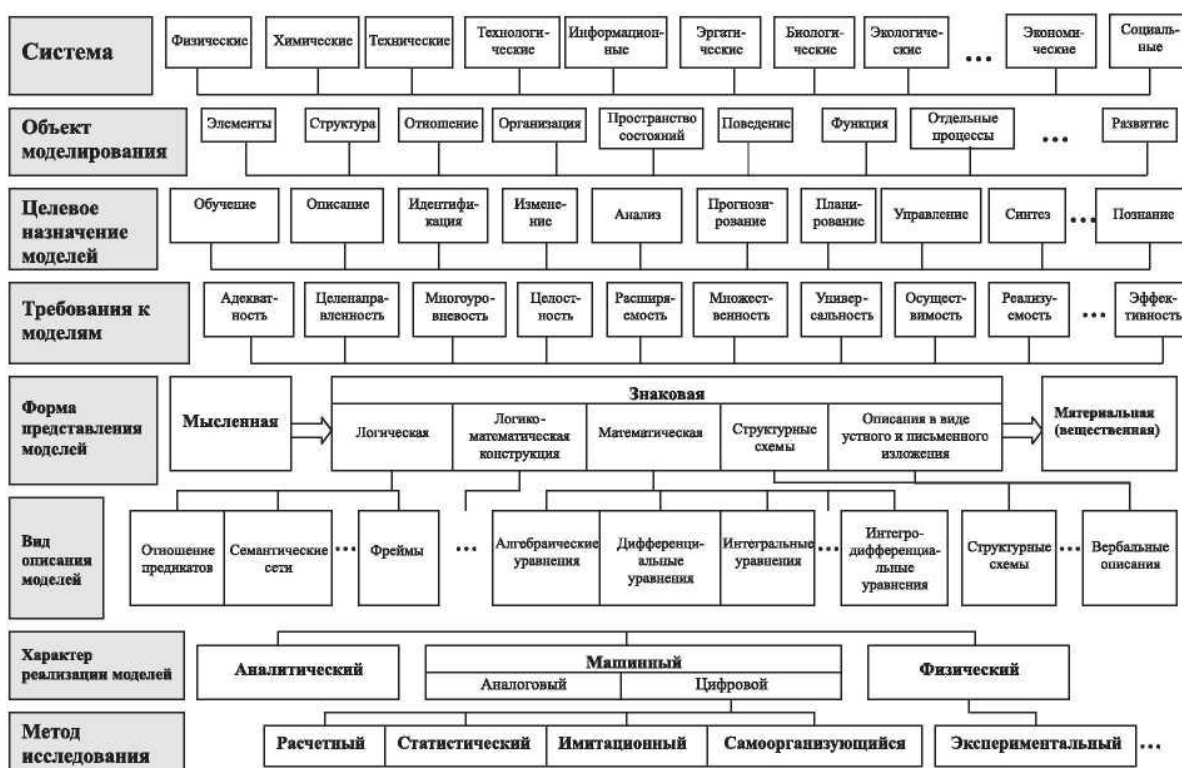


Рисунок 1.4 – Общая процедурно-технологическая схема построения и исследования сложных систем

Сложная система - это совокупность отдельных частей, обладающих своими свойствами и законами функционирования. Изучение такой системы предполагает решение задач, связанных как со свойствами отдельных элементов, так и с функционированием этих элементов в рамках единой системы. Наиболее важную роль для исследования играют свойства элементов системы, которые определяют отношение к другим элементам или ко всей системе в целом. В качестве примера сложной телекоммуникационной системы можно рассмотреть городские телефонные сети [7]. К таким сетям

относятся информационные сетевые технологии (коммутационные блоки телефонных станций, централизованные автоматизированные системы обслуживания, биллинговые системы и др.), телекоммуникационное оборудование (абонентское терминальное оборудование, оборудование автоматических телефонных станций, модемов, маршрутизаторов, кабельных систем) и сопутствующие услуги (обеспечение доступа к телефонным сетям и Интернету, предоставление услуг мобильной телефонии). Таким образом, процесс эксплуатации телефонной сети является наглядным примером функционирования современной сложной системы. Управление играет важную роль в функционировании сложных систем. Управление-это развитие целенаправленного функционирования сложной системы посредством определенных действий, формируемых различными средствами. Замкнутые контуры управления подразумевают обратную связь, то есть, когда фактическое значение параметра отклоняется от заданного значения, генерируется соответствующий управляющий сигнал. Средствами формирования таких управляющих воздействий для реальных ТКС являются различные вычислительные устройства, обеспечивающие обработку фактических значений контролируемых параметров и планирование соответствующих корректирующих действий. Долгосрочное планирование и оперативное управление сложными системами должно осуществляться в рамках одного алгоритма управления. Алгоритм управления может быть централизованным, распределенным или децентрализованным. Качественная децентрализация алгоритма управления сокращает объем обрабатываемой информации и время реагирования на корректирующие действия. В противном случае качество управления может значительно снизиться. На процессы функционирования всех сложных систем в реальных условиях оказывают влияние многочисленные случайные факторы. К таким факторам относятся влияние внешней среды и внутрисистемные искажения, связанные с физическими законами передачи информации. Внешние и внутренние факторы приводят к различным отклонениям от заданных значений. В ТКС такими факторами могут быть неожиданные пиковые нагрузки, вызванные одновременными запросами абонентов на телефонные разговоры, внезапные труднопрогнозируемые перераспределения трафика, неожиданное включение или выключение потребителей электроэнергии и др. [1].

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, определяющие 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.

Функциональные группы, непосредственно зависящие от показателей эффективности, являются наиболее заметными с точки зрения представления результатов исследования, поскольку важны свойства системы, влияющие на ее эффективность. Каждая сложная система образует набор из большого количества элементов, которые могут прийти в негодность в процессе эксплуатации, что влечет за собой замену или ремонт. Оценка

отказоустойчивости элементов определяет параметры надежности системы, т. е. показатели надежности системы.

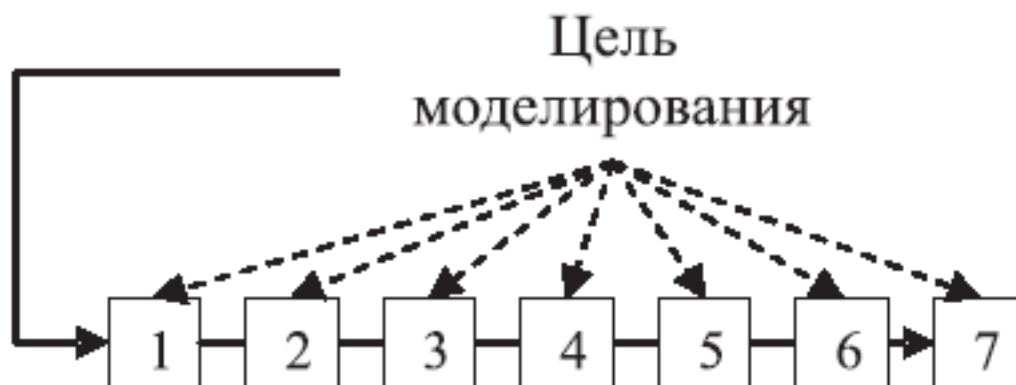


Рисунок 1.5 – Технологическая схема имитационного моделирования

- 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).

Многие системы, которые моделируются с использованием методов моделирования, предполагают использование метода Монте-Карло в своей основе. Этот метод позволяет рассмотреть большое количество возможных реакций системы на внешние воздействия и спрогнозировать вероятность их возникновения [6]. Смоделированные сложные ТКС относятся к системам массового обслуживания, для которых разработаны многочисленные алгоритмы [7]. Алгоритм моделирования должен не только адекватно и полно отражать процесс функционирования КФР, но и в то же время обладать приемлемой вычислительной сложностью для его реализации на ЭВМ и использования. Таким образом для алгоритма моделирования выдвигаются следующие требования:

- моделируется одновременная и независимая работа любого количества элементов;
- малые затраты ресурсов ЭВМ на реализацию;
- простота реализации алгоритма, возможность декомпозиции на независимые модули, максимальная стандартизация. Большое количество СМО требует создания различных моделирующих алгоритмов.

Классифицировать эти алгоритмы можно только условно, так как при моделировании большой системы можно использовать части различных алгоритмов. Все используемые в настоящее время алгоритмы моделирования могут быть использованы для СМО. Поэтому модель сложной телекоммуникационной системы, рассматриваемой как система массового

обслуживания, также может быть описана алгоритмом моделирования. Выбор алгоритма варьируется в зависимости от сложности реализации, стоимости машинных ресурсов и других параметров. Представляя ТКС как сложную производственную систему, можно сделать вывод, что в ней функционируют следующие агрегатные группы оборудования: непрерывный агрегат, дискретный агрегат, дискретно-непрерывный агрегат [7; 8].

