

РЕФЕРАТ

Тема данной дипломной работы – «Программирование модели услуг реального времени в архитектуре IMS». Целью данной дипломной работы является создание программной имитационной модели, задачей которого будет сбор статистических данных услуги Push to Talk для расчета эффективности ее работы при разных нагрузках в сети.

В дипломной работе приведена архитектура IMS и для раскрытия темы взята одна из её услуг, услуга Push-to-Talk. Создана модель услуги Push-to-Talk и рассчитаны её вероятностно-временные характеристики. Сделано сравнения с результатами ручных расчетов и результатов моделирования.

Ключевые слова: IP Multimedia Subsystem (IMS), Push-to-Talk, модель, GPSS, услуги, расчет, результаты, моделирование, пользователь.

Дипломная работа содержит 84 страниц, 23 рисунков, 6 таблиц, 8 источников и 1 приложение.

Содержание

РЕФЕРАТ	3
ВВЕДЕНИЕ	6
ГЛАВА 1. Инфокоммуникационные услуги как объект моделирования	
1.1 Специфика предоставления мультимедийных услуг в сетях мобильной связи	7
1.2 Доступа к услугам в пакетной коммутации.....	9
1.3 Архитектура IMS и услуги реального времени	11
ГЛАВА 2. Обзор методов и средств математического моделирования случайных процессов	
2.1 Основы компьютерного моделирования случайных процессов	23
2.2 Специализированные языки моделирования случайных процессов	27
2.3 Выбор языка моделирования.....	31
2.4 Программирование на языке GPSS для моделирования услуг IMS	32
2.4.1 Пользовательский интерфейс программы моделирования на GPSS.....	32
2.4.2 Основные правила и операторы языка GPSS.....	39
2.4.3 Создания моделей в GPSS	45
ГЛАВА 3. Алгоритм услуги Push-to-Talk в архитектуре IMS	
3.1 Технология Push to Talk.....	50
3.2 Поддержка возможности Push-to-Talk по сотовой связи.....	52
3.3 Использование в IP сетях.....	54
ГЛАВА 4. Расчет вероятностно-временные характеристик услуги Push-to-Talk	57

ГЛАВА 5. Разработка программной модели услуги Push-to-Talk на GPSS	
5.1 Модель услуги Push-to-Talk на GPSS.....	62
5.2 Основной программный код.....	64
5.3 Результаты.....	67
ГЛАВА 6. Сравнение результатов расчета и моделирования услуги Push-to-Talk.....	71
ЗАКЛЮЧЕНИЕ.....	73
СПИСОК СОКРАЩЕНИЙ	74
СПИСОК ЛИТЕРАТУРА	76
ПРИЛОЖЕНИЕ 1: ЛИСТИНГ ПРОГРАММЫ	77

ВВЕДЕНИЕ

В настоящее время операторы сетей мобильной связи поколения 2.5G сталкиваются с тенденцией снижения темпов роста доходов, получаемых в результате предоставления услуг мобильной телефонии и простых услуг обмена сообщениями. Данная тенденция развивается под воздействием нескольких факторов. Снижение темпов роста доходов связано со снижением темпов роста абонентской базы операторов, так как рынок услуг мобильной связи постепенно переходит в состояние насыщения, то есть дальнейшее привлечение абонентов возможно только за счет перехода абонентов от одного оператора к другому, но не за счет подключения новых абонентов.

На протяжении последних лет основным средством «переманивания» абонентов от оператора к оператору служила политика демпинга цен на услуги, которая напрямую снижает показатель ARPU (average revenue per user) – средней выручки на одного пользователя мобильной связи.

Помимо традиционных услуг мобильной телефонии (голос, обмен короткими сообщениями), портфели операторов, как правило, включают в себя дополнительные услуги *добавочной стоимости (VAS – Value Added Services)*, с предоставлением которых операторы связывают дальнейшее успешное ведение бизнеса.

Хотя спрос на такие услуги на сегодняшний день не является значительным и не играет решающей роли в сложившейся ситуации, а сами предоставляемые услуги являются достаточно сложными как с точки зрения их подключения, так и с точки зрения использования, будущее именно за ними. Поэтому тема дипломной работы, посвященной моделированию услуг IMS является безусловно актуальной.

ГЛАВА 1. ИНФОКОММУНИКАЦИОННЫЕ УСЛУГИ КАК ОБЪЕКТ МОДЕЛИРОВАНИЯ

1.1 Специфика предоставления мультимедийных услуг в сетях мобильной связи

На протяжении двух последних десятилетий сети мобильной связи эволюционировали в целях удовлетворения и создания рыночного спроса на телекоммуникационные услуги. На заре эры мобильной связи этот спрос ограничивался простыми голосовыми сервисами, предоставляемыми, как правило, бизнес сегменту рынка. Затем произошел ощутимый скачок в распространении мобильной связи, так как ограниченное использование услуг мобильной телефонии бизнес сегментом переросло во всемирный рынок, породивший спрос на голосовые сервисы и простые текстовые сервисы передачи данных.

Сегодня, после первых пробных шагов в направлении мультимедийных мобильных услуг, появился спрос на услуги, аналогичные предоставляемым в сети Интернет пользователям с фиксированным доступом. Однако в мобильных сетях предоставление данных услуг должно удовлетворять определенному количеству условий. Спрос на новые услуги будет существовать только в том случае, если они будут цельными и персонализированными. Они должны быть интуитивны в вопросах доступа и использования. То есть, в настоящее время потребители желают получить сложные персонифицированные сервисы, предоставляемые в любое время любым способом через любое устройство.

IMS предоставляет возможность удовлетворить данный спрос простым способом, обеспечивающим быстрое и гибкое создание услуг, снижение стоимости их разработки и использования. Несмотря на то, что IMS оказывает радикальное воздействие на наборы предоставляемых

сервисов и бизнес модели операторов мобильной связи, как этап развития сетей она должна рассматриваться в качестве эволюционного, а не революционного шага.

Для того чтобы понять, какое место занимает IMS в терминах развития сервисов и технологий, полезно вспомнить процесс эволюции мобильных сетей в целях удовлетворения потребительского спроса.

Эволюция сети доступа

Эффективное использование полосы пропускания является одним из главных принципов операторов беспроводной связи и инженеров, разрабатывающих сети. Неэффективное использование полосы пропускания ведет к уменьшению трафика, переносимого сетью, уменьшение трафика ведет к уменьшению пользователей и уменьшению отдачи от использования инвестируемых средств.

На протяжении последних лет GSM операторы прикладывали все усилия на увеличение эффективности своих сетей, для того чтобы справиться с растущим числом абонентов, одновременно пытаясь найти способ предоставления требовательных к полосе пропускания Интернет-подобных услуг, от которых зависит их будущее.

Одним из способов усовершенствования работы мобильной сети является улучшение технологии сети доступа. Эта часть архитектуры беспроводной сети, предоставляющая связь между устройством пользователя и ядром сети.

Технология доступа UMTS W-CDMA передает цифровую информацию через широкополосный CDMA радиоканал. При данной технологии большое число пользователей могут одновременно использовать ресурсы одного широкополосного канала. Кроме того, использование более широкой полосы пропускания предоставляет более широкий канал переноса информации, который недостижим в распространенной сейчас технологии GSM. W-CDMA технология доступа обеспечивает операторам более широкую полосу

пропускания, требуемую для предоставления насыщенных контентом приложений.

1.2 Доступа к услугам в пакетной коммутации

Не менее важным шагом в эволюции сетей к сетям третьего поколения был переход от коммутации каналов к коммутации пакетов в ядре сети. Это технология, отвечающая за доставку и направление трафика в другие части сети. До появления технологии GPRS GSM сети были построены исключительно с использованием технологии коммутации каналов.

Технология коммутации каналов подразумевает, что при установлении соединения между вызывающим и вызываемым абонентами устанавливается выделенный канал связи. Основной недостаток данной технологии состоит в том, что данный канал становится недоступным для использования кем-либо еще на протяжении всего времени разговора.

Технология пакетной коммутации, напротив, работает по принципу расщепления трафика (голоса или данных) в набор контейнеров, называемых пакетами. Впоследствии эти пакеты передаются вместе с пакетами других пользователей через один канал связи, затем они перераспределяются и собираются воедино уже в устройстве пользователя, которому они предназначены.

Такое совместное использование ресурсов вместе с возможностью использования обычных сетевых протоколов и инфраструктуры ведет к более эффективной системе передачи информации и повышению удобства пользователя. Для GSM операторов появление технологии GPRS также означало, что их сети поддерживали индустриальный стандартный транспорт, такой как IP.

Пакетная коммутация более удобна для пользователя, так как она обеспечивает возможность предоставления так называемых «всегда – на - связи» услуг, используемых на обычных персональных компьютерах, подключенных к Интернет.

После того, как соединение установлено, пользователи получают постоянное виртуальное соединение с сетью, которое позволяет быстро принять или передать информацию, если таковая имеется. Поскольку технология коммутации пакетов использует более эффективные технологии маршрутизации, сети с коммутацией пакетов предоставляют большие скорости передачи данных по сравнению с сетями коммутации каналов. Кроме того, домен с коммутацией пакетов обеспечивает возможность использования более одной услуги в рамках одной сессии.

Например, в процессе голосового вызова через сеть пакетной коммутации возможно передать картинку или видеоклип от абонента к абоненту.

Стандарты 3GPP

Стандарты IMS определяют сетевой домен, предназначенный для управления мультимедийными сервисами и их интеграции. Данный домен является частью архитектуры сети третьего поколения, описанной в пятом выпуске стандартов 3GPP.

Первая волна стандартов 3GPP, определяющих большинство существующих на данный момент сетей третьего поколения была завершена в 1999 году и известна под названием «выпуск 99».

Эти стандарты представляли новый 3G интерфейс (W-CDMA) для ядра 2G сетей. Два года спустя 3GPP выпустила серию стандартов улучшения радио доступа под названием «выпуск 4».

В марте 2002 года 3GPP выпустила пятую серию стандартов, которые описывали технологию HSPDA, улучшающую характеристики нисходящего потока и главное усовершенствование окружения сервисов – стандарты IMS.

Хотя 3GPP был ответственным за написание большинства стандартов IMS, IMS является системой, построенной на основе протоколов, разработанных IETF. С того момента, как 3GPP определила понятие IMS, другие участники индустрии, такие как 3GPP2 и ETSI внесли большой вклад

в ее развитие. Несмотря на то, что IMS была разработана для изменения и эволюции UMTS сети, спецификации были детально разработаны как не зависящие от технологии доступа. Другими словами, IMS поддерживает множество способов доступа – GSM, W-CDMA, CDMA2000, wireline broadband access, WiMAX, WLAN. Одной из целей создателей IMS было создание архитектуры гибкого сервисного домена, независимого от уровня доступа.

1.3 Архитектура IMS и услуг реального времени

Состав ядра IMS и окружения

В 3GPP Release 5 были сформулированы функциональные требования к архитектуре IMS:

- организация мультимедийных сессий на IP транспорте;
- механизм согласования QoS;
- поддержка взаимодействия с Интернет и сетями с коммутацией каналов;
- поддержка роуминга;
- осуществление контроля доставки сервисов конечным абонентам;
- быстрый процесс разработки новых сервисов не требующий дополнительной стандартизации;
- инвариантность к методу доступа.

Эти требования нашли своё отражение в трёхуровневой архитектуре IMS. Идея трёхуровневой модели перекликается с концепцией сетей нового поколения NGN, разработанной ITU (International Telecommunication Union). В IMS выделяются (сверху вниз):

- Уровень приложений;
- Уровень управления вызовами;
- Транспортный уровень.

Все три уровня независимы, но связаны друг с другом через стандартизированные интерфейсы при использовании открытых протоколов.

Следует отметить, что 3GPP стандартизирует не конкретные узлы сети, а функциональные элементы и интерфейсы между ними. Разработчики конечных решений вправе реализовать функционал нескольких элементов в одном устройстве, либо наоборот разделить функциональный элемент и воплотить его в виде кластера.