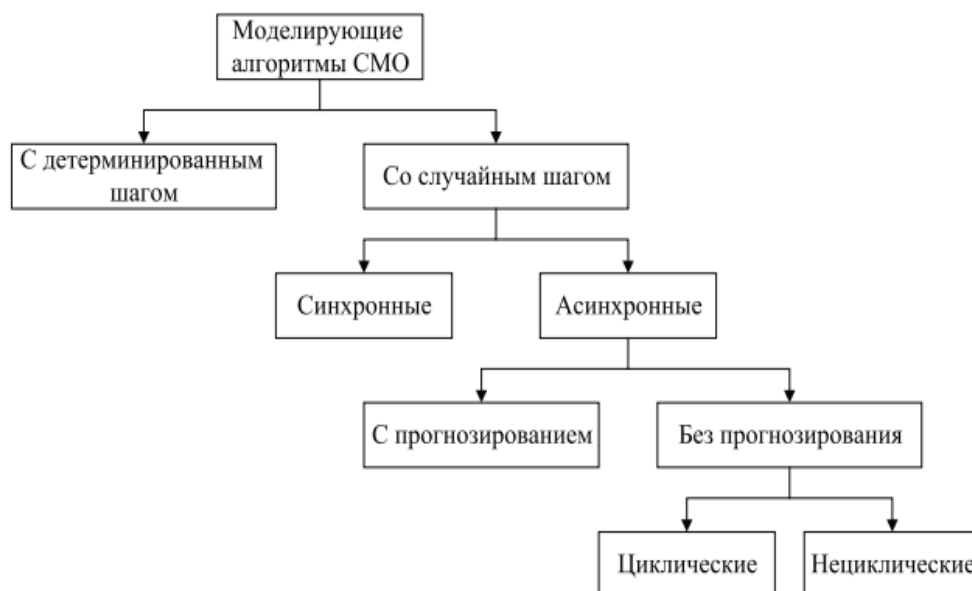


Рисунок 1.6 - Классификация моделирующих алгоритмов СМО

Входным управляющим сигналом и выходным сигналом для ТКС, как сложной производственной системы, является информация. Анализ функционирования показывает, что ТКС представляет собой дискретно-непрерывную совокупность, поскольку совокупность случайных событий приводит к изменению ее поведения [9]. Среди дискретно-непрерывных агрегатов можно выделить подклассы, для которых можно получить простое решение. Типичные примеры дискретно-непрерывных агрегатов включают математическую модель системы массового обслуживания. Таким образом, моделирование процессов ТКС может быть основано на соответствующих моделях и алгоритмах СМО. Однако исследование зависимости условий работы ТКС от времени настолько сложно, что провести его без использования компьютера невозможно. Нам нужна модель, учитывающая случайные обстоятельства функционирования СМО. Единственной практической ценностью является анализ работы СМО как функции времени. Решение этой проблемы возможно только при использовании современных электронных вычислительных машин. Во время работы СМО происходят различные случайные события, которые носят статистический характер. Таким образом, для моделирования ТКС необходимо использовать методы

имитационного и статистического моделирования, в частности метод Монте-Карло.

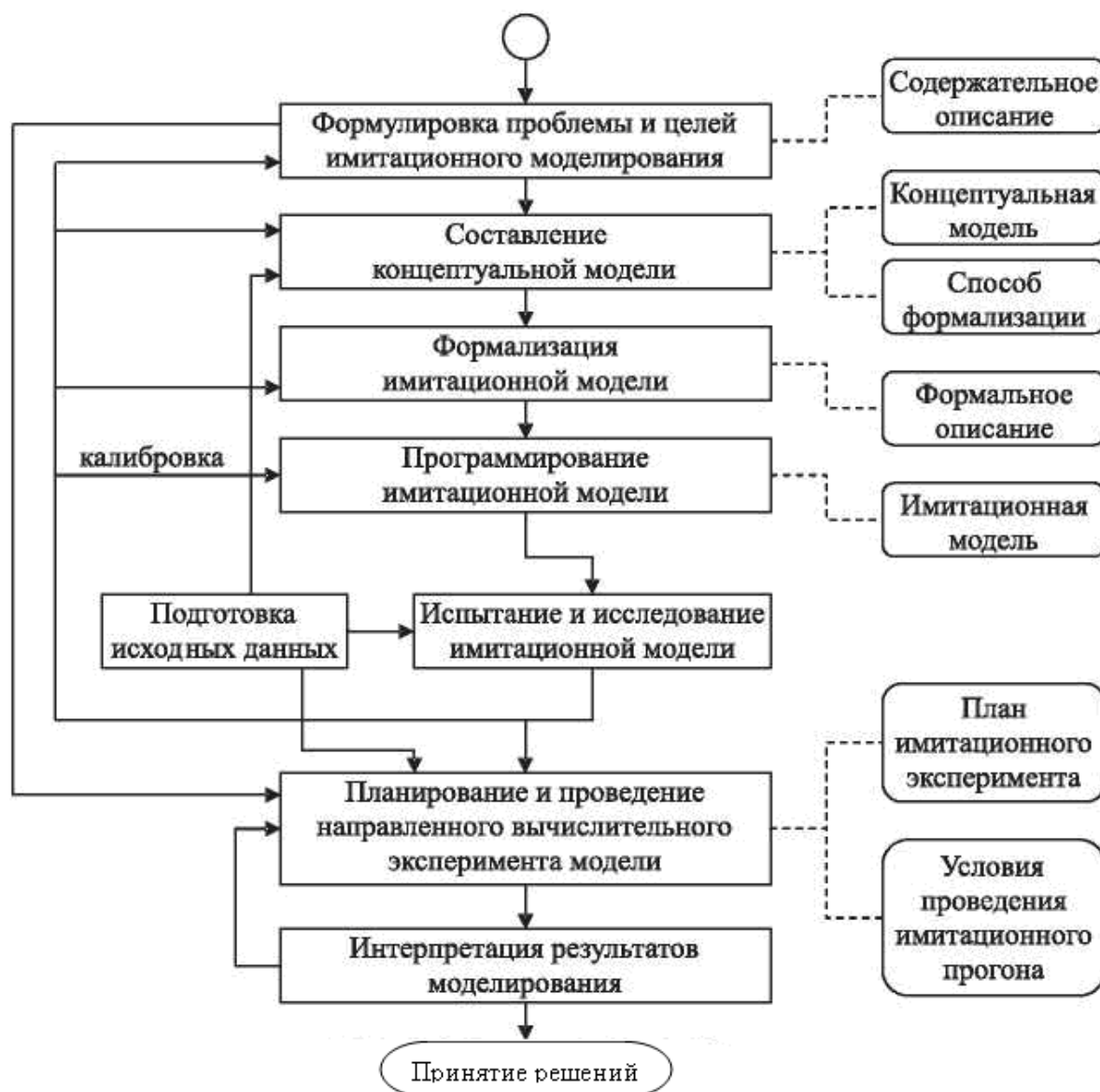


Рисунок 1.7 - Технологические этапы имитационного моделирования

В качестве критериев оценки эффективности ТКС могут использоваться такие интегральные показатели, как стабильность, надежность, производительность и помехозащищенность. С математической точки зрения, на множестве процессов функционирования ТКС показатель эффективности задается функционалом. Увеличение доли математических методов в практике моделирования и значительное повышение производительности современных вычислительных машин привели к тому, что формализация и исследование сложных процессов (технических, производственных, экономических), протекающих в ТКС, стали реальной задачей. Компьютерное моделирование стало наиболее подходящим методом для анализа функционирования и управления такими системами. Таким образом,

разработанная система моделирования представляет собой комплекс модели, имитирующей исследуемый процесс на компьютере, и соответствующего программного обеспечения. Таким образом, эксперимент с реальным процессом заменяется реализацией математической модели этого процесса на компьютере.

Подводя итог, можно сделать вывод, что она применима для решения задач управления сложными ТКС. Вычислительная мощность современных вычислительных машин позволяет эффективно использовать методы и средства теории сложных систем для ИМ ТКС. В частности, целесообразно использовать метод Монте-Карло, являющийся основой группы эффективных методов исследования сложных процессов и систем, для моделирования условий работы ТКС под влиянием случайных факторов. При реализации метода Монте-Карло на компьютере результаты им представляют собой числовые последовательности выходных данных, которые должны быть статистически обработаны и впоследствии интерпретированы. Кроме того, математический аппарат СМО может быть использован для моделирования современных ТКС. Хорошо известные алгоритмы, моделирующие работу CFR, отличаются сложностью реализации, стоимостью машинных ресурсов и другими параметрами. Поэтому в рамках выбранной методологии необходимо в каждом конкретном случае сделать обоснованный выбор алгоритма моделирования, от которого зависит точность результатов моделирования сложных ТКС и, соответственно, стратегия управления.

2 Моделирование систем телекоммуникаций как систем массового обслуживания

2.1 Основные понятия систем массового обслуживания

Система массового обслуживания — математический (абстрактный) объект, который имеет нижеперечисленные элементы(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.

Заявка (запрос, требование, вызов, клиент, сообщение, пакет) — объект, поступающий в СМО и требующий обслуживания в приборе. Совокупность последовательных заявок, распределенных во времени, образуют входной поток заявок.

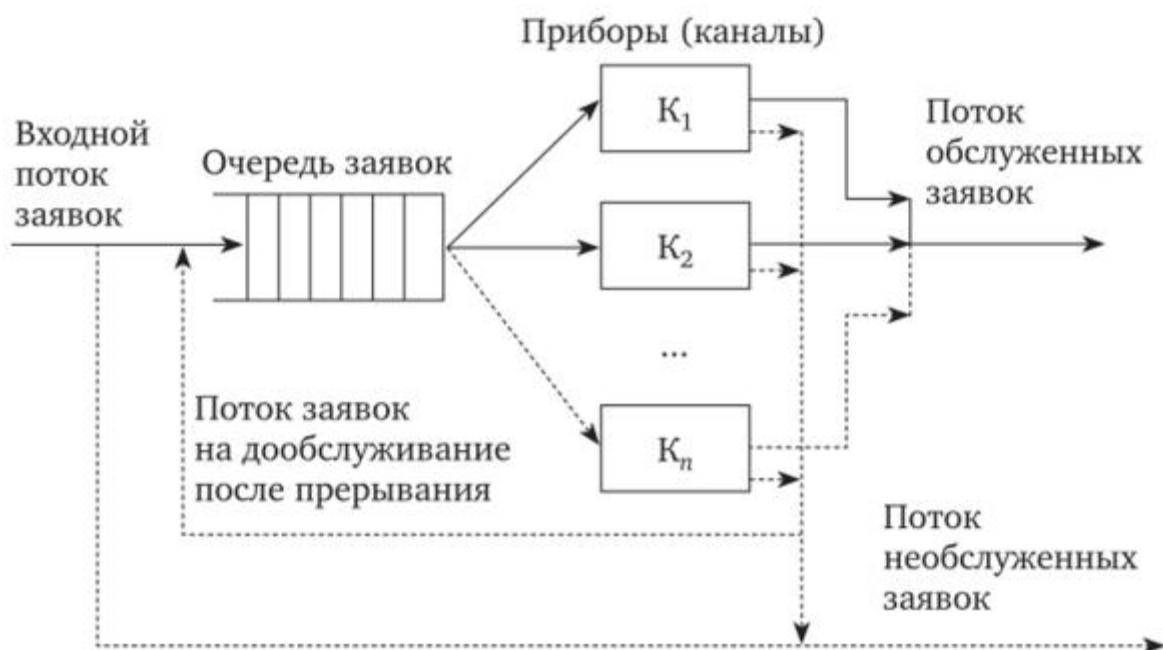


Рисунок 2.1 – Система массового обслуживания

Обслуживающий прибор (прибор, устройство, канал, линия, орудие, автомобиль, маршрутизатор и т.п.) — элемент СМО, функция которого является обслуживание запросов.