Основные функциональные элементы IMS могут быть условно разделены на следующие группы:

- Элементы управления установлением сессий (CSCFs);
- Базы данных (HSS, SLF);
- Элементы, обеспечивающие предоставление дополнительных видов обслуживания (application server, MRFC, MRFP);
- Элементы обеспечения межсетевое взаимодействия (BGCF, MGCF, IMS-MGW, SGW);
- Элементы обеспечения безопасности (PDF, SEG, THIG);
- Элементы обеспечения тарификации

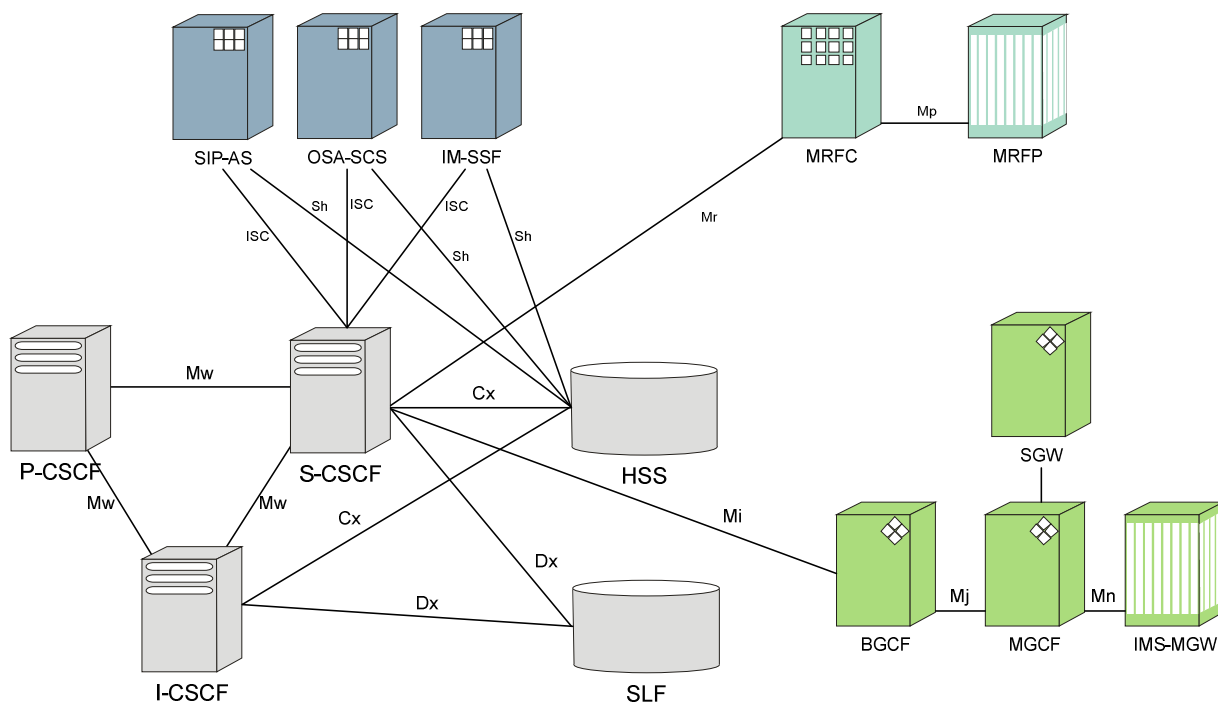


Рис 1. Архитектура IMS

Элементы управления сессиями - Call Session Control Function (CSCF)

Важнейшим функциональным элементом IMS являются Call Session Control Functions (CSCF's) – набор функциональных элементов управления установлением, изменением и прекращением мультимедийных сессий. По своей сути это SIP сервер производящий обработку сигнальной информации. Разделение на отдельные элементы обусловлено конкретным функционалом того или иного CSCF. Выделяется 3 типа CSCF:

- P-CSCF;
- S-CSCF;
- I-CSCF.

▪ P-CSCF

Proxy-CSCF – это первая точка контакта IMS терминала с ядром сети на сигнальном уровне и выполняет роль прокси сервера. P-CSCF привязывается к терминалу на этапе регистрации и не меняется на весь период действия регистрации. P-CSCF связан с S-CSCF и I-CSCF через *Mw* интерфейс и с GGSN (Gateway GPRS Support Node) через интерфейс *Gm*, основанные на протоколе Diameter для обмена управляющей информацией.

P-CSCF выполняет ряд функций:

- Установление IPsec связей. Установление этих связей обеспечивает уверенность в сохранности содержания сообщения на пути от терминала к P-CSCF, т.е. позволяет убедиться что оно не было изменено по пути следования;
- Аутентификация пользователя на P-CSCF позволяет избежать проведения этой процедуры на всех остальных элементах сети, которые «доверяют» P-CSCF;
- P-CSCF проверяет SIP сообщения на предмет правильности формата и отбрасывает те, что сформированы не по правилам;

- Компрессия/декомпрессия SIP сообщений позволяет уменьшить задержки при передаче по радио интерфейсу. SIP является текстовым протоколом и сообщения могут составлять значительные объёмы при сравнении с сообщениями бинарных протоколов;
- P-CSCF генерирует информацию необходимую для тарификации абонента .

P-CSCF может размещаться как в домашней для абонента сети, так и в сети другого оператора при роуминге.

▪ S-CSCF

S-CSCF (Serving Call Session Control Function) - центральный элемент ядра, производящий обработку сигнальной информации. S-CSCF производит маршрутизацию SIP сообщений и выполняет роль SIP сервера регистрации, то есть устанавливает соответствие между текущим месторасположением абонента (имеется ввиду IP-адрес и порт) и записью в базе данных с присущим набором доступных сервисов. Связь с базой данных HSS организуется через *Cx* и с SLF через *Dx* интерфейсы, через *Mw* интерфейс с P-CSCF и I-CSCF производится обмен управляющей информацией, для управления Breakout Gateway Control Function используется *Mi* интерфейс, управление Media Resource Function Controller производится через *Mr* интерфейс. Описание этих интерфейсов остаётся за рамками моей работы, отметим только, что реализованы они при использовании протокола Diameter.

S-CSCF выполняет ключевые функции при обслуживании абонента:

- Управляет процессом регистрации. S-CSCF при получении запроса на регистрацию контактирует с базой данных HSS и генерирует проверочную последовательность, которая отсылается пользователю. Пользовательский терминал генерирует ответ на основании полученной последовательности и секретного ключа известного только

абоненту и хранящемуся кроме этого в профиле абонента в базе данных. При получении ответа от пользователя, S-CSCF сравнивает полученную последовательность с той, что была сгенерирована на основании проверочной и ключа из базы данных, при положительном результате, пользователь авторизуется и может получить доступ к набору услуг, закреплённому за данной записью в базе данных.

- Маршрутизирует SIP сообщения. Во время регистрации S-CSCF загружает из базы данных Service Profile, в котором хранится набор правил, на основании которых S-CSCF осуществляет маршрутизацию всех сообщений. Основными моментами этой маршрутизации может быть направление сообщения по умолчанию в соответствии с внутренней логикой S-CSCF, либо на определённый элемент сети с целью инициировать предоставление той или иной услуги.
- Управляет состоянием соединения. S-CSCF может как создавать соединения, так и отдавать команды на его разрушение, например при получении информации от биллинговой системы S-CSCF может рассчитать допустимую длительность сессии и послать команду на закрытие соединения при недостаточности средств для продолжения сессии.
- Участвует в тарификации абонента. S-CSCF производит различные действия по взаимодействию с биллинговой системой в зависимости от метода тарификации, способа организации сбора тарификационной информации и пр.

▪ I-CSCF

Interrogating Call Session Control Function выполняет роль контактной точки при меж операторском взаимодействии, при этом I-CSCF выполняет ряд характерных для этой роли функций:

- Определяет адрес следующего элемента для пересылки сообщения основываясь на информации из базы данных HSS. При получении сообщения из сети другого оператора I-CSCF должен определить адрес S-CSCF, обслуживающего вызываемого абонента. При вызове инициированном в родной сети, I-CSCF должен определить адрес сети назначения и передать запрос.
- Реализует функцию Topology Hiding Inter-network Gateway (THIG), для сокрытия топологии сети оператора.

Для выполнения этих функций I-CSCF имеет связи с базой данных HSS и SLF через интерфейс *Cx* и *Dx* соответственно, с S-CSCF и P-CSCF через интерфейс *Mw*.

База данных – Home Subscribers Server (HSS)

Home Subscriber Server является базой данных, хранящей всю информацию имеющую отношение к пользователю. Технически она является аналогом HLR (Home Location Registrar) в сетях GSM. HSS представляет сетевым элементам необходимую информацию через *Cx* интерфейс основанный на протоколе Diameter. Кроме информации о пользователях HSS хранит информацию о сервисах, которые объединяются в так называемые сервисные профили (Service Profile). Сервисный профиль содержит набор правил, по которым инициируется предоставление той или иной услуги.

▪ SLF

В сети оператора, в зависимости от ёмкости и производительности, может находиться некоторое число HSS, но следует учитывать что дублированные с целью надежного хранения данных HSS логически рассматриваются как одна. В любом случае информация, относящаяся к определённому абоненту, хранится только в одной HSS. Для определения HSS, в которой хранится информация о данном пользователе, используется

Subscription Locator Function (SLF). Для сетей с одной HSS в использовании SLF нет необходимости.

Элементы, обеспечивающие предоставление дополнительных видов обслуживания

Следует отметить, что инициацией дополнительных видов обслуживания практически всегда занимается S-CSCF, т.е. основываясь на анализе входящих сообщений и данных из Service Profile, к которому привязан текущий идентификатор конкретного пользователя, S-CSCF принимает решение перенаправить сообщение на тот или иной элемент, реализующий ДВО. К таким элементам относятся Application Server (AS) и Media Resource Function (MRF).

- Application Server

Сервер приложений это элемент, на котором хранится приложение, реализующее ДВО, и логика для его выполнения. С точки зрения S-CSCF все серверы приложений являются SIP серверами. Это один из важнейших принципов IMS, позволяющий подключать серверы приложений, реализованные на различных технологиях, по единому правилу и через единый интерфейс – IMS Service Control (ISC) интерфейс, основанный на SIP.

- MRF

Media Resource Function (MRF) является источником мультимедиа потоков в домашней сети. MRF реализует следующие функции:

- является источником мультимедийных потоков (например для проигрывания уведомлений);
- смешивает мультимедиа потоки (в случае централизованной мультимедиа конференции);

- производит обработку мультимедийных потоков (перекодирование между кодеками, сбор статистики).

MRF логически разделяется на Media Resource Function Controller (MRFC) и Media Resource Function Processor (MRFP). MRFC реализует взаимодействие с S-CSCF на сигнальном уровне посредством протокола SIP, а MRFP производит манипуляции с медиа потоками. MRFC осуществляет управление MRFP по протоколу H.248.

Элементы обеспечения межсетевого взаимодействия

В архитектуре IMS предусмотрено 4 элемента для взаимодействия с сетями ТфОП:

- BGCF

Breakout Gateway Control Function (BGCF) используется в случаях, когда источником вызова является IMS терминал, а в качестве вызываемого выступает терминал, находящийся в сети ТфОП или терминал абонента мобильной сети с коммутацией каналов, например GSM. Основными функциями BGCF являются:

- выбор нужной сети, в которой будет произведено взаимодействие с сетью ТфОП или GSM;
- или выбор нужного шлюза ТфОП/GSM, если это взаимодействие происходит в той же сети, где находится и BGCF.

На BGCF, при выборе сети, в которой будет производиться взаимодействие с сетью с коммутацией каналов, может быть возложена функция, результатом которой будет выбор наиболее дешёвого маршрута. Логично, что чем менее протяжённый путь проходит вызов по участку сети с коммутацией каналов, тем более дешёвым будет это соединение. Таким образом, даже при наличии ТфОП шлюза в своей сети, бывает выгоднее воспользоваться услугами шлюза другого оператора.

Шлюз к сетям с коммутацией каналов предоставляет возможность абонентам IMS сети устанавливать соединения с абонентами сетей ТфОП и GSM, или любой другой сети с коммутацией каналов. IMS наследует идею о декомпозиции шлюза из сетей H.323 и Softswitch. Рисунок описывает способ декомпозиции шлюза в IMS:

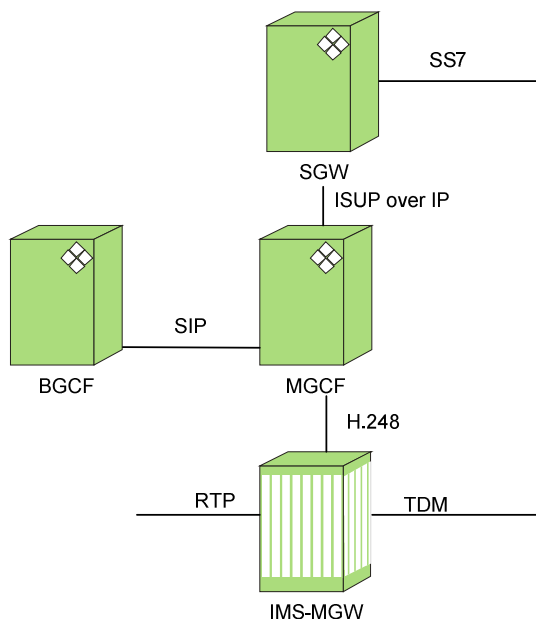


Рис 2. Логическая структура декомпозиции шлюза в IMS

Декомпозированный шлюз распадается на три составляющие:

- **SGW**

Signaling Gateway (SGW) – сигнальный шлюз, производит взаимодействие с ТфОП на уровне сигнализации. SGW производит конвертацию ISUP или BICC поверх MTP в ISUP или BICC поверх SCTP/IP.

- **MGCF**

Media Gateway Control Function (MGCF) – является контроллером шлюза, выполняет функции по транслированию протокола SIP в ISUP или BICC поверх SCTP/IP. И осуществляет контроль медиа-поток в IMS-MGW.