Обслуживание — задержка заявки в обслуживающем устройстве на определенн время.

Длительность обслуживания — время задержки (обслуживания) заявки в устройстве.

Накопитель (буфер, входной буфер, выходной буфер) — совокупность мест для ожидания заявок перед обслуживающим прибором. Количество мест для ожидания — емкость накопителя.

Заявка, поступившая в СМО, может находиться в двух состояниях [10]:

- 1) обслуживания (в приборе);
- 2) ожидания (в накопителе), если все приборы заняты обслуживанием других заявок.

Заявки, которые находятся в накопителе и ожидают обслуживания, создают очередь заявок. Количество заявок в накопителе, ожидающих обслуживания, - длина очереди.

Дисциплина буферизации (Queuing discipline) - правило для ввода входящих запросов в хранилище (буфер).

Служебная дисциплина-правило выбора запросов из очереди на обслуживание в устройстве.

Приоритет - преимущественное право (захватывать ресурсы) вводить запросы одного класса в запоминающее устройство или выбирать их из очереди на обслуживание в устройстве по отношению к запросам других классов.

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 представляет собой совокупность уравнений или формул, позволяющих 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, а время обслуживания распределяется экспоненциально). В этом случае все процессы в ЦМО можно описать обыкновенными дифференциальными уравнениями, а в предельном случае — для стационарных состояний — линейными алгебраическими уравнениями и, решая их любыми методами, доступными в математических программных пакетах, определить выбранные показатели эффективности.

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.

Для описания потока запросов в общем случае необходимо указать временные интервалы $t = t_k - t_{k-1}$ между соседними моментами t_{k-1} и t_k поступления запросов с порядковыми номерами $k-1$ и k соответственно ($k=1, 2, \dots$; $t_0=0$ -начальное время).

Основной характеристикой потока запросов является его интенсивность X — среднее количество запросов, поступающих на вход СМО в единицу времени. Значение $t = 1/X$ определяет средний интервал времени между двумя последовательными запросами.

Поток называется детерминированным, если временные интервалы t_k между соседними запросами принимают определенные заранее известные значения. Если интервалы одинаковы ($x_k = t$ для всех $k = 1, 2, \dots$), то этот поток называется регулярным. Чтобы полностью описать регулярный поток запросов, достаточно задать скорость потока X или значение интервала $t = 1/X$.

Поток, в котором временные интервалы x_k между соседними запросами являются случайными величинами, называется случайным. Чтобы полностью описать случайный поток запросов в целом, необходимо указать законы распределений $F_{fc}(x_{fc})$ для каждого из временных интервалов x_k , $k = 1, 2, \dots$

Случайный поток, в котором все временные интервалы x_1, x_2, \dots между соседними последовательными приложениями находятся независимые случайные величины, описываемые функциями распределения $F_1(x_1), F_2(x_2), \dots$ соответственно, это называется потоком с ограниченным последствием.

Случайный поток называется рекуррентным потоком, если все временные интервалы x_1, x_2, \dots между собой запросы распределяются по одному и тому же закону $F(t)$. Существует много повторяющихся потоков. Каждый закон распределения порождает свой собственный рекуррентный поток. Рекуррентные потоки иначе называются ладонными потоками [11].

Если интенсивность X и закон распределения $F(t)$ интервалов между последовательными запросами не изменяются во времени, то поток запросов называется стационарным. В противном случае поток запросов будет нестационарным.

Если только один запрос может появиться на входе СМО в любой момент времени, то поток запросов называется ординарным. Если в любой момент времени может появиться более одного запроса, то поток запросов будет неординарным или групповым.

Поток запросов называется потоком без последствия, если запросы принимаются независимо друг от друга, то есть момент получения следующего запроса не зависит от того, когда и сколько запросов было получено до этого момента.

Стационарный обычный поток без последствия называется самым простым. Интервалы времени t между заявками в простейшем потоке распределены по экспоненциальному (показательному) закону: с функцией распределения $F(t) = 1 - e^{-\lambda t}$; плотностью распределения $f(t) = \lambda e^{-\lambda t}$, где $\lambda > 0$ — параметр распределения — интенсивность потока заявок.

Простейший поток часто называют пуассоновским. Название происходит от того, что для этого потока вероятность $P_{fc}(At)$ появления

ровно к заявок за некоторый интервал времени Δt определяется законом Пуассона:

$$P_k(\Delta t) = \frac{(\lambda \cdot \Delta t)^k}{k!} e^{-\lambda \cdot \Delta t}$$

Следует заметить, что пуассоновский поток, в отличие от простейшего, может быть:

- стационарным, если интенсивность X не меняется со временем;
- нестационарным, если интенсивность потока зависит от времени: $X \geq(t)$.

В то же время простейший поток, по определению, всегда является стационарным.

Аналитические исследования моделей массового обслуживания часто проводятся в предположении о простейшем потоке запросов, что обусловлено рядом присущих ему замечательных 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.

Сумма N независимых стационарных ординарных потоков с интенсивностями x_1, x_2, \dots, x_N образует простейший поток с интенсивностью при условии, что добавленные потоки оказывают более или менее одинаково малое влияние на общий поток. На практике полный поток близок к простейшему при $N > 5$. Это означает, что при суммировании независимых простейших потоков суммарный поток будет самым простым для любого значения N .

2. Вероятностное разбавление потока. Вероятностное (но не детерминированное) 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

Системы массового обслуживания используются для исследования реальных вероятностных объектов. Общий вид имитационной модели ОСМО по классификации Кендалла М/М/м/Е. Параметр m определяет количество обслуживающих аппаратов, а параметр E - ограничение по количеству 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 программного продукта для создания имитационной модели была выбрана система структурного и имитационного моделирования (ССИМ) GPSS Studio [12]. На рисунке 2 приведена структурная модель ОСМО М/М/5/0 в ССИМ GPSS Studio.

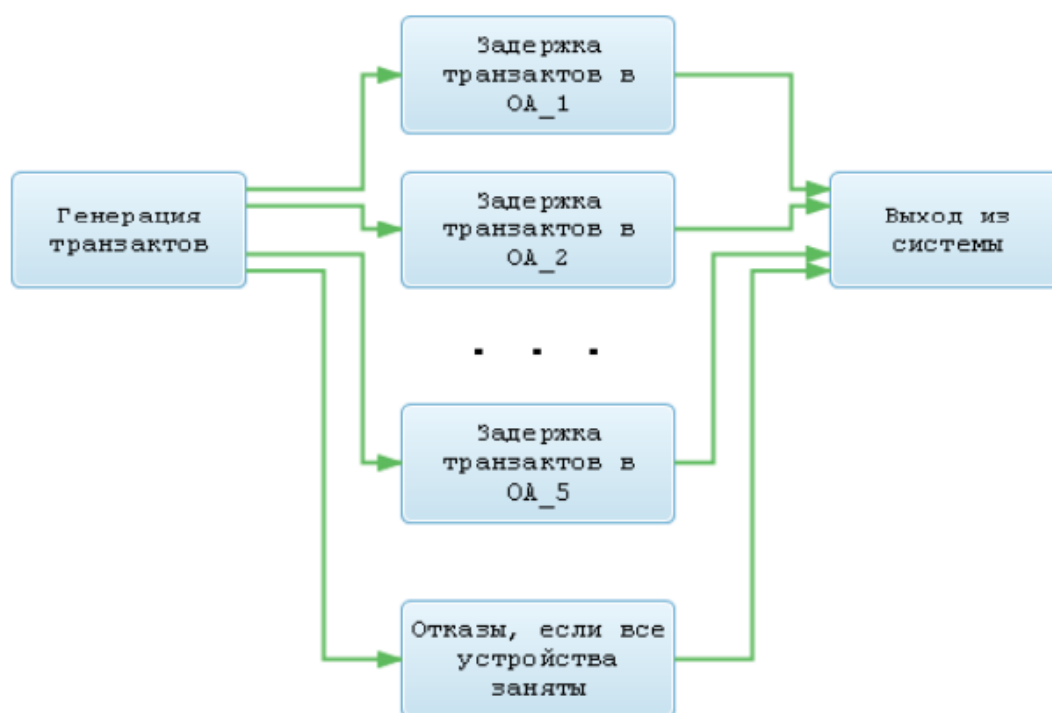


Рисунок 2.2 – Структурная модель ОСМО М/М/5/0 в ССИМ GPSS Studio

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

Имя / номер	Кол-во тран. ожидающих выполнения	Значение сохраняемой величины	
1. Среднее время между поступлением заявок.	TIMEOFFARRIVAL	0	10.000
2. Среднее время обслуживания одной заявки.	TIMEOFFSERVICE	0	30.000
3. Общее количество обслуживаемых заявок.	AMOUNTOFFTRANSACTS	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.	AVERAGETIMEOFFSTAY	0	30.825
8. Среднее время обслуживания заявки в ОА.	AVERAGETIMEOFFSERVICE	0	30.825
9. Среднее время ожидания заявки в очереди.	AVERAGEOFFWAIT	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

Для сравнения результатов ИМ и АМ использованы параметрические и непараметрические методы. В качестве параметрических были выбраны 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	А		ИМ	Разница по Стьюденту
	М			
Среднее время пребывания заявки в системе	0.0	3 143	30.	1.475
Среднее время задержки заявки в ОА	0.0	3 143	30.	1.475
Среднее количество заявок в ОА	.67	2 95	2.6	1.812
Среднее количество заявок в системе	.67	2 95	2.6	1.812
Итого:		По 4 - 0	тестам:	+4
Сравнение АМ и ИМ по критерию Пирсона для М/М/5/0				
Наименование	А		ИМ	Разница по Стьюденту
	М			
Среднее время пребывания заявки в системе	.90	1 84	1.8	10085.9 41
Среднее время задержки заявки в ОА	.90	1 84	1.8	10085.9 41
Среднее количество заявок в ОА	00	9 .62	945	10505.8 62
Среднее количество заявок в системе	00	9 .62	945	10505.8 62
Итого:		По 4 - 0	тестам:	+4

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

Непараметрические методы сравнения не требуют, чтобы были заданы

параметры в явном виде, и используют для оценки достоверности интегрально-важные 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 требуются для вычисления средних квадратических отклонений. Для этого используются 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:

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

Ввиду того что вычисленное значение коэффициента линейной корреляции ненамного отличается от 1 является верным заключение о наличии между значениями параметра «Среднее время пребывания в системе» для АМ и ИМ сильной корреляционной связи, очень близкой к линейной [11].

По результатам проделанной работы сделаны выводы о том, что проанализированная ССИМ GPSS Studio показала высокие результаты по всем тестам, поэтому констатируется ее пригодность для проведения ИМ различных СМО. Проведено сравнение 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

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.

Устройства интернета вещей могут быть мобильными, маломощными, недорогими, и в зависимости от условий окружающей среды результирующие приложения имеют различные 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.

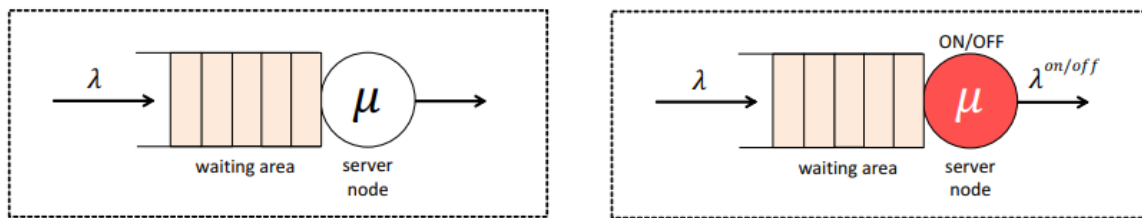


Рисунок 2.4 – Модели M/M/1 и ON/OFF

Очередь M/M/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 .

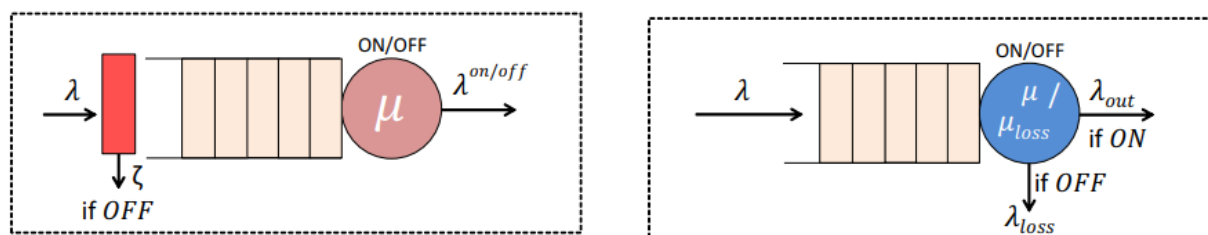


Рисунок 2.5 – Вероятностные и убыточные очереди ВКЛ /ВЫКЛ

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:

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

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.

D. 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.

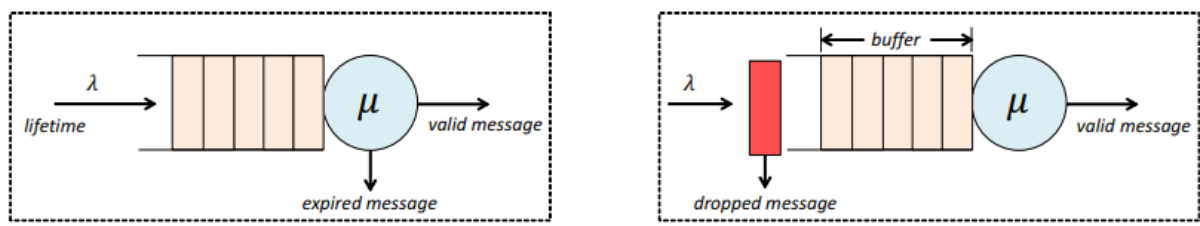


Рисунок 2.6 – Очереди с истечением срока действия событие и ограничение по мощности

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

Предсказание поведения сложных систем реального мира - вот что такое мир GPSS.

Многие дорогостоящие проекты в прошлом потерпели неудачу, потому что конечный результат не был точно охарактеризован. От максимальной мощности до стоимости эксплуатации важно, чтобы детальные знания о поведении строящейся системы были известны как можно скорее. Хотя чисто математические модели чрезвычайно ценны и должны использоваться там, где это возможно, сложность большинства реальных систем требует использования компьютерного моделирования для получения необходимых ответов. Вот тут-то и появляется мир GPSS.

Мир GPSS основан на основополагающем языке компьютерного моделирования-GPSS, который расшифровывается как система моделирования общего назначения. Этот язык был разработан главным образом Джеффри Гордоном в IBM около 1960 года и внес важные концепции в каждый коммерческий язык компьютерного моделирования дискретных событий, разработанный с тех пор. GPSS World является прямым потомком GPSS / PC, ранней реализации GPSS для персональных компьютеров. С момента своего появления в 1984 году GPSS / PC и его преемники сэкономили тысячам пользователей миллионы долларов. Теперь реализация Windows GPSS World расширяет эти возможности в среде с поддержкой Интернета.

GPSS World предназначена для быстрого и надежного предоставления ответов, с наименьшими усилиями, достигая наивысшей надежности результатов. В соответствии с этими целями визуализация выполняемых симуляций сильно стилизована, и в нее встроена статистическая обработка по умолчанию. Этот подход означает, что анимация является "свободной", не требующей дополнительных усилий для создания, но не является фотореалистичной. Сильной стороной GPSS World является прозрачность, а не фотореализм. Доступны сторонние анимационные системы, которые могут обеспечить графическую анимацию на основе моделирования мира GPSS.

Прозрачность ценна по трем причинам. Во-первых, опасно полагаться на непрозрачную симуляцию "черного ящика", внутренние механизмы которой невозможно наблюдать. Вы не только не можете быть уверены, что он соответствует вашей ситуации, но и трудно быть уверенным, что он даже работает так, как задумано. Во-вторых, успешные симуляции ценны и имеют удивительно долгий срок службы. Вполне возможно, что новым сотрудникам потребуется ознакомиться с внутренней работой имитационного моделирования - задача почти невыполнимая, если только уже не будет обеспечен высокий уровень транспарентности. В-третьих, одним из наиболее эффективных, но наименее упоминаемых преимуществ компьютерного моделирования является понимание поведения системы, достигаемое, когда опытный специалист по моделированию может видеть внутреннюю динамику в критические моменты моделирования.

GPSS World была создана для решения этих проблем. Его визуализируемая природа позволяет выявить и запечатлеть внутренние механизмы моделей. Его интерактивность позволяет исследовать и манипулировать моделированием. Его встроенный инструмент анализа данных позволяет легко вычислять доверительные интервалы и дисперсионный анализ. И теперь он может даже создавать и запускать сложные эксперименты по скринингу и оптимизации автоматически, с относительно небольшими усилиями с вашей стороны.

Большинство систем можно смоделировать любым из нескольких способов, используя GPSS World. Обычно требуется использовать только небольшое подмножество доступных функций. Однако самое большое мастерство требует знакомства со всем, что может предложить мир GPSS. Это руководство является основным источником такой информации.

GPSS состоит из 4 объектов. Объекты Model, Simulation и Report образуют 3 основных типа, которые используются во всех моделях GPSS World simulations. Как правило, модель разрабатывается путем редактирования операторов в объекте модели. Затем выдается команда Create Simulation, тем самым создается объект моделирования, структурированный в соответствии с инструкциями в модели. Моделирование выполняется путем отправки им команд GPSS или путем включения команд в сам объект модели. Обычно после завершения моделирования объект отчета создается автоматически.

Общий контроль над несколькими прогонами моделирования может быть осуществлен в рамках этой структуры путем включения дополнительного эксперимента в объект модели. (Плюс-это "язык программирования под симуляцией", который можно использовать для определения процедур и экспериментов.) Затем на объект моделирования может быть отправлена команда поведения для управления серией запусков моделирования и обработки результатов.

Текстовый объект завершает 4 типа объектов мира GPSS. Они названы в инструкции INCLUDE, чтобы помочь разработке больших симуляций, а также для нескольких других удобных применений, таких как создание библиотеки исходного кода. Кроме того, текстовые объекты часто используются объектами моделирования, когда выполняются операции открытия, закрытия, чтения, записи и поиска.

Все объекты GPSS могут быть сохранены в любое время, в любом состоянии и вновь открыты в более позднем сеансе. Это включает в себя объекты моделирования, которые могут быть запущены любое количество раз из сохраненного состояния, что облегчает отладку и демонстрацию. При некотором тщательном планировании объекты моделирования могут быть сохранены в середине сложных экспериментов, что позволит вам завершить запуски моделирования в более позднее время.

Все объекты GPSS, кроме текстовых объектов, содержат данные в дополнение к основному тексту ASCII. Вот почему вы можете использовать внешний текстовый редактор для редактирования текстового объекта, но не любого другого. Объекты модели, моделирования и отчета содержат закладки для удобства навигации и список значений атрибутов, называемых настройками, которые могут быть изменены и унаследованы дочерними объектами. При создании объекта моделирования он начинается с копии настроек модели, которая его создала. То же самое относится и к объектам отчета, которые наследуют свои настройки от родительского объекта моделирования. Кроме того, объекты модели запоминают все ошибки перевода, возникшие при последней попытке создать из них объект моделирования. Таким образом, вы можете внести некоторые исправления, а остальное оставить на потом.