- IMS-MGW

IP Multimedia Subsystem Media Gateway (IMS-MGW) – транспортный шлюз, взаимодействует с ТфОП или любой другой сетью с коммутацией каналов на уровне потоков медиа информации, транспортном уровне. Задачей IMS-MGW является преобразование потока данных, поступающих по ИКМ линии, в данные, упакованные и переносимые протоколом RTP. Так же IMS-MGW производит перекодирование между различными кодеками используемыми в IMS и в сетях с коммутацией каналов. Например, как правило на ТфОП используется кодек G.711, который, с целью уменьшить объем передаваемых данных, конвертируется в кодек применяющий различные алгоритмы компрессии.

Элементы обеспечения безопасности

К этой группе условно отнесены элементы обеспечивающие выполнение функций по обеспечению безопасности (SEG), сокрытию топологии (THIG) и реализации контроля определённых оператором политик в отношении принятых на сети форматов передаваемой медиа информации (PDF).

- PDF

Во время сессии абоненты IMS сети обмениваются тем или иным видом медиа информации. Контроль за тем какой кодек используется возложен на policy decision function (PDF). PDF использует информацию прокола SDP передаваемую в теле сообщения SIP и полученную у P-CSCF. Так в SDP для каждой устанавливаемой сессии описывается кодек, который UA собирается использовать во время сессии. Политика оператора может быть направлена на оптимизацию объёма передаваемой медиа информации и

может выражаться в запрете использования некоторых кодеков, обеспечивающих недостаточное сжатие данных.

Кроме политики в отношении всех пользователей на сети, PDF может реализовывать политики в отношении конкретных пользователей. Так для некоторых пользователей может быть запрещена возможность установления видео сессий. Стоит прояснить роль PDF в процессе этого ограничения. В случае, когда UA пытается установить, например, видео сессию, что для него запрещено, S-CSCF, через соответствующий абоненту P-CSCF, направит ответ 488 Not Acceptable Here, тем самым проинформировав UA о невозможности использования запрашиваемого кодека. В случае когда предлагаемый со стороны UA кодек принят к использованию на сети, необходимо проинформировать точку доступа к сети на транспортном уровне о характеристиках создаваемой сессии. Такой точкой в мобильных сетях 3G является GGSN (General GPRS Support Node). Происходит обмен информацией между PDF и GGSN, конечным итогом которой является установление параметров для создаваемой сессии, включающей в себя информацию об IP адресе и порте источника, применяемом кодеке и заданных параметрах качества обслуживания (QoS). Как правило, PDF при реализации объединяется с P-CSCF в единый модуль.

- SEG

Security gateway (SEG) – шлюз, обеспечивающий безопасность на уровне сигнализации. SEG, как правило, устанавливается на границе административного домена оператора, формируя тем самым «безопасный домен». SEG, контактируя с другими «безопасными» доменами для пересылки информации, реализует так называемую модель пошаговой безопасности. SEG может использовать методы фильтрации пакетов и/или выполнять функции фаервола. Функциональность SEG важна при меж

операторском обмене, поэтому при практической реализации его функции реализуются в едином модуле с I-CSCF.

- THIG

Использование Topology Hiding Gateway (THIG) обусловлено желанием оператора скрыть топологию, ёмкость и конфигурацию его внутренней сети от других операторов и третьих лиц. Для этого THIG должен быть установлен на границе административного домена оператора и производить обработку проходящей через него информации на пример удаления (шифрования) заголовков, содержащих конфиденциальную с точки зрения оператора информацию. По аналогии с SEG реализация THIG совмещается с исполнением I-CSCF.

Элементы обеспечения тарификации

Система IMS предоставляет оператором очень удобные средства для тарификации, позволяя разделить списания средств в зависимости от предоставляемого сервиса, т.е. телефонии, видеосвязи, передачи данных или ДВО. Не вдаваясь в подробности и не перечисляя конкретных функциональных элементов, отметим лишь что, в архитектуре IMS предусмотрена поддержка он-лайн и офф-лайн моделей тарификации различными функциональными модулями. Главной задачей этих модулей является сбор тарификационной информации от сетевых элементов и преобразование этой информации в вид, требуемый биллинговой системой.

ГЛАВА 2. ОБЗОР МЕТОДОВ И СРЕДСТВ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ СЛУЧАЙНЫХ ПРОЦЕССОВ

2.1 Основы компьютерного моделирования случайных процессов

Основные понятия и определения моделирования

Моделирование — метод научного исследования явлений, процессов, объектов, устройств или систем (обобщенно – объектов исследований), основанный на построении и изучении моделей с целью получения новых знаний, совершенствования характеристик объектов исследований или управления ими.

Модель — материальный объект или образ (мысленный или условный: гипотеза, идея, абстракция, изображение, описание, схема, формула, чертеж, план, карта, блок-схема алгоритма, ноты и т.п.), которые упрощенно отображают самые существенные свойства объекта исследования.

Любая модель всегда проще реального объекта и отображает лишь часть его самых существенных черт, основных элементов и связей. По этой причине для одного объекта исследования существует множество различных моделей. Вид модели зависит от выбранной цели моделирования.

В основе термина «модель» лежит латинское слово *modulus* — мера, образец. Модель – это заместитель реального объекта исследования. Модель всегда проще исследуемого объекта. При изучении сложных явлений, процессов, объектов не удастся учесть полную совокупность всех элементов и связей, определяющих их свойства.

Но все элементы и связи в создаваемой модели и не следует учитывать. Нужно лишь выделить наиболее характерные, доминирующие составляющие, которые в подавляющей степени определяют основные

свойства объекта исследования. В результате объект исследования заменяется некоторым упрощенным подобием, но обладающим характерными, главными свойствами, аналогичными свойствам объекта исследования. Появившийся вследствие проведенной подмены новый объект (или абстракция) принято называть моделью объекта исследования.

Виды моделей

Создать исчерпывающую классификацию моделей достаточно сложно, поэтому рассмотрим наиболее часто употребляемые определения моделей.

Концептуальная модель (содержательная) — это абстрактная модель, определяющая структуру системы (элементы и связи).

В концептуальной модели обычно в словесной (вербальной) форме приводятся самые главные сведения об объекте исследования, основных элементах и важнейших связях между элементами. Процесс создания концептуальной модели в настоящее время не формализован: не существует точных правил ее создания.

Основная проблема при создании концептуальной модели заключается в нахождении компромисса между компактностью модели и ее точностью (адекватностью). Имеется множество теоретических проработок этой проблемы, но их трудно применить для решения каждой новой задачи. Поэтому разработчик модели, руководствуясь своими знаниями, оценочными расчетами, опытом, интуицией, мнением экспертов, должен принять решение об исключении какого-либо элемента или связи из модели, изъятии из рассмотрения второстепенных факторов, воздействующих на объект.

Термин «адекватна» (происходит от лат. *adaequatus* — приравненный, равный) означает верное воспроизведение в модели связей и отношений объективного мира. Этим термином характеризуют качество созданной модели.

Процесс создания концептуальной модели, вероятно, никогда не сможет быть полностью формализован. Трудно придумать набор простых правил, выполняя которые, можно создать хорошую концептуальную модель. Именно в связи с этим иногда говорят, что моделирование является не только наукой, но и искусством.

Концептуальную модель, содержащую основные сведения об объекте исследований, порой называют информационной моделью.

Математическое моделирование — метод изучения объекта исследования, основанный на создании его математической модели и использовании её для получения новых знаний, совершенствования объекта исследования или управления объектом.

В научной литературе широко используется термин математическая модель (ММ). ММ — описание объекта исследования, выполненное с помощью математической символики.

Для составления ММ можно использовать любые математические средства — дифференциальное и интегральное исчисления, регрессионный анализ, теорию вероятностей, математическую статистику и т. д. Математическая модель представляет собой совокупность формул, уравнений, неравенств, логических условий и т.д. Используемые в ММ математические соотношения определяют процесс изменения состояния объекта исследования в зависимости от его параметров, входных сигналов, начальных условий и времени. По существу, вся математика создана для формирования математических моделей.

Математическое моделирование можно подразделить на аналитическое и компьютерное (машинное) моделирование.

При аналитическом моделировании ученый — теоретик получает результат «на кончике пера» в процессе раздумий, размышлений, умозаключений. Формирование модели производится в основном с помощью точного математического описания объекта исследования.

При компьютерном моделировании математическая модель создается и анализируется с помощью вычислительной техники. В этом случае нередко используются приближенные (численные) методы расчета. При компьютерном моделировании используются наиболее прогрессивные информационные технологии, например, виртуальная реальность. При этом моделирование медицинской операции вызывает иллюзию реально происходящего события. Моделирование игровых ситуаций сопровождается мультимедийными эффектами (звуками, видеоэффектами).

Компьютерная модель – модель, реализованная на одном из языков программирования (программа для ЭВМ).

Преимущества использования моделирования

Используя моделирование при проектировании или реинжиниринге вычислительной системы, мы можем сделать следующее:

- оценить пропускную способность сети и ее компонентов, определить узкие места в структуре вычислительной системы;
- сравнить различные варианты организации вычислительной системы;
- осуществить перспективный прогноз развития вычислительной системы;
- предсказать будущие требования по пропускной способности сети, используя данные прогноза;
- оценить требуемое количество и производительность серверов в сети;
- сравнить различные варианты модернизации вычислительной системы;
- оценить влияние на вычислительную систему модернизации ПО, мощности рабочих станций или серверов, изменения сетевых протоколов.

Исследование параметров вычислительной системы при различных характеристиках отдельных компонентов позволяет выбрать сетевое и вычислительное оборудование с учетом производительности, качества обслуживания, надежности и стоимости.

Поскольку стоимость одного порта активного сетевого оборудования в зависимости от производителя оборудования, используемой технологии, надежности, управляемости может меняться от десятков рублей до десятков тысяч, моделирование позволяет минимизировать стоимость оборудования, предназначенного для использования в вычислительной системе.

Моделирование становится эффективным при числе рабочих станций 50-100, а когда их более 300, общая экономия средств может составить 30--40% от стоимости проекта.

2.2 Специализированные языки моделирования случайных процессов

Существует немало компьютерных программ моделирования, предназначенных для выполнения на компьютерах различных типов - от микрокомпьютеров до "мэйнфреймов" (больших ЭВМ). Как же из столь внушительного перечня выбрать самую подходящую программу?

Прежде всего нужно понимать, что существуют различные типы моделирования. Затем, внимательно изучив программы, которые предлагает рынок, нужно найти такую, которая соответствует вашим конкретным потребностям. Если же подходящей программы не удастся найти, иногда не так уж сложно создать собственную специализированную программу моделирования. Тем более, что такая программа будет полностью соответствовать вашим потребностям и с ее использованием у вас наверняка не возникнет проблем.

Для имитационного моделирования используются как универсальные, так и специальные языки. Универсальные языки дают большие возможности программисту в смысле гибкости разработки, отладки и использования модели. Однако это требует больших усилий, затрачиваемых на программирование. В этом случае усложняется порядок выполнения операций, отсчет системного времени и контроль. К универсальному программному обеспечению относятся языки программирования, которые позволяют программистам строить свои собственные модели. Примерами универсальных языков моделирования являются SLAM II, SIMSCRIPT II, 5, SI MAN, GPSS/H, GPSS/PC, PC-MODEL и RESQ.

Специализированный язык отличается от универсального своими специфическими свойствами. Специализированное программное обеспечение (например, MAP/1 и SIMFACTORY) предназначено для моделирования конкретных, специальных приложений. Например, в специализированном программном обеспечении для моделирования производства предусмотрены средства, с помощью которых можно указывать количество рабочих центров, их описания, интенсивность поступления, время обработки, размеры партий, объемы незавершенного производства, наличные ресурсы (в том числе трудовые), последовательности и т.д. Кроме того, такая программа нередко позволяет аналитику наблюдать производственный процесс в анимационном представлении и следить по ходу моделирования за количественными показателями и потоками в системе. Данные собираются, анализируются и представляются в форме, наиболее подходящей для приложения соответствующего типа.

К этим свойствам или требованиям можно отнести:

- * способность генерировать случайные числа;
- * возможность генерировать случайные величины и процессы;
- * возможность "продвигать" время либо на одну единицу Δt , либо до следующего события;
- * способность накапливать выходные данные;
- * способность проводить статистический анализ накапливаемых данных;
- * способность распределять выходные данные по заранее заданным форматам;
- * способность выполнять идентификацию конкретных событий.

Некоторые из языков имитационного моделирования являются и описательными языками. Они близки к естественному языку, поэтому имитационные модели, написанные на таком языке, легче воспринимаются руководителями и специалистами, не имеющими непосредственного отношения к программированию.

Специальные языки цифрового имитационного моделирования делятся на две группы, соответствующие двум видам имитации: дискретных и непрерывных процессов.

Для моделирования непрерывных процессов используются языки:

- Динамо - для аппроксимации непрерывных процессов используют дифференциальные уравнения первого порядка;
- CSMP, Midas и др., которые помимо блочного построения, применяемого в языке Динамо, еще обладают мощностью и удобством алгебраической и логической алгоритмизацией.

Языки для моделирования дискретных процессов можно разбить на четыре категории:

- 1) языки, ориентированные на действия (CSL, Форсим IV и др.);
- 2) языки, ориентированные на события (Симскрипт, Симком и др.);

- 3) языки, ориентированные на процессы (Симула, SOL);
- 4) языки, ориентированные на потоки сообщений (GPSS, BOSS).

В первой группе действия представлены в модельном времени как мгновенные. В этих языках нет регламентации действиям. Вместо этого исполнительные программы просматривают набор всех условий, от которых зависит появление какого-либо события. И только тогда, когда выполняются все контролируемые условия происходит изменение состояния и сдвиг времени в этой части программы.

Некоторые задачи удобнее программировать на языке ориентированном на события. При этом событие регламентировано. Регламентация обеспечивает наступление события именно в тот момент времени, когда динамическое состояние показывает, что сложились условия для его появления.

Языки, ориентированные на процессы, объединяют достоинства первых двух языков, т.е. краткость языков, ориентированных на действия и эффективность языков, ориентированных на события. Написанная на этом языке программа работает как несколько независимых программ: одна посредством просмотра действий, другая посредством регламентирования событий.

Что касается четвертой группы языков, то они на самом деле являются языками процессов и отличаются только схемным построением.

2.3 Выбор языка моделирования

Язык программирования GPSS – этот язык с 1968 года входит в математическое обеспечение машин фирмы IBM, один из наиболее популярных языков ИМ.

GPSS составлен из объектов и операций (логических правил). Объекты делятся на семь классов:

- динамические (ДО);
- аппаратно-ориентированные (АО);
- статические (СО);
- операционные (ОО);
- вычислительные (ВО);
- запоминающие (ЗО);
- группирующие (ГО).

ДО — элементы потока обслуживания заявки или "транзакты". Они создаются и уничтожаются, с каждым транзактом может быть связано некоторое число "параметров"

АО — соответствуют элементам оборудования, которые управляются ДО.

К ним относятся:

- накопители;
- устройства;
- логические переключатели.

СО:

- очереди;
- таблицы.

ЗО:

- ячейки;
- матрицы ячеек.

ГО:

- группы;
- списки.

ВО:

- арифметические и булевы переменные;
- функции.

Каждой очереди соответствует перечень транзактов, задержанных в какой-либо точке системы и запись длительности этих задержек: $T_3 = \{\tau_{i3}\}$. Таблицы могут использоваться для построения распределений выбранных величин.

ОО – блоки – формируют логику системы, давая транзактам указания, куда идти дальше.

Язык GPSS обычно выделяют в отдельную группу. Язык GPSS представляет собой интерпретирующую языковую систему, которую применяют для описания пространственного движения объектов.

2.4 Программирование на языке GPSS для моделирования услуг ИМ

2.4.1 Пользовательский интерфейс программы моделирования на GPSS

Запуск и начало работы в GPSS

Для запуска пакета GPSS в главном меню операционной системы Windows (кнопка «Пуск» или «START») необходимо выбрать приложения «GPSS World».

Создание новой модели.

Для создания новой ИМ необходимо в главном меню GPSS World выбрать пункт «File → New», затем в диалоговом окне выбрать режим создания новой модели. В рабочем окне GPSS World появится новое текстовое окно с заголовком «Untitled Model 1», в котором записывается текст создаваемой ИМ (Рис. 3).

Сохранение модели.

Для сохранения текста создаваемой ИМ необходимо в главном меню GPSS World выбрать пункт «File → Save» или «File → Save As» (Рис. 4). Во

втором случае пользователю будет предложено задать новое имя для файла с текстом ИМ. Файл с текстом модели в GPSS имеет расширение «gps».

Использование существующей модели.

Для открытия уже существующего файла с текстом ИМ необходимо в главном меню GPSS World выбрать пункт «File→ Open», затем указать имя необходимого файла в стандартном окне Windows. Три перечисленные операции с файлами могут инициироваться кнопками на панели инструментов GPSS World.

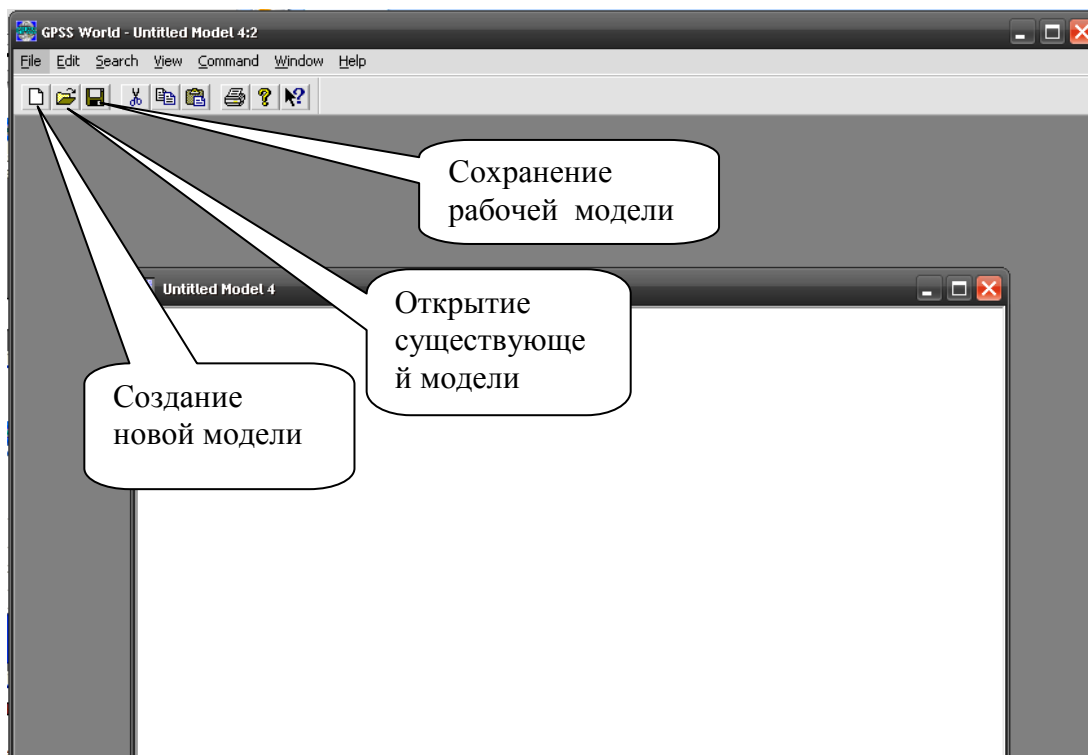


Рис.3 Окно GPSS World

После написания текста ИМ на языке GPPS модель необходимо *оттранслировать*, т.е. преобразовать последовательность текстовых операторов в последовательность команд, написанную в машинных кодах компьютера. Для этого необходимо выбрать пункт создания модели «Command Create Simulation» главного меню (Рис. 5)

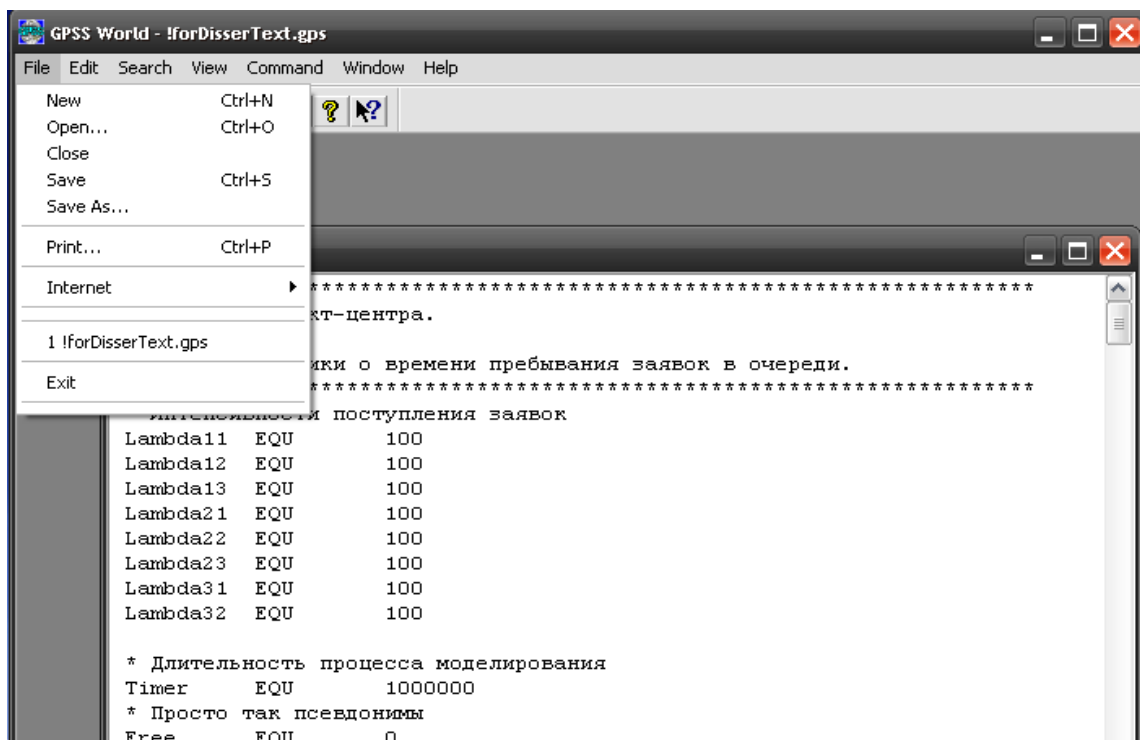


Рис. 4 Содержимое главного меню GPSS World.

Если при трансляции в тексте ИМ будут обнаружены некорректные директивы, значения параметров или синтаксические ошибки, то в появившемся окне документирования событий («JOURNAL») GPSS приведет список ошибочных строк. Для повторной трансляции модели после исправления ошибок используется пункт главного меню «Command» (Рис. 5).

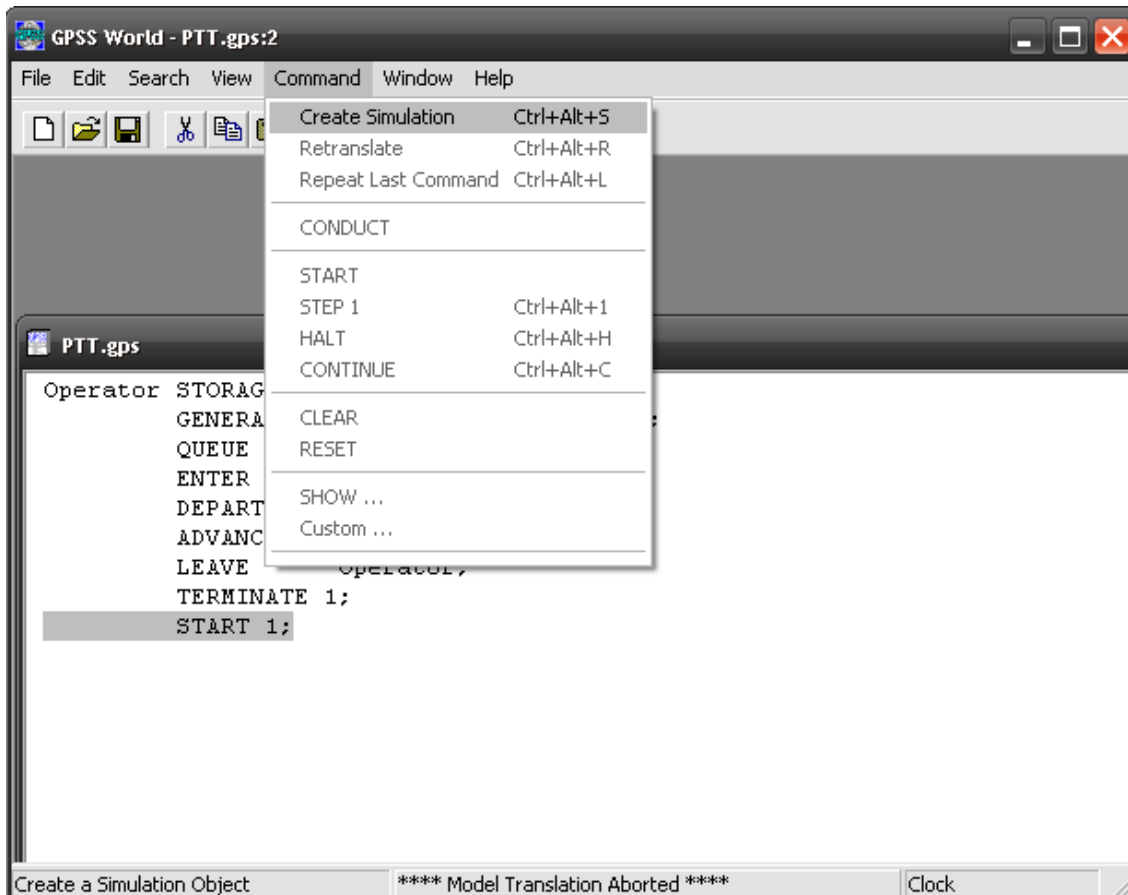


Рис. 5 Трансляция текста имитационной модели

Для запуска процедуры моделирования в конце текста ИМ следует задать команду «START A» или выбрать пункт главного меню «Command START» и в диалоговом окне указать начальное значение счетчика завершений *TGI*.