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.

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 А.

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

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 А. Все наблюдения на данном уровне лечения анализируются с использованием одного и того же значения для а. 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 а, 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 А и фактором в, 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 а и в 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.

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

Каждый фактор и взаимодействие в статистической модели представлены отдельной строкой в таблице ANOVA. В каждой строке мы имеем сначала сумму квадратов и степеней свободы, связанных с этой оценкой. Это основы, из которых выводятся другие числа. Простое деление приводит к среднему квадрату, и, разделив этот фактор на средний квадрат ошибки, из нижней строки таблицы мы получаем статистику F для этого эффекта.

Теперь мы готовы сделать некоторые выводы. Мы должны решить, достаточно ли велико значение F, чтобы объявить, что эффект значителен.

Пороговое значение, которое мы будем использовать для сравнения, называется "критическим значением F" и помещается справа от нашей статистики F в той же строке. Если наше значение F превышает критическое значение, мы приходим к выводу, что имеем дело со значительным эффектом. Если нет, то мы приходим к выводу, что эффект не является значительным, и мы игнорируем любые связанные с ним изменения в наблюдениях, поскольку они вызваны только случайным шумом. Чем больше значение F, тем сильнее эффект. Таблица ANOVA показывает, что влияние фактора A является значительным, а влияние фактора B и взаимодействия AB - незначительным.

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.

Желательно удалить любую часть суммы квадратов погрешности, которая обусловлена каким-либо важным эффектом, не включенным нами в анализ. Если мы сможем это сделать, то наша статистика F, как правило, будет больше. 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.

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

верно, что эти эффекты взаимодействия не существуют, это позволит GPSS World использовать дополнительные степени свободы для достижения лучшей оценки статистики F. Кроме того, большее число степеней свободы означает также меньшее критическое значение F. Однако мы должны понимать, что удаление терминов из статистической модели предполагает отсутствие эффекта высокого порядка. 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 [3,2]. На самом деле, поскольку реплики обычно помечаются на конце, первый запуск в этой комбинации лечения

будет идти в элемент [3,2,1] матрицы результатов. Каждый размер матрицы результат анализируется как фактор, с помощью процедуры дисперсионного анализа библиотеки. Размер измерения в Матрице результатов должен быть по меньшей мере таким же большим, как число уровней обработки этого фактора или максимальное число реплик.. Нет никакого произвольного ограничения на количество уровней обработки в пределах фактора, только то, что накладывается виртуальной памятью вашего компьютера.

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 Основные понятия теории планирования экспериментов

Теория исходит из абстрактной схемы сложной системы, называемой "черным ящиком" (рис.1). Предполагается, что исследователь может наблюдать входы и выходы "черного ящика" (имитационная модель) и определять взаимосвязь между входами и выходами на основе результатов наблюдений.

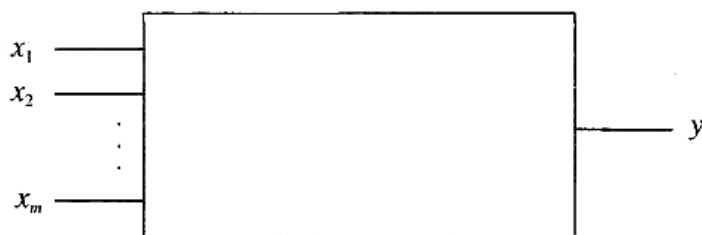


Рисунок 3.1 - Абстрактная схема системы

Эксперимент на имитационной модели будет рассматриваться как состоящий из наблюдений, а каждое наблюдение — из прогона модели.

Входные переменные x_1, x_2, \dots, x_T называются факторами. Выходная переменная y называется наблюдаемой переменной (реакция, отклик).

Факторное пространство - это совокупность факторов, значения которых исследователь может контролировать в процессе подготовки и проведения модельного эксперимента [21].

Каждый фактор имеет свои уровни. Уровни-это значения, которые устанавливаются для каждого фактора при определении условий выполнения модели в наблюдении.

Цель эксперимента-найти функцию y , и предполагается, что значение отклика состоит из двух составляющих

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

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: $D[y] = D[e]$.

Дисперсионный анализ - это статистический метод анализа результатов наблюдений, зависящих от различных одновременно действующих факторов, выбора наиболее важных факторов и оценки их влияния.

В экспериментальных условиях факторы могут изменяться, так что можно исследовать влияние фактора на наблюдаемую переменную. Если влияние какого-то фактора на наблюдаемую переменную изменяется при изменении уровня какого-то другого фактора, то говорят, что между факторами существует взаимодействие.

Суть анализа заключается в разложении общей вариации случайной величины на независимые термины-эффекты, каждый из которых характеризует влияние конкретного фактора (основной эффект) или их взаимодействие (эффект взаимодействия) [22].

Эксперимент, реализующий все возможные комбинации уровней факторов, называется полным факторным экспериментом (ПФЭ).

Общее число различных сочетаний уровней в ПФЭ для t факторов можно вычислить по формуле

$$S = k_1 \cdot k_2 \cdot k_3 \cdot \dots \cdot k_i \cdot \dots \cdot k_m, \quad (3.2)$$

где k_i — число уровней i -го фактора.

Если количество уровней для всех факторов одинаково, то $S = k^m$. Каждая комбинация уровней факторов соответствует одному наблюдению.

Недостатком ПФЭ является высокая стоимость подготовки и внедрения, так как количество наблюдений в эксперименте увеличивается с увеличением количества факторов и их уровней. Например, если имеется шесть факторов с двумя уровнями каждый, то даже при одном прогоне модели в каждом наблюдении нужно $S = 2^6 = 64$ наблюдения. Очевидно, что каждый прогон удваивает это число, следовательно, увеличивает затраты машинного времени.

Возрастает также число взаимодействий различных порядков. В таблице 1 приведено число главных эффектов и эффектов взаимодействий различных порядков, которые могут быть выявлены при полном исследовании 2^T сочетаний уровней.

Таблица 3.1 - Число главных эффектов и эффектов взаимодействий различных порядков

m	2^m	Число главных эффектов	Число эффектов взаимодействий				
			1-го порядка	2-го порядка	3-го порядка	4-го порядка	5-го порядка
5	32	5	10	10	5	1	1
6	64	6	15	20	15	6	1
7	128	7	21	35	35	21	7
8	256	8	28	56	70	56	28

На практике бывает необходимо исследовать влияние на наблюдаемую переменную, например, десяти факторов ($t = 10$), каждый из которых имеет четыре уровня ($k = 4$). В этом случае $S = k^t = 4^{10} = 1\,048\,576$.

Такого рода проблемы были одной из причин появления теории планирования экспериментов.

Планирование эксперимента - это раздел математической статистики, изучающий рациональную организацию измерений, подверженных случайным ошибкам.

План эксперимента - это набор значений факторов, содержащих значения оценок функции отклика, удовлетворяющих некоторому критерию оптимальности, например точности.

Существует разница между стратегическим планированием эксперимента и тактическим планированием эксперимента.

3.2 Стратегическое планирование эксперимента

Целью стратегического планирования эксперимента является определение количества наблюдений и комбинаций факторных уровней в них для получения наиболее полной и достоверной информации о поведении исследуемой системы.

При стратегическом планировании эксперимента необходимо решить две основные задачи:

- а) идентификация факторов;
- б) выбор уровней факторов.

Под идентификацией факторов понимается их ранжирование по степени влияния на значение наблюдаемой переменной.

По итогам идентификации целесообразно разделить все факторы на две группы — первичные и вторичные.

Первичные - это факторы, исследование которых необходимо провести.

Вторичные - факторы, которые не являются предметом исследования, но влиянием которых нельзя пренебречь.

Выбор уровней факторов производится с учетом двух противоречивых требований:

а) уровни фактора должны перекрывать весь возможный диапазон его изменения;

б) общее количество уровней по всем факторам не должно приводить к большому количеству наблюдений.

Нахождение компромиссного решения, отвечающего этим требованиям, является задачей стратегического планирования эксперимента.

Поскольку ПФЭ может потребовать много машинного времени, необходимо иметь методы отбора факторов, оказывающих существенное влияние на ответную реакцию [23]. Получается, что если исследователь не интересуется взаимодействиями высокого порядка, то можно получить большой объем информации, исследуя только определенную ее часть (1/2, 1/4, 1/8 и т. д.) всех возможных комбинаций уровней факторов.

Если в эксперименте делается только часть возможных наблюдений, т. е. выборка сокращается, то эксперимент называется частным факторным экспериментом (ПФЭ).

Когда используется образец, который меньше, чем требуется для ПФЭ, плата взимается из-за риска смешивания эффектов. Смешивание относится к тому факту, что исследователь, измеряя один эффект, может также измерять какой-то другой эффект. Например, если основной эффект смешивается с взаимодействием более высокого порядка, то эти два эффекта больше не могут быть отделены друг от друга. Таким образом, если анализ показывает наличие какого-то эффекта, то нельзя с уверенностью сказать, является ли это основным эффектом, или эффектом взаимодействия, или какой-то аддитивной комбинацией этих эффектов.

При построении плана ЧФЭ исследователь должен определить эффекты, которые они могут позволить смешать. Успех ЧФЭ достигается, если его план позволяет не смешивать один основной эффект с другим..

Если количество факторов невелико (обычно меньше пяти), то ПФЭ нецелесообразен из-за смешения эффектов, что не позволяет различать основные эффекты и важные взаимодействия.

В качестве примера рассмотрим план дробного факторного эксперимента (ДФЭ) — один из видов ДФЭ, с полным числом возможных комбинаций 25.