Средства отладки имитационных моделей.

GPSS имеет в своем составе развитые средства отладки ИМ, доступ к которым осуществляется из пункта главного меню «Window Simulation Window» (Рис. 6)

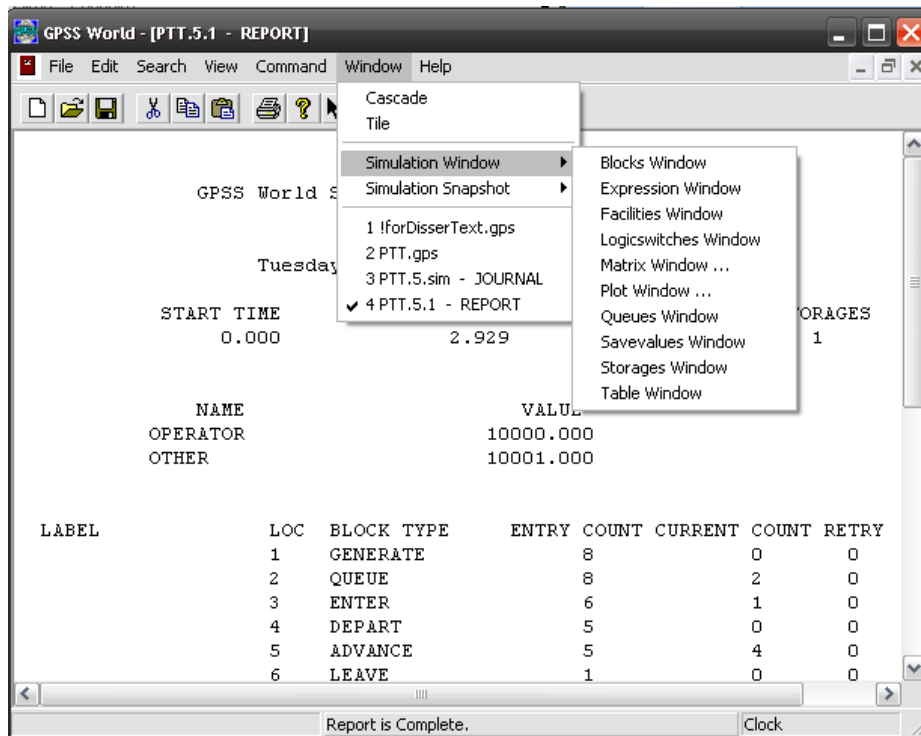


Рис. 6 Доступ к диалоговым окнам отладки ИМ

Анализ результатов моделирования в GPSS

После завершения процедуры моделирования GPSS автоматически открывает окно отчета «REPORT»(Рис. 7), содержащее стандартную информацию о результатах моделирования.

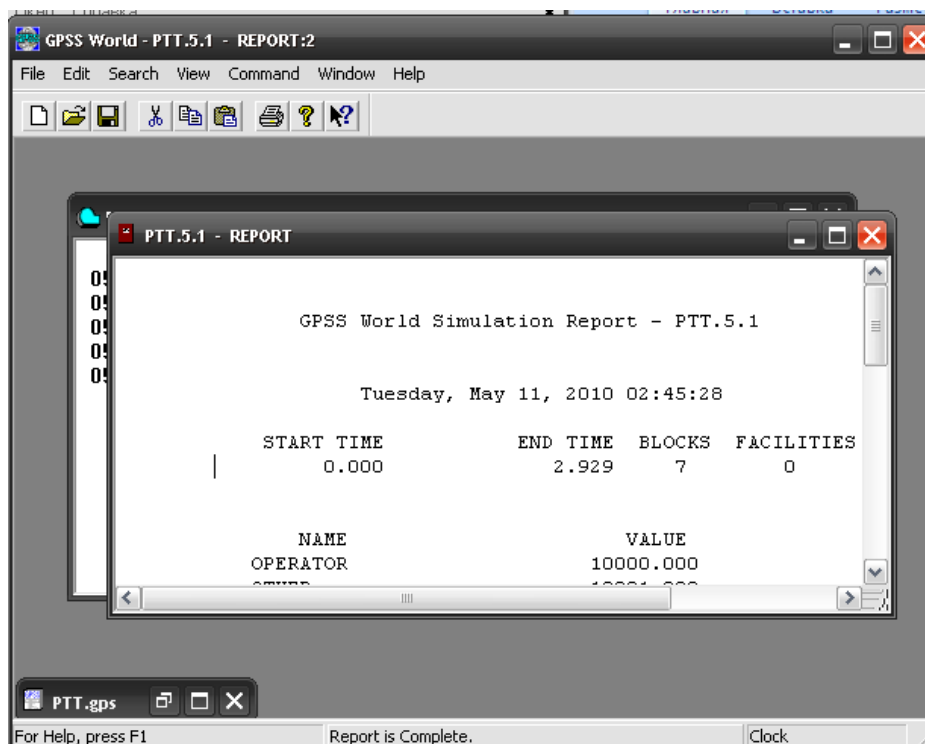


Рис. 7 Окно результатов моделирования в GPSS

Выходной файл статистики состоит из подразделов, содержащих стандартную статистику об объектах GPSS, используемых в данной модели. Начинается файл статистики с заголовка, который содержит имя модели, дату и время моделирования. Далее следует выходная информация, содержащая следующие основные сегменты вывода:

Информацию о результатах работы модели

START TIME	END TIME	BLOCKS	FACILITES	STORAGES
------------	----------	--------	-----------	----------

Здесь START TIME – абсолютное модельное время в момент начала моделирования;

END TIME – абсолютное время или момент, когда счетчик завершенный принял значение 0;

BLOCKS – количество блоков, использованных в текущей модели, к моменту завершения моделирования;

FACILITES – количество одноканальных устройств, использованных в модели к моменту завершения моделирования;

STORAGES – количество многоканальных устройств, использованных в модели к моменту завершения моделирования;

Информация об именах

NAME	VALUE
------	-------

Здесь поле NAME содержит имена, используемые в программе модели;

VALUE – определяет значение (номер), соответствующее имени, устанавливает начальный номер GPSS равным 10000.

Информация о блоках текущей модели

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
-------	-----	------------	-------------	---------------	-------

Здесь LABEL – метка оператора, связанного с блоком GPSS;

LOC – номер строки модели, связанной с блоком;

BLOCK TYPE – тип блока GPSS;

ENTRY COUNT – количество транзактов, вошедших в данный блок после последнего выполнения команд RESET или CLEAR или с начало процедуры моделирования;

CURRENT COUNT – количество транзактов, находящихся в данном блоке и ожидающих выполнения некоторых условий;

RETRY – количество транзактов, ожидающих выполнения некоторых условий;

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER	RETRY	DELAY
----------	---------	-------	--------------	--------	-------	------	-------	-------	-------

Здесь FACILITY – номер или имя одноканального устройства;

ENTRIES - количество транзактов, вошедших в данный блок после последнего выполнения команд RESET или CLEAR или с начало программы.

UTIL. – часть периода моделирования, в течение которого устройство было занято (коэффициент загрузки);

AVE. TIME – среднее время занятости устройства одним транзактом в течение процедуры моделирования после последнего выполнения команд CLEAR или RESET;

AVAILABLE – состояние готовности устройства в конце периода моделирования;

OWNER – номер последнего транзакта, занимавшего устройство (0 означает, что устройство не занималось);

PEND – количество транзактов, ожидающих устройство (находящееся в режиме прерывания);

INTER – количество транзактов, обработка которых прервана на устройстве в данный момент модельного времени;

RETEY – количество транзактов, ожидающих выполнения некоторых условий;

DELAY – количество транзактов, ожидающих занятия устройства (включая транзакты, ожидающие освобождение устройства в режиме прерывания).

2.4.2 Основные правила и операторы языка GPSS

Для описания имитационной модели на языке GPSS полезно представить ее в виде схемы, на которой отображаются элементы СМО - устройства, накопители, узлы и источники . Описание на языке GPSS есть совокупность операторов (блоков), характеризующих процессы обработки заявок. Имеются операторы и для отображения возникновения заявок, задержки их в ОА, занятия памяти, выхода из СМО, изменения параметров заявок (напрмер, приоритетов), вывода на печать накопленной информации, характеризующей загрузку устройств, заполненность очередей и т.п.

Каждый транзакт, присутствующий в модели, может иметь до 12 параметров. Существуют операторы, с помощью которых можно изменять значения любых параметров транзактов, и операторы, характер исполнения которых зависит от значений того или иного параметра обслуживаемого транзакта.

Пути продвижения заявок между ОА отображаются последовательностью операторов в описании модели на языке GPSS специальными операторами передачи управления (перехода). Для моделирования используется *событийный метод*. Соблюдение правильной временной последовательности имитации событий в СМО обеспечивается

интерпретатором GPSSPC - программной системой, реализующий алгоритмы имитационного моделирования.

Структура операторов GPSS

В записи оператора выделяют три части: метку, название, поле переменных.

Пример оператора:

```
110 L1 GENERATE 30,5          Первый сегмент модели
      2.....6.8.....18 ..19.....70
<метка> <название> <поле переменных>      <комментарии>
```

В поле переменных выделяют *подполя*, разделяемые при записи запятыми и служащие для указания чисел, стандартных числовых атрибутов (СЧА), символов, обозначающих метки, идентификаторов, указателей разновидностей операторов и т.п. Подполя могут быть пустыми. Возможна запись *комментария* после последнего непустого поля через пробел.

Стандартные числовые атрибуты служат для сокращенного указания различных величин, фигурирующих в модели.

Примеры СЧА: K126 - константа, равная 126; V2 - переменная N 2; Q4 - длина очереди N 4; X5 - хранимая величина N 5; FN7 - функция N 7; P4 - значение параметра N 4 транзакта; *6 - содержимое параметра N 6 транзакта; S*3 (или FN*3) - память (или функция), определенная в параметре N 3 транзакта.

Основные операторы языка GPSS

Основные операторы языка GPSS приведены в виде примеров с конкретными значениями подполей в поле переменных.

```
GENERATE 12,4,50,5,1
```

- генерация транзактов, интервалы времени между появлениями транзактов распределены равномерно в диапазоне [12-4, 12+4], первый транзакт

появится с задержкой в 50 единиц модельного времени, всего будет создано 5 транзактов, приоритет транзактов равен единице.

GENERATE 12,4,50,,1

- то же, но количество генерируемых транзактов неограничено.

GENERATE 6, FN\$FFF,50,5,1

- то же, но интервал времени между появлениями транзактов есть целая часть произведения числа 6 на значение функции FFF.

FNK FUNCTION RN1,C4

0,0/0.1,0.8/0.5,1.6/1.0,1.9

- описание функции FNK, ее аргументом является случайная величина (на это указывает значение RN1), равномерно распределенная в диапазоне [0,1], функция является непрерывной числовой (указатель C), заданной таблично четырьмя точками: (0;0), (0.1; 0.8), (0.5, 1.6), (1.0; 1.9).

FNK FUNCTION *2,D4

0,12/1,9/2,8/3,6

- то же, но аргументом является значение второго параметра транзакта, для которого вычисляется значение дискретной величины (D) числовой функции FNK, заданной таблично четырьмя узловыми точками. Это текущее значение округляется до ближайшего большего значения аргумента в узловой точке.

SEIZE PLOT

- занятие устройства PLOT приходящим на его вход транзактом; если устройство занято, то транзакт задерживается в очереди к этому устройству.

RELEASE PLOT

- освобождение устройства PLOT обслуженным транзактом.

ENTER MEM,12

- занятие транзактом 12 единиц емкости в накопителе MEM.

LEAVE MEM,*2

- освобождение k единиц памяти в накопителе MEM, гдк k - значение 2-го параметра транзакта.

STR STORAGE 4096

- описание накопителя STR емкостью 4096 единиц.

TERMINATE 3

- удаление транзакта из системы, при этом содержимое итогового счетчика уменьшается на 3 единицы, моделирование заканчивается, если содержимое счетчика станет равным или меньше нуля.

ADVANCE A,B

- задержка транзакта на время, определенное содержимым полей А и В, смысл величин, записываемых в этих подполях, такой же, как и в операторе GENERATE.

SPLIT 3,LLL,6

- копирование транзактов, в данном случае создаются три копии исходного транзакта, исходный транзакт направляется в следующий по порядку блок, а созданные копии - в блок с меткой LLL, при этом параметр 6 основного транзакта увеличивается на единицу, а транзактов - копий - на 2, 3, 4 соответственно.

ASSEMBLE 5

- объединение транзактов, первый из вошедших в блок транзактов продолжит движение в системе после того, как в блок придут еще четыре транзакта.

ASSIGN 2,NAP

- изменение параметров транзактов, в данном случае второй параметр транзакта получит значение NAP.

ASSIGN 3+,V4

- изменится значение третьего параметра транзакта - к нему прибавится значение V4.

TRANSFER ,MET

- безусловная передача управления оператору с меткой (номером) MET.

TRANSFER BOTH,LAB1,UNN

- переход к оператору с меткой LAB1, если он невозможен, то к оператору с меткой UNN, если и он невозможен, то транзакт задерживается до

следующего момента дискретного модельного времени, в который повторяются указанные попытки перехода.

TRANSFER .4,AAA,LAB

- транзакт с вероятностью 0.4 переходит к оператору с меткой LAB и с вероятностью 0.6 к оператору с меткой AAA.

TRANSFER PICK,STK7,STK21

- равновероятный переход к операторам с номерами STK7, STK7+1, STK7+2, ..., STK21.

TRANSFER FN,AAA,5

- переход к оператору, метка которого равна сумме значения функции AAA и числа 5.

TRANSFER P,4,41

- переход к оператору, метка которого равна сумме значения параметра N 4 транзакта и числа 41.

TRANSFER SBR,PRC,7

- переход к оператору PRC с записью в параметр N 7 транзакта метки данного оператора.

LOOP 6,MET

- организация цикла - переход 5 раз к оператору с меткой MET и на шестой раз - к следующему по порядку оператору.

TEST E V7,K256,LAB

- переход по условию (условная передача управления): в позициях 13-18 записывается знак отношения, в первых двух подполях поля переменных записываются сравниваемые величины, если условие выполняется, то перехода нет, иначе переход есть к оператору с меткой LAB. Символы отношений: G - больше, L - меньше, E - равно, NE - неравно, LE - меньше или равно, GE - больше или равно. В данном примере перехода нет, если V7 = 256, иначе переход к оператору с номером LAB.

QUEUE SQV

- оператор организации очереди, длина очереди SQV увеличивается на единицу.

DEPART SQV

- то же, но длина очереди уменьшается на единицу.

PRIORITY 2

- транзакту присваивается приоритет 2.

SIMULATE

- начальная карта программы, если разработчик намерен выполнить прогон модели. Если эта карта отсутствует, то интерпретатор проверяет правильность записи модели на языке GPSS, но прогона модели не выполняет.

START 100,,25

- занесение значения 100 в итоговый счетчик, вывод накопленных статистических данных производится с интервалом изменения содержимого итогового счетчика в 25 единиц.

TABULATE MAT7

- в соответствующий интервал гистограммы с именем MAT добавляется единица.

MAT7 TABLE P3,8,1,5

- описание таблицы (гистграммы) MAT7, предназначенной для табулирования величины, значения которой находятся в третьем параметре транзакта, входящего в оператор TABULATE MAT7, верхний предел левого интервала гистограммы равен 8, ширина каждого последующего интервала равна 1, всего интервалов 5.

5 VARIABLE X2

- K25 - вычислительный оператор, в данном случае из хранимой величины N 2 вычитается число 25 и результат присваивается переменной N 5.

SAVEVALUE 5,*3

- хранимая величина N 5 (X5) получает значение третьего параметра транзакта.

Основные команды интерпретатора GPSS/PC

@<имя файла>

- загрузить исходный текст модели

SAVE <имя файла>

- сохранить текст модели

DISPLAY [<строка N 1>],[< строка N 2>]

- вывести на экран текст модели, начиная со строки N1 , до строки N 2.

DELETE [<строка N 1>],[<строка N 2>]

- удалить из программы текст, начиная со строки N1 , до строки N 2.

EDIT <номер строки>

- отредактировать строку текста

CLEAR

- сбросить всю статистику модели в ноль (включая таймеры абсолютного и относительного модельного времени), вернуть все транзакты в пассивный буфер .

END

- завершить работу интерпретатора

2.4.3 Создания моделей в GPSS

Модели систем на GPSS могут быть записаны в виде блок-схем или представлены в виде перфокарт, эквивалентных блок-схеме. Разработчик модели обычно начинает свою работу с построения блок-схемы системы, которую он собирается промоделировать. Именно в процессе построения блок-схемы разработчик продумывает основные методы моделирования своей системы.

Блок-схема представляет собой набор фигур с характерным очертанием блоков, соединенных между собой линиями. Разработчику моделей предоставляется выбор из набора более чем 40 блоков. Вид каждого из этих блоков стандартен. Различие между видами блоков необходимо для того, чтобы легко было изучать юлок-схемы моделей.

Модель строится таким образом, что из допустимого множества блоков выбирают необходимые и далее их выстраивают в диаграмму, для того чтобы в процессе использования моделей они как бы взаимодействовали друг с другом. Использование блоков при построении моделей зависит от логической схемы работы реальных систем. После того, как модель создана, взаимодействие между блоками моделей аналогично взаимодействию элементов моделируемых реальных систем.

Типичный для моделей на GPSS вид блок-схемы предоставлен на рисунке рис. 3.1. На этом рисунке показано 13 различных блоков. Некоторые из них присутствуют на рисунке более чем один раз. Мы не будем детально рассматривать способ записи тех или иных блоков. Наша цель – дать общий вид блок-схем, чтобы ими можно было пользоваться далее.

Это подмножество выбрано таким образом, чтобы можно было создавать законченные, относительно простые модели систем на GPSS. Чем больше видов блоков знает разработчик, тем более сложные системы он может строить. После того, как пользователь будет знать блоки всех видов, он может разрабатывать сложные модели с использованием GPSS относительно легко и просто.

Операции. Операциями блоков являются глаголы, описывающие основное функциональное назначение блоков. Каждый из блоков характеризуется своим собственным предписанным ему глаголом. В большинстве случаев разрешается задавать сокращенное название операции путем записи первых четырех символов слова, обозначающего операцию. Некоторые из блоков GPSS используют вспомогательные операторы, которые в общем случае обозначают буквой X.

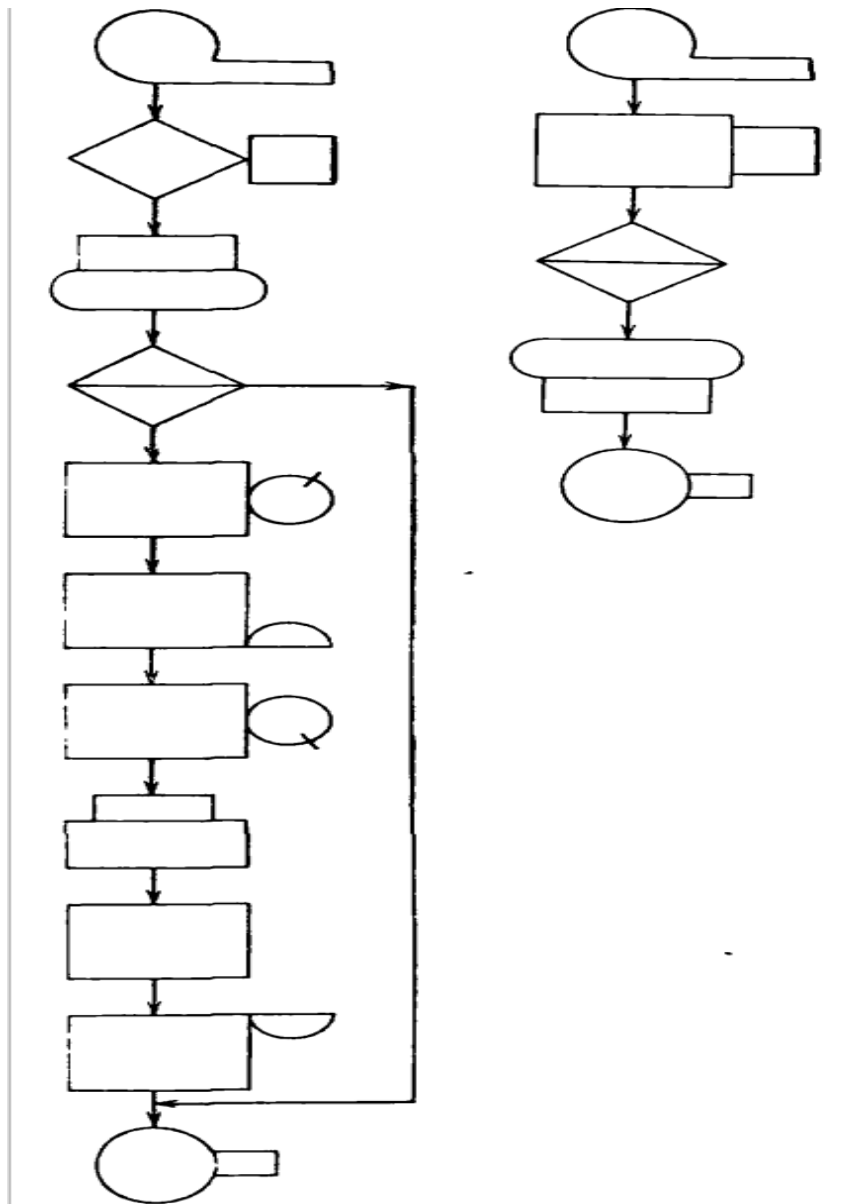


Рис. 8 Типичный вид блок-схем на GPSS

Основные сведения о блоках в GPSS

Блок-схемы в GPSS делятся на три категории:

1. Местоположение.
2. Операции.
3. Операнды.

Местоположение. Каждый блок занимает вполне определенное место в блок-схеме. Это место может быть определено нумерацией. Часто нужно

знать, какое место занимает тот или иной блок в модели. Это может оказаться необходимым для реализации ссылки из одного или более блоков на указанный блок. Для этого, казалось бы, необходимо перенумеровать заранее все блоки в модели. Однако вместо того, чтобы нумеровать блоки, разработчику предоставляется возможность присвоить любому блоку символические имена. При использовании символических имен интерпретатор записывает абсолютные значения номеров вместо этих имен. Символические имена состоят из алфавитно-цифровых символов, причем их число не должно превышать пяти, к тому же первые три символа должны быть алфавитными.

Операнды. Блоки могут содержать различные связанные с ними операнды. Операнды блоков задают информацию, специфичную для действия данного блока. Под значениями операндов могут подразумеваться значения аргументов, используемых при обращениях к подпрограммам. Число операндов каждого блока зависит от типа блоков. Ни один из блоков не использует более семи операндов. Операнды, в общем случае, обозначены символами А, В, С, D, E, F и G.

Транзакты

Транзакты являются динамическими элементами GPSS-модели. Работа GPSS-модели заключается в перемещении транзактов от блоков к блокам.

В самом начале моделирования в GPSS-модели нет ни одного транзакта. В процессе моделирования транзакты входят в модель в определенные моменты времени в соответствии с теми логическими потребностями, которые возникают в моделируемой системе. Подобным же образом транзакты покидают модель в определенные моменты времени в зависимости от специфики моделирования. В общем случае в модели существует большое число транзактов, однако в один момент времени движется только один транзакт.

Если транзакт начал свое движение, он перемещается от блока к блоку по пути, предписанному блок-схемой. В тот момент, когда транзакт входит в блок, на исполнение вызывается соответствующая подпрограмма, и далее транзакт пытается войти в следующий блок. Такое продвижение транзакта продолжается до тех пор, пока не произойдет одно из следующих возможных событий:

- 1) транзакт входит в блок, функцией которого является задержка транзакта на некоторое определенное моделью время;
- 2) транзакт входит в некоторый блок, функцией которого является удаление транзакта из модели;
- 3) транзакт «пытается» войти в следующий блок в соответствии с предписанной блок-схемой логикой, однако блок «отказывается» принять этот транзакт. Позднее, когда условия в модели изменятся, транзакт попытается еще раз войти в следующий блок.

Если возникло одно из описанных условий, транзакт остается на месте, и начинается перемещение в модели другого транзакта.

ГЛАВА 3. АЛГОРИТМ УСЛУГИ PUSH-TO-TALK В АРХИТЕКТУРЕ IMS

3.1 Технология Push to Talk

Те кто, хоть как-то знакомы с английским не составит труда дословно перевести эту фразу – нажимай чтоб говорить. В принципе, ответ верный, но далеко не полный.

Функция Push to Talk в мобильных телефонах автоматически превращает ваш аппарат в подобие рации. Есть у этого понятия и более длинная расшифровка, а именно PoC (Push to Talk over Cellular – «нажми, чтобы говорить в сотовой сети»). Эта функция предоставляет возможность двум или более абонентам мобильной сети общаться между собой, не применяя набор привычных номеров, а лишь удерживая нажатой одну кнопку телефона, именуемую тангентой.

Надо сказать, что подобным образом работают переносные радиостанции, или, как принято называть их, рации. Главная особенность функции Push to Talk – это то, что в большинстве случаев разговор осуществляется в режиме IP-телефонии, например, через GPRS сеть, а значит, обходится абоненту дешевле обычных вызовов. Еще одна важная деталь в этом вопросе – очень быстрое соединение с выбранным абонентом. По заявлению большинства разработчиков PoC систем, оно составляет в среднем 0,5–1,5 секунды.