В ДФЭ каждый фактор имеет два уровня-нижний и верхний, поэтому общее число наблюдений $S = 2^T$. планы, построенные с помощью этого метода, обладают определенными свойствами (симметрия, нормальность, ортогональность и вращаемость), которые улучшают качество экспериментов.

Этот план строится следующим образом. Мы смешиваем взаимодействия 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.

Мы будем делать это, выбирая двух - и трехфакторные взаимодействия с общим фактором. В данном случае мы выбрали BCE и ADE, имеющие

общий фактор E. Смешав эти взаимодействия, мы автоматически смешали ABCD. Поэтому мы получили так называемое тождество или определяющее разбиение, $I=BCD=ABCE=ACDE=ABDE=BCDE=ABCE=ACDE=ABDE=BCDE$, которое было использовано для деления 32 наблюдений на четыре блока по восемь наблюдений в каждом [21]. Это означает, что мы утратили возможность распознавания таких эффектов. Используя это тождество как основу для смешивания в нашем плане, мы должны сделать то же и со следующими эффектами:

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

Получили семь групп смешивания. Из этого следует, что если взаимодействий нет или ими можно пренебречь, то мы можем по восьми наблюдениям получить информацию о каждом из пяти главных эффектов (A, B, C, D и E) и использовать данные эффектов взаимодействий AB и BC для оценки ошибки дисперсионного анализа.

3.3 Тактическое планирование эксперимента

Целью планирования тактического эксперимента является определение необходимого количества пробегов модели в каждом наблюдении.

Поскольку моделирование является статистическим экспериментом, необходимо получить достоверный результат с заданной точностью.

В общем случае количество пробегов модели (размер выборки), необходимое для получения оценок наблюдаемой переменной с заданной точностью, зависит от следующих факторов [24]:

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

При отсутствии информации о перечисленных факторах количество пробегов модели (размер выборки) 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 среднеквадратического отклонения σ_α до начала эксперимента неизвестно хотя бы ориентировочно, целесообразно выполнить пробное количество прогонов N_0 и вычислить на их основе выборочное среднеквадратическое отклонение или дисперсию, значение которой подставить в (3.1) и получить предварительную оценку числа прогонов N_T . Затем выполнить $N_T - N_0$ оставшиеся прогоны, 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 случай», 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 .

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

Двухпроцессорный вычислительный комплекс (ВК) обслуживает три автоматизированных рабочих места (АРМ). Первое АРМ имеет абсолютный приоритет по отношению к остальным, а второе — более высокий приоритет по отношению к третьему АРМ. От первого АРМ поступают запросы на обслуживание, интервалы между которыми распределены равномерно $[t_1, t_2]$. Интервалы времени поступления запросов от второго и третьего АРМ распределены по экспоненциальному закону с параметрами t_3, t_4 соответственно. На обслуживание одного запроса 1-й категории любой процессор затрагивает время, экспоненциально распределенное с параметром t_5 с, 2-й категории — t_6 с, 3-й категории — t_7 с. В оперативной памяти ВК организован накопитель емкостью L запросов.

Каждый процессор имеет конечную надежность. Длительность интервалов между отказами распределена по экспоненциальному закону с параметром t_8 с. Длительность восстановления распределена по нормальному закону с параметрами t_9 с и t_{10} с. При отказе любого процессора обслуживаемые и прерванные запросы теряются.

Наша задача разработать имитационную модель с целью исследования в течение 2 ч функционирования двухпроцессорного ВК зависимости вероятности потерь запросов от емкости накопителя, времени обработки запросов, интервалов времени выхода из строя и времени восстановления процессоров. 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
ASSIGN      2,(Exponential(13,0,T6_))
TRANSFER    ,Met0
GENERATE    (Exponential(2,0,T4_))

```

Продолжение Listing

```

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 нужно открыть модель
Прямая задача. Выбираем Edit/InsertExperiment/ Screening ...

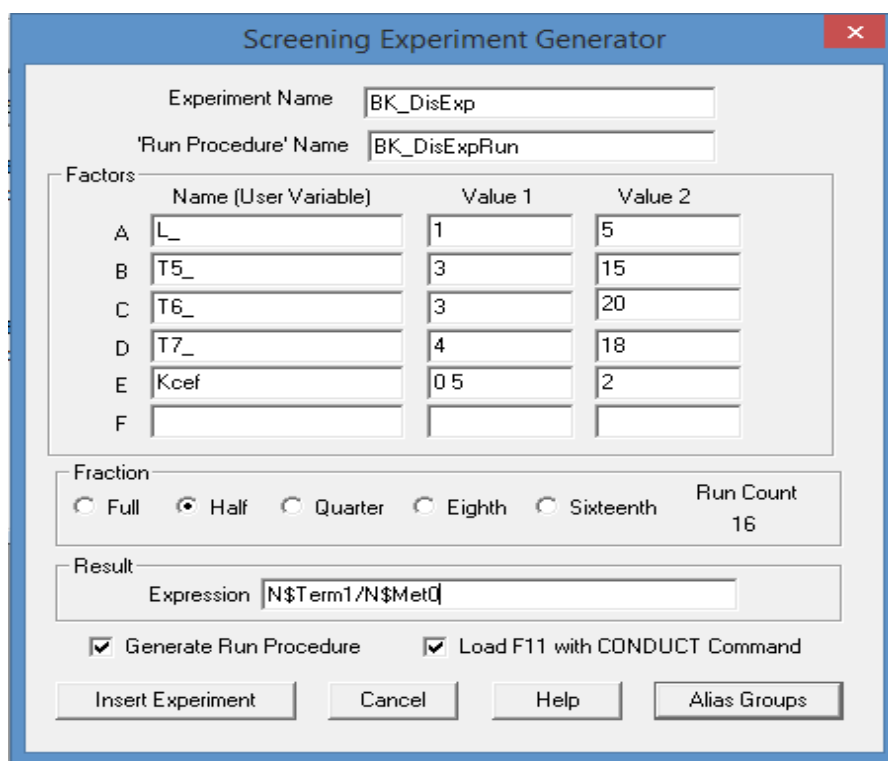


Рисунок 3.2 - Диалоговое окно Screening Experiment Generator (Генератор отсеивающего эксперимента)

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

Дальше идет группа полей Factors (Факторы) в примере Listing определяется вероятность потерь запросов, поступающих с АРМ. В качестве факторов, влияние, которых на вероятность потерь запросов необходимо исследовать, рекомендуются следующие:

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;

Koef - коэффициент изменения среднего времени восстановления T9_ и среднеквадратического отклонение T10_ времени восстановления.

Каждое имя фактора должно быть именем переменной пользователя. Оно должно начинаться с буквы и не совпадать с ключевым словом, СЧА или классом СЧА. Это было учтено при разработке модели Пример 3.1. Для изменения среднего времени восстановления T9_ и среднеквадратического отклонение T10_ времени восстановления введена переменная пользователь Koef. За счет этого вместо двух факторов используется один (Koef)[25].

В GPSS World максимальное количество факторов, влияние которых на функцию отклика можно исследовать посредством дисперсионного анализа, равно шести. Для каждого фактора необходимо выбрать два уровня.

Рекомендуется выбирать уровни, значительно отстоящие друг от друга. Это необходимо для получения также значительно отличающихся откликов.

В рассматриваемом примере выбраны пять факторов и для каждого фактора - по два уровня.

Следующей идет группа Fraction (Часть дробного эксперимента). Эксперимент, проводимый в GPSS World, может быть ПФЭ или ДФЭ. Группа Fraction позволяет это задавать. Справа под Run Count появится число 16. Это количество наблюдений, которые необходимо сделать. Количество прогонов модели в каждом наблюдении будет указано позже.

В поле Expression (Выражение) группы Resu(t (Результат) ввели выражение:

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

по которому вычисляется вероятность потерь запросов с АРМ.

После группы Result (Результат) идут два флажка, позволяющие выбирать опции. При выборе опции Generate Run Procedure GPSS World создала вместе с экспериментом стандартную процедуру запуска, которую мы можем откорректировать согласно своим требованиям. Выбор опции Load F11 with CONDUCT Command закрепляет соответствующую команду CONDUCT за функциональной клавишей [F11]. Тогда после создания объекта «Процесс моделирования» для запуска эксперимента нужно только нажать функциональную клавишу [F11]. Выбрала обе опции.

Перед созданием эксперимента необходимо изучить группы смешивания с целью осуществления стратегического планирования эксперимента [26]. Для этого нужно нажать кнопку Alias Groups (Группы смешивания). Появится диалоговое окно Alias Groups (Группы смешивания) (рисунок 3). При изучении групп смешивания необходимо вначале найти отсутствующие факторы, а затем факторы, которые неразличимы, так как находится в одной группе смешивания. Факторы обозначаются одной буквой. Например, пятый фактор обозначается буквой E, а взаимодействие первого и пятого факторов - AE.

Из рисунка 3 видно, что фактор E отсутствует, о нем не будет никакой информации. Факторы C и D находятся в одной группе и, следовательно, будут неразличимы

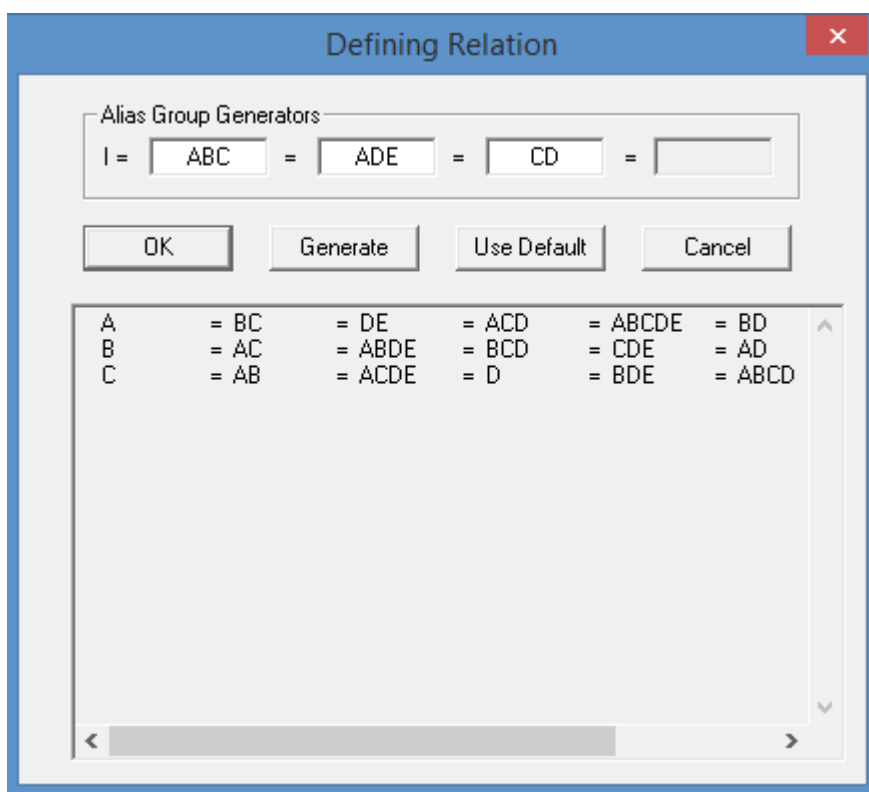


Рисунок 3.3 - Группы смешивания

Распределение факторов по группам смешивания, а следовательно, и получаемых эффектов, можно изменить, используя поле Alias Generators (Порождающие функции). Для этого нужно изменить порождающие функции и нажать кнопку Generate (Создать). В качестве порождающих функций следует использовать наборы взаимодействий высших порядков. Когда GPSS World создает группы смешивания, сначала он расширяет множество порождающих функций, формируя 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 включаются в 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. При невыполнении какого-либо из этих условий GPSS World известит об ошибке.

Для возврата к исходным порождающим функциям, предложенным GPSS World, нужно нажать кнопку Use Default (По умолчанию).

В рассматриваемом примере имеются только три группы смешивания, поэтому невозможно различить эффекты каждого из пяти факторов. Закроем диалоговое окно Alias Groups (Группы смешивания), нажав кнопку Cancel (Отмена).

В диалоговом окне Screening Experiment Generator (Генератор отсеивающего эксперимента) в группе Fraction (Часть дробного эксперимента) установила Quarter (1/4). Снова открываем диалоговое окно Alias Groups (Группы смешивания).

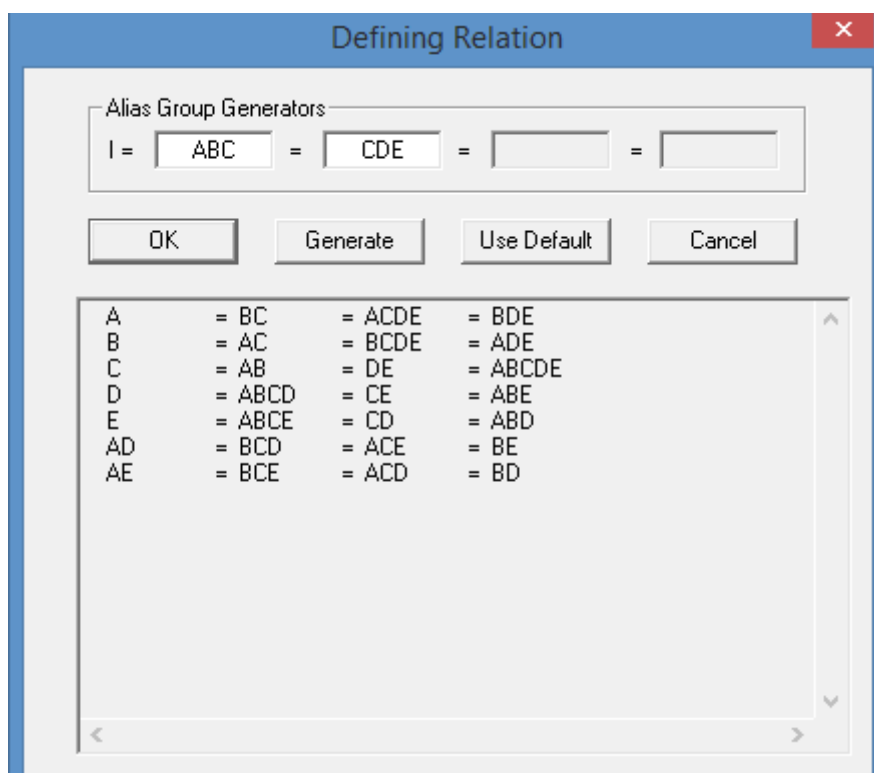


Рисунок 3.4 - Группы смешивания 1/4

Теперь факторы находятся в отдельных группах. Однако невозможно будет судить об эффектах, т.е. 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. Не делаем этого и закроем диалоговое окно Alias Groups (Группы смешивания), нажав кнопку Cancel (Отмена).

В диалоговом окне Screening Experiment Generator (Генератор отсеивающего эксперимента) в группе Fraction (Часть дробного эксперимента) установила Half (1/2). Снова откроем диалоговое окно Alias Groups (Группы смешивания) (рисунок 5).

Предложенный GPSS World план эксперимента лучше предыдущих. Вместо 32 нужно будет выполнить 16 наблюдений. В плане выделены все факторы и двухфакторные взаимодействия нескольких факторов. Создала эксперимент с этим планом, нажав кнопку Ok.

Теперь необходимо создать Plus-операторы и вставить их в нижнюю часть модели Пример 9.1. В левой нижней части диалогового окна Screening Experiment Generator (Генератор отсеивающего эксперимента) нажимаем кнопку Insert Experiment (Вставить эксперимент).

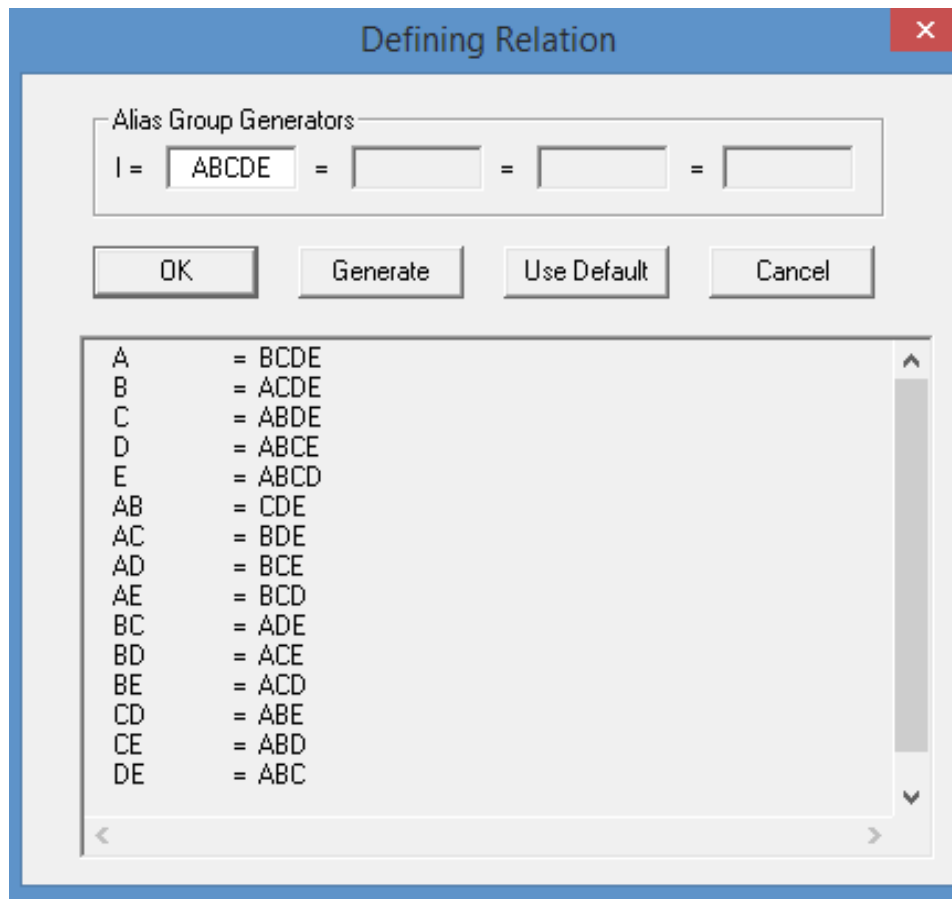


Рисунок 3.5 - Группы смешивания ½

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.

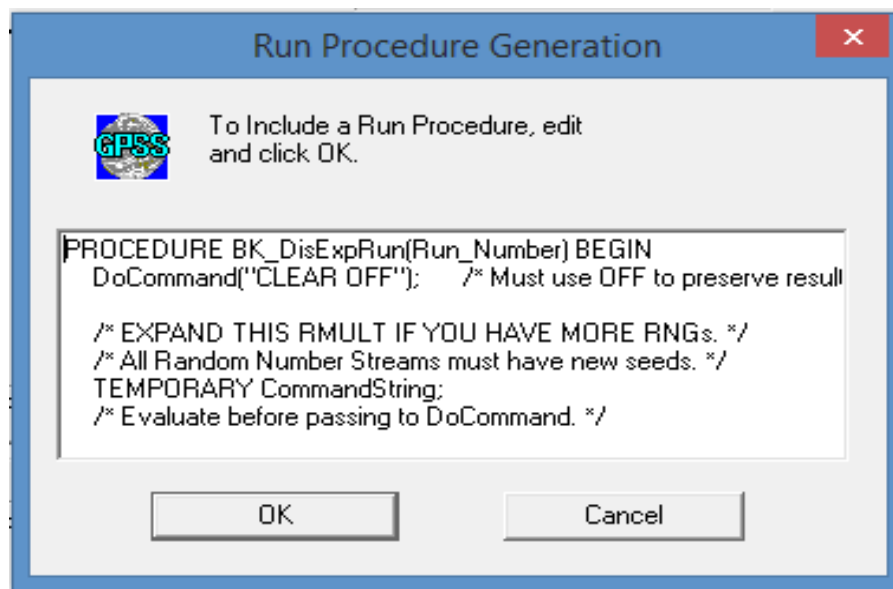


Рисунок 3.6 - Диалоговое окно стандартной процедуры Запуска

Если перелистать стандартную процедуру запуска, в конце ее найдем условия моделирования в каждом наблюдении по умолчанию (рисунок 7). Откорректируем их так, как показано на рисунке 8 (провели тактическое планирование эксперимента, и мы получили требуемое количество прогонов, значительно отличающееся от предложенного).