Да, многим на ум, особенно тем у кого есть компьютер, пришло слово «скайп». По большому счёту вы правы, принцип действия схож, разве что есть несколько технических особенностей, которые простому абоненту знать и не обязательно.

Чтобы воспользоваться функцией PoC, абонент должен выбрать собеседника (или группу собеседников) из списка контактов и нажать на клавишу «Разговор». При этом пока нажата тангента, можно говорить, а

когда эта клавиша отпущена, появляется возможность услышать ответ собеседника. Такой режим связи давно известен как «полудуплекс», когда одновременно возможна только односторонняя связь.

РоС системы часто используются, прежде всего, для оперативной передачи краткой, иногда predetermined или даже кодовой информации. Зачастую это какая либо инструкция или команда, иногда – короткое «статусное» сообщение, адресуемое группе лиц.

В отличие от систем профессиональной связи, услуга РоС основана на использовании подсистемы пакетной передачи данных сотовых сетей связи: GPRS, EDGE для систем GSM, PDSN для сетей CDMA 2000 или сетей поколения 3G (например, WCDMA или CDMA 1x EV). Чтоб не загружать вас всеми этими латинскими аббревиатурами, сразу скажем, что речь при использовании РоС передается не по выделенному коммутируемому речевому каналу, как при обычных мобильных звонках, а в пакетном режиме, в виде последовательно посылаемых блоков данных. То есть происходит виртуальное накопление вашего голоса, который маленькими порциями отправляется абоненту на том конце «провода». В этом принципиальное отличие Интернет-передачи голоса от простого мобильного разговора.

По оценкам компании Northstream, одного из лидирующих мировых аналитических агентств в области мобильной связи, решения Push to Talk для сетей GSM с GPRS, по меньшей мере, в пять раз эффективнее, чем обычная мобильная сеть в её классическом понятии. А дальнейшая модификация технологии GPRS и появление нового поколения 3G показывает 14-кратное преимущество. С экономической точки зрения реализация РоС имеет неоспоримые преимущества и обходится более чем в 6 раз дешевле, чем даже самые выгодные мобильные тарифы. И, наконец, ещё один важный фактор – это неограниченность расстояния, чего не скажешь об обычных рациях.

Что касается практической реализации системы Push to Talk, то тут, безусловно, выделяются два вопроса: откуда говорить и где говорить? Откуда – имеем ввиду мобильные телефоны (на профессиональном языке –

терминалы). Где – имеем ввиду мобильные сети. Да, именно мобильные сети, так как наличие данной функции в вашем телефоне не залог того, что вы сможете ею воспользоваться.

3.2 Поддержка возможности Push-to-Talk по сотовой связи

Сотовая связь по принципу push-to-talk (Push-to-Talk over Cellular, PoC) является первой из многочисленных сфер применения IMS. В 2003 году ведущие поставщики услуг и операторы (в том числе Ericsson) объявили о завершении совместной разработки спецификации Push-to-Talk over Cellular, основанной на IP Multimedia Subsystem (IMS), как определено стандартами 3GPP и 3GPP2. Большую поддержку многих операторов и поставщиков услуг получила не только данная спецификация, но и PoC, предоставленная для ратификации в Open Mobile Alliance (OMA). Open Mobile Alliance в настоящее время проводит работу по стандартизации, и учитывая тот факт, что в альянс входит более 350 представителей производителей, данная организация обеспечит преемственность стандартов и их функциональную совместимость.

Это должно сделать Push-to-talk общедоступным сервисом, подобно SMS и MMS, за счет обеспечения "прозрачности" сети для конечных пользователей. Такая спецификация разработана для удовлетворения огромной потребности рынка в Push-to-talk и IMS. Стандартизация ведет к расширению ассортимента типов и моделей терминалов за счет увеличения объемов производства у всех операторов, вовлеченных в процесс стандартизации.

PoC предлагает разнообразные сервисы для связи "абонент - абонент" и групповой связи, включая чаты, индивидуальные сигналы оповещения и управление присутствием. PoC работает только в среде с коммутацией пакетов и базируется на средствах оказания услуг IMS и общих функциях,

как, например, управление группой, списком и присутствием, проведение конференц-связи, безопасность, биллинг и O&M.

Сервисы Push-to-talk можно использовать в потребительском сегменте: оставаясь на связи с друзьями, планировать досуг, или общаться с членами семьи посредством нажатия кнопки. Эта услуга нацелена также на корпоративный сегмент, где она используется, в частности, для обмена информацией в рабочих группах - например, для находящегося на выезде специалиста IS/IT, которому необходимо связаться с коллегами для получения нужной информации.

Сервис Push-to-talk с учетом соответствующего позиционирования и ценообразования может быть представлен как новая голосовая услуга в сегменте между голосовыми вызовами и сервисами текстовых сообщений типа SMS. Данный сервис является более быстродействующим по сравнению с обычными голосовыми вызовами и обеспечивает более простое групповое соединение. Также он, в отличие от SMS, кроме собственно сообщения, обеспечивает более быструю и эмоциональную связь между абонентами, а также мгновенную ответную реакцию.

Устройство типа портативной радиостанции для связи в мобильных сетях впервые с успехом было применено в США. Уже в середине девяностых годов мобильный оператор Nextel запустил региональный сервис, который к 2004 году превратился в общенациональную сеть с 12,3 миллионами абонентов. Nextel продемонстрировал рынку впечатляющие финансовые результаты наряду с очень высоким уровнем проникновения (>90%) сервисов типа портативной радиостанции, что побудило остальных игроков рынка к анализу возможностей создания конкурирующих сервисов.

Гарантией успеха данного сервиса является его совместимость, аналогичная нынешней совместимости SMS, которая дает возможность клиентам общаться друг с другом вне зависимости от конкретного провайдера услуг. Совместимость сетей и технологий является сегодня ключевым фактором для клиентов, поскольку невозможность установить

связь между двумя абонентами, пользующимися различными пакетами мобильных услуг, может привести к замедлению распространения этой услуги. Именно поэтому IMS стал базой для PoC.

Технология Push-to-talk от компании Ericsson - Ericsson Instant Talk (EIT) - является комплексным решением, состоящим из трех основных компонентов: системы Ericsson IMS (IPMM), сервера приложений EIT и PoC-клиента на терминале пользователя.

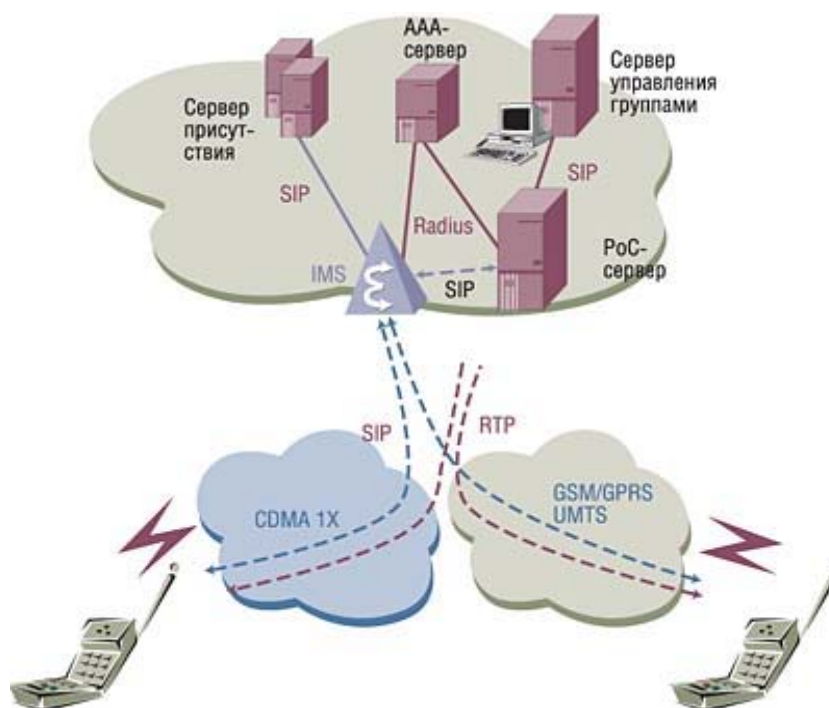


Рис. 9 Обобщенная схема PoC сети

3.3 Использование в IP сетях

В последнее время получило распространение использование РТТ в IP-сетях. Основной причиной этому послужила легкость организации передачи аудиосигнала точка-мультиточка (point-to-multipoint) для системы РТТ. За счет этого легко организуются отдельные группы, когда говорящего слышат все участвующие в данной группе пользователи.

РТТ базируется на двух основных технологиях:

- Серверный РТТ
- Бессерверный РТТ

Серверный РТТ

Все клиенты РТТ обращаются к серверу (например, сервер IMS) для получения разрешения на передачу сигнала и для получения информации об имеющихся группах и пользователях. Обмен служебной информацией между клиентом и сервером производится, как правило, по протоколу SIP.

Клиент выбирает группу (или одиночного пользователя) для разговора и, при нажатии на кнопку "разговор", обращается к серверу за разрешением начала сессии. Сервер не даст разрешение на начало сессии, если группа (пользователь) занята, например, другой пользователь уже передает голосовые данные этой же группе. Если разрешение получено, клиент начинает передачу голосовых данных другому пользователю (как unicast) или группе пользователей (как multicast).

После отпускания кнопки "разговор" серверу отсылается сигнал об окончании сессии. В зависимости от организации системы, сервер может самостоятельно прервать сессию, пошлав клиенту сигнал "конец сессии".

Преимущества: централизованное управление сессиями, в том числе между удаленными пользователями.

Недостатки: необходимо наличие центрального элемента (сервера), весьма дорогостоящего.

Бессерверный РТТ

Клиент в бессерверном РТТ, в отличие от серверного РТТ, самостоятельно принимает решение о начале сессии, анализируя, приходят ли сообщения для той группы, для которой он хочет начать сессию. Конец сессии клиент определяет по специфическому сигналу другого клиента или по тайм-ауту. Списки пользователей/групп необходимо организовывать отдельным протоколом (например, на основе отдельной мультикаст-группы).

Преимущества: дешевизна и простота построения системы.

Недостатки: отсутствие централизованного управления сессиями, возможны накладки передачи одновременно от двух и более пользователей, особенно в случае удаленных пользователей (с большой задержкой сигнала).

ГЛАВА 4. РАСЧЕТ ВЕРОЯТНОСТНО-ВРЕМЕННЫХ ХАРАКТЕРИСТИК

Теория массового обслуживания— целью исследований которого является рациональный выбор структуры системы обслуживания и процесса обслуживания на основе изучения потоков требований на обслуживание, поступающих в систему и выходящие из неё, длительности ожидания и длины очередей. В теории массового обслуживания используются методы теории вероятностей и математической статистики.

Рассмотрим поток заявок на обслуживание, поступающий на сервер. Ограничим время наблюдения за потоком одним часом, например, часом наибольшей нагрузки. В этот промежуток времени поток вызовов может считаться стационарным. Поток можно считать ординарным, так как в определенный момент времени поступление двух и более заявок является невозможным. Поток вызовов можно считать потоком без последствия, учитывая большое число источников заявок. Таким образом, поток заявок на обслуживание можно считать стационарным пуассоновским потоком, а функцию распределения промежутков времени между заявками - показательной.

Предположим, что сервер обслуживает поступающие заявки в режиме с потерями, то есть заявки, попадающие на систему в момент полной занятости ресурсов, получают отказ в обслуживании.

Вышеперечисленные предположения позволяют применить для расчета вероятностно-временных характеристик системы математическую модель, классифицируемую как $M/M/m:loss$ – система с m обслуживающими логиками и дисциплиной обслуживания с явными потерями. Основной характеристикой такой системы является доля времени, когда все логики сервера являются занятыми, то есть поступающие заявки получают отказ в обслуживании.

Этот параметр может быть рассчитан по В-формуле Эрланга:

$$p(m) := \frac{\left(\frac{\lambda}{\mu}\right)^m}{m!} \bigg/ \sum_{k=0}^m \frac{\left(\frac{\lambda}{\mu}\right)^k}{k!}$$

где λ – интенсивность поступления заявок,
 μ – интенсивность обслуживания заявок,
 m – количество обслуживающих логик.

Рассмотрим несколько случаев с разными количествами обслуживающих логиков m и разной интенсивностью поступающих заявок λ .

Первый случай.

Количество обслуживающих логиков $m=5$ и интенсивность обслуживания заявок $\mu=3$. Интенсивность поступления заявок λ будет меняться в диапазоне 1-2 заявок в секунду.