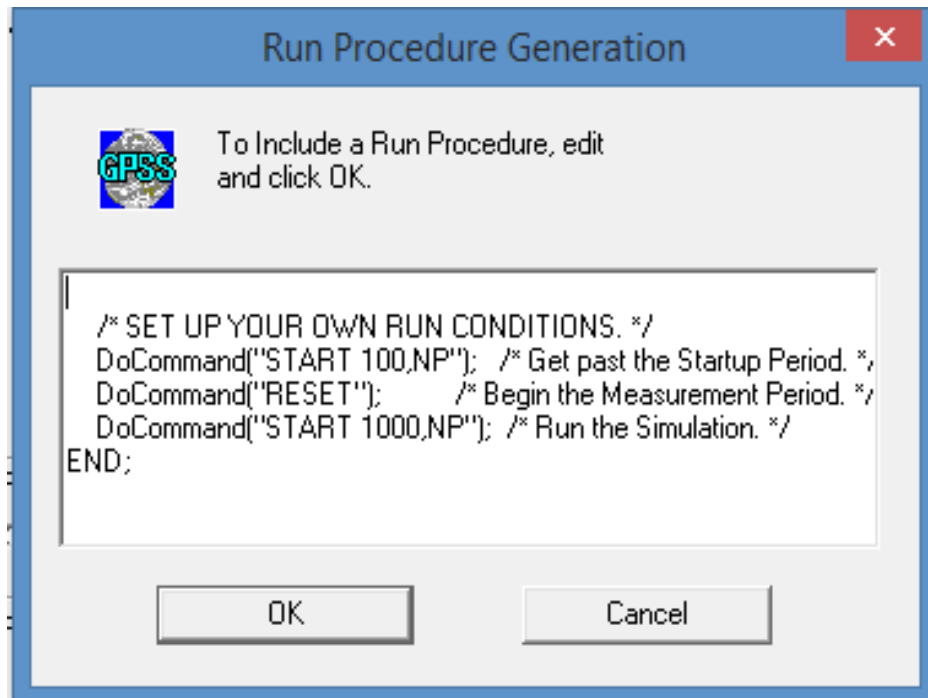


Рисунок 3.7 - Условия стандартной процедуры Запуска по умолчанию

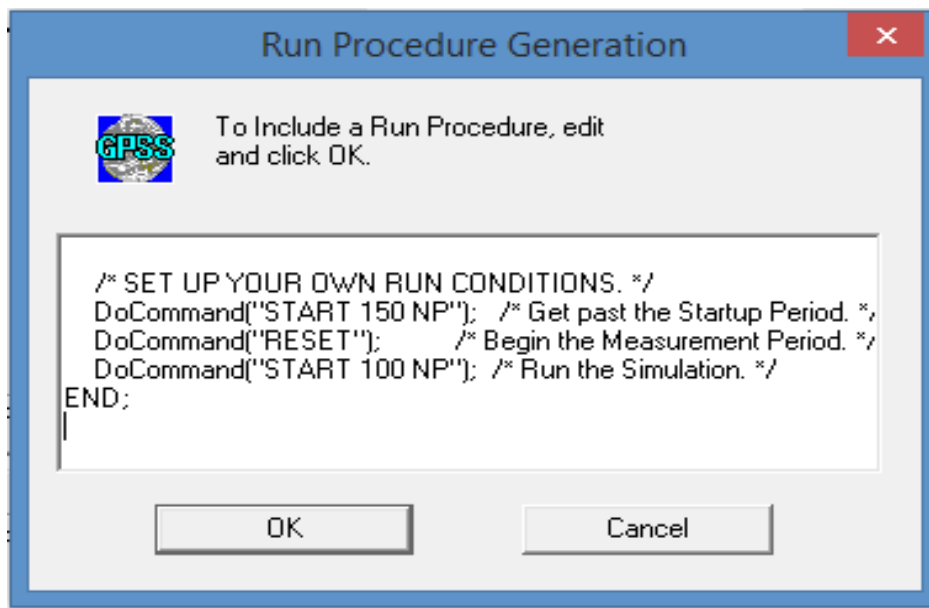


Рисунок 3.8 - Условия стандартной процедуры после корректировки

Корректировка процедуры запуска возможна до и после того, как она будет добавлена к объекту «Моделью».

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

Plus-эксперимент содержит также вызов Plus-процедуры запуска. Процедура запуска осуществляет связь между генерируемым экспериментом и процессом моделирования. Она вызывается столько раз, сколько требуется сделать наблюдений. Так как процедура запуска вызывается Plus-экспериментом, ей разрешается вызывать библиотечную процедуру DoCommand и, следовательно, выполнять RMULT, CLEAR, RESET и многие другие команды GPSS. Поэтому все команды необходимые для определения условий наблюдения, следует помещать в процедуру запуска. Для сохранения матрицы результатов при обнулении переменных перед очередным наблюдением используется команда CLEAR OFF. Для изменения начального числа генератора случайных чисел в каждом наблюдении процедуре запуска передается номер запуска.

Проведем эксперимент. Для вызова эксперимента предназначена команда CONDUCT [28]. Однако за функциональной клавишей [F11] была закреплена соответствующая команда CONDUCT (Edit > Settings > Function Keys (Правка > настройки > функциональные клавиши)).

Провели трансляцию, т.е. создали объект «Процесс моделирования», для чего нажимаем [Ctrl]+[Alt]+[S]. После этого нажимаем [F11]. Эксперимент начинает работать. В ходе работы создается отчет, который по

готовности записывается в окно Journal (Журнал) созданного объекта «Процесс моделирования» (рисунок 9).

```

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 этого 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 таблице каждый фактор и взаимодействие факторов представлены отдельной строкой. В каждой строке для данного эффекта (Effect) имеются результаты основных вычислений и F-статистика. Чем больше значение F-статистики (F - for Only Plain Effects), тем сильнее эффект[27].

Эффект, а, следовательно, и фактор, считается значимым, если превышает критическое значение (Critical Value of F (p=0.05)), которое также вычисляется GPSS World. В нашем примере факторы А, В, С и D являются значимыми, так как их F-статистики больше критического значения, равного 4,96. Наибольший эффект на вероятность потери запросов оказывает фактор С, так как он имеет самую большую статистику (182,140).

Таким образом, по результатам моделирования можно сделать вывод, что при данных потоках и характеристиках процессоров вероятность потерь запросов в среднем составит 0,353. Для 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

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 22:23:44 A Simulation in an Experiment has ended. Clock is 540000.000000.
05/25/20 22:23:44 Simulation in Progress.
05/25/20 22:23:45 A Simulation in an Experiment has ended. Clock is 900000.000000.
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

Продолжение Приложения А

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 22:24:10 A Simulation in an Experiment has ended. Clock is 540000.000000.
 05/25/20 22:24:10 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

05/25/20 22:24:14	Alias	Effect	Sum of	Degrees of	F - for Only
05/25/20 22:24:14	Group		Squares	Freedom	Main Effects
05/25/20 22:24:14					of
05/25/20 22:24:14	A = BCDE	-0.083	0.027	1	19.193
4.96					
05/25/20 22:24:14	B = ACDE	0.162	0.105	1	73.618
4.96					
05/25/20 22:24:14	AB = CDE	0.012			
05/25/20 22:24:14	C = ABDE	0.255	0.260	1	182.140
4.96					
05/25/20 22:24:14	AC = BDE	0.022			
05/25/20 22:24:14	BC = ADE	-0.027			
05/25/20 22:24:14	DE = ABC	0.012			
05/25/20 22:24:14	D = ABCE	0.165	0.109	1	76.060
4.96					
05/25/20 22:24:14	AD = BCE	0.003			
05/25/20 22:24:14	BD = ACE	-0.007			
05/25/20 22:24:14	CE = ABD	0.016			
05/25/20 22:24:14	CD = ABE	-0.037			
05/25/20 22:24:14	BE = ACD	0.014			
05/25/20 22:24:14	AE = BCD	-0.013			

Продолжение Приложения А

05/25/20 22:24:14	E = ABCD	-0.022	0.002	1	1.346
-------------------	----------	--------	-------	---	-------

4.96
05/25/20 22:24:14

05/25/20 22:24:14	Error		0.014	10	
05/25/20 22:24:14	Total		0.518	15	
05/25/20 22:24:14	Grand Mean	0.353			

05/25/20 22:24:14 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 */
        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;

```

Продолжение Приложения Б

```
/* Run 4 */
L_ = 1;
T5_ = 3;
T6_ = 20;
T7_ = 18;
Kcoef = 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;
Kcoef = 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;
```

Продолжение Приложения Б

```
T6_ = 20;
T7_ = 18;
Koeff = 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;
Koeff = 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;
Koeff = 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;
Koeff = 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;
Koeff = 2;
```


Продолжение Приложения Б

```
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;
Koeff = 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;
Koeff = 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;
Koeff = 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;
Koeff = 0.5;
IF
(StringCompare(DataType(BK_DisExp_Results[2,2,2,2,1]),"UNSPECIFIED")'E'0)
THEN BEGIN
```

Продолжение Приложения Б

```
        /* 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. */
```

Продолжение Приложения Б

```
DoCommand("START 150,NP"); /* Get past the Startup Period. */
DoCommand("RESET");      /* Begin the Measurement Period. */
DoCommand("START 100,NP"); /* Run the Simulation. */
END;
*****
```