Таблица 1.1

λ	1	1.3	1.68	1.7	1.9	2
p	0.00003	0.00008	0.0002	0.0003	0.0004	0.0006

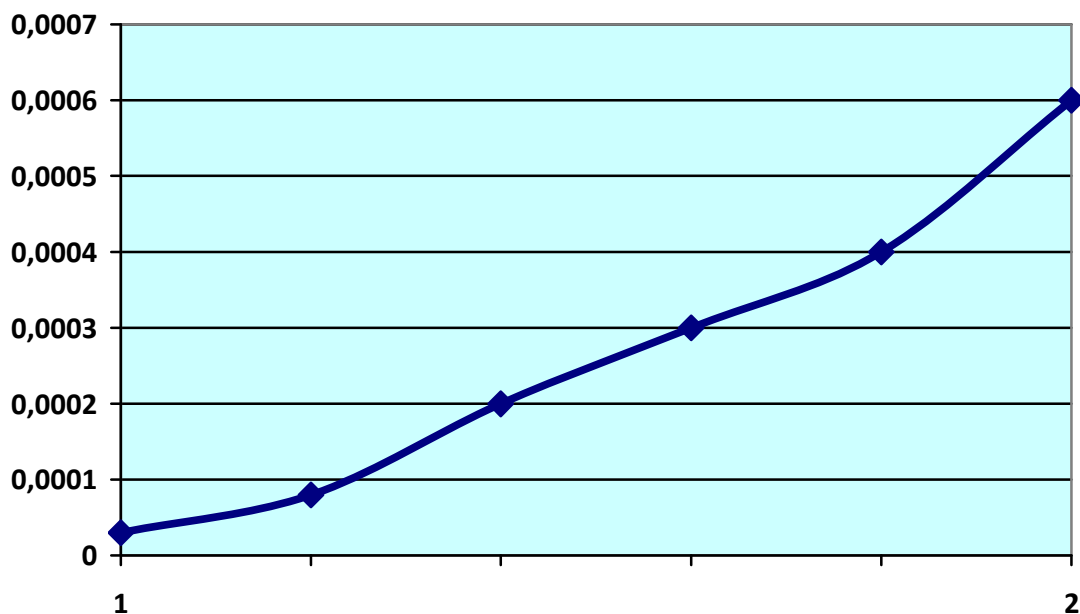


Рис. 10 График зависимости p от λ по Таблице 1.1

В графике видно что потери увеличиваются с увеличением потока заявок. (Рис.10) Но эти потери совсем не значительны. Так как они очень малы и не сильно влияют. В процентном виде максимум потери будет 0.06%.

Второй случай.

Теперь увеличим количество интенсивности поступления заявок λ до диапазона 2-3 заявок в секунду так говоря создадим ситуацию «Час пик», а интенсивности обслуживания заявок все такая же $\mu=3$. И количество обслуживающих логиков оставим таким же $m=5$.

Таблица 1.2

λ	2.1	2.3	2.5	2.7	2.8	2.95
p	0.0006	0.0001	0.0014	0.002	0.0023	0.0029

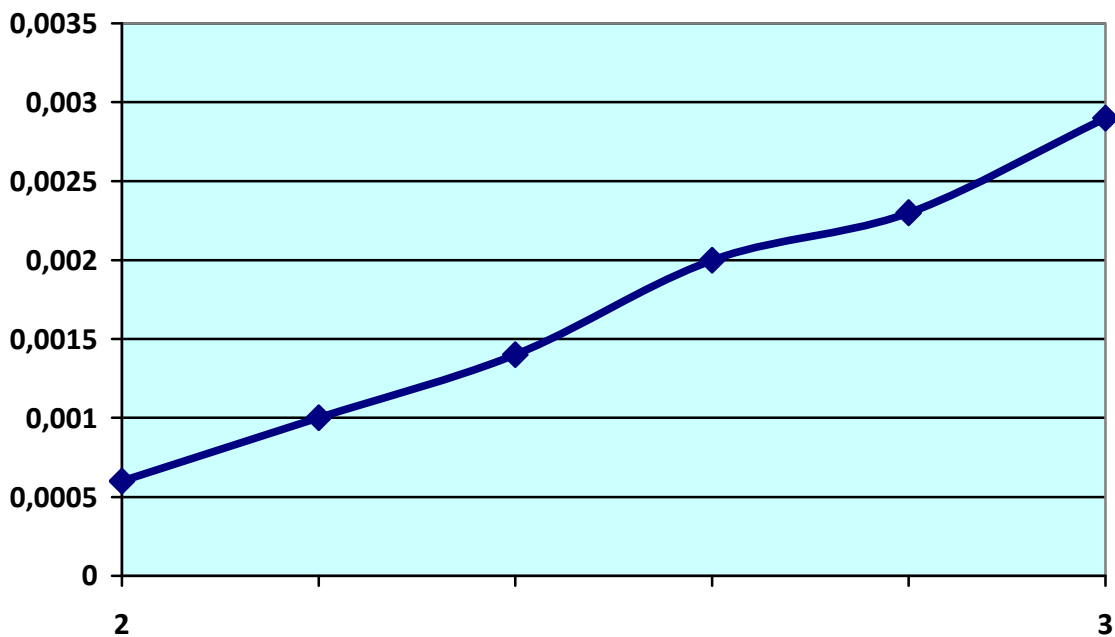


Рис. 11 График зависимости p от λ по Таблице 1.2

А на этом графике видно как потери увеличились, но они все еще не значительны. (Рис.11) В процентном виде максимальная потеря будет таким 0.3%. Так как у нас количество обслуживающих логиков $m=5$. И это дает нам обрабатывать заявки с меньшими потерями.

Третий случай.

А теперь уменьшим количество обслуживающих логиков $m=3$. Количество интенсивности поступления заявок λ и интенсивности обслуживания заявок μ оставим такими же, λ в диапазоне 2-3 заявок в секунду и $\mu=3$.

Таблица 1.3

λ	2.1	2.3	2.5	2.7	2.8	2.95
p	0.029	0.035	0.042	0.05	0.054	0.06

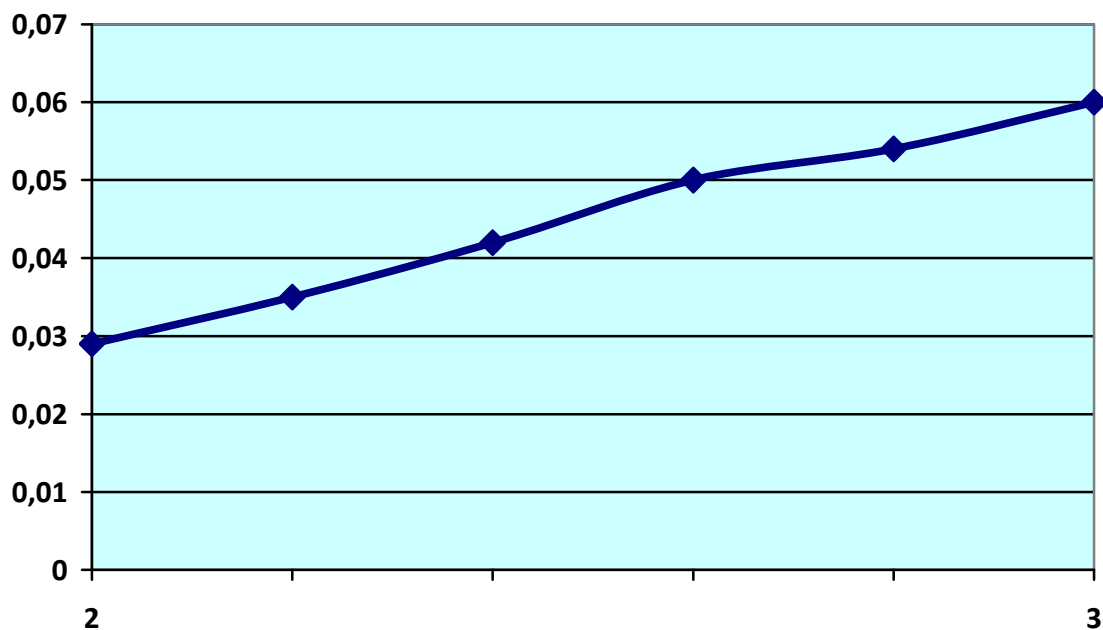


Рис. 12 График зависимости p от λ по Таблице 1.3

Но вот тут уже видно что потери стали значительными.(Рис.12)
 В процентном виде максимальная потеря будет 6%. Мы получили этот результат за чет того что у нас количество обслуживающих логиков уменьшалось. А это значить нагрузка на сет увеличиться, соответственно потери тоже увеличатся и качество обслуживания ухудшаться.

ГЛАВА 5. РАЗРАБОТКА ПРОГРАММНОЙ МОДЕЛИ УСЛУГИ PUSH-TO-TALK НА GPSS.

5.1 Модель услуги Push-to-Talk на GPSS

С учетом анализа, проделанного в ГЛАВЕ 2, для разработки программной модели услуги Push-to-Talk выбран язык имитационного моделирования GPSS. Из указанных операторов в разделе 2.4.2, здесь используются следующие:

GENERATE - генерация транзактов, интервалы времени между появлениями транзактов распределены равномерно в диапазоне [12-4, 12+4], первый транзакт появится с задержкой в 50 единиц модельного времени, всего будет создано 5 транзактов, приоритет транзактов равен единице.

FUNCTION

- описание функции,

ENTER - занятие транзактом емкости в накопителе.

LEAVE - освобождение единиц памяти в накопителе

STORAGE - описание накопителя емкостью.

TERMINATE - удаление транзакта из системы.

ADVANCE A,B - задержка транзакта на время, определенное содержимым полей A и B, смысл величин, записываемых в этих подполях, такой же, как и в операторе GENERATE.

ASSIGN - изменение параметров транзактов.

TRANSFER - безусловная передача управления оператору.

QUEUE SQV - оператор организации очереди, длина очереди SQV увеличивается на единицу.

DEPART SQV - то же, но длина очереди уменьшается на единицу.

START - занесение значения в итоговый счетчик.

TABULATE MAT7 - в соответствующий интервал гистограммы с именем MAT добавляется единица.

MAT7 TABLE P3,8,1,5 - описание таблицы (гистграммы) MAT7, предназначенной для табулирования величины, значения которой находятся в третьем параметре транзакта, входящего в оператор TABULATE MAT7, верхний предел левого интервала гистограммы равен 8, ширина каждого последующего интервала равна 1, всего интервалов 5.

VARIABLE - вычислительный оператор.

SAVEVALUE - хранимая величина транзакта.

Основные команды интерпретатора GPSSPC

@<имя файла> - загрузить исходный текст модели

SAVE <имя файла> - сохранить текст модели

DISPLAY[<строка N 1>],[< строка N 2>] - вывести на экран текст модели, начиная со строки N1 , до строки N 2.

DELETE [<строка N 1>],[<строка N 2>] - удалить из программы текст, начиная со строки N1 , до строки N 2.

EDIT <номер строки> - отредактировать строку текста

CLEAR - сбросить всю статистику модели в ноль (включая таймеры абсолютного и относительного модельного времени), вернуть все транзакты в пассивный буфер .

END - завершить работу интерпретатора.

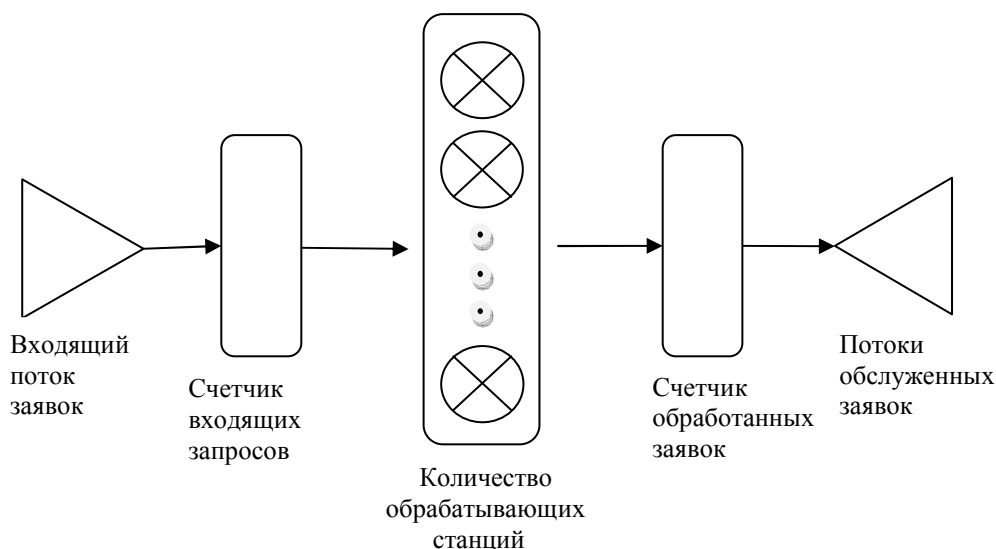


Рис. 13 Схема обслуживания заявок

5.2 Основной программный код

```
Operator STORAGE 5;  
GENERATE (Exponential(1,0,0.5));  
QUEUE Other;  
ENTER Operator;  
DEPART Other;  
ADVANCE 3,1;  
LEAVE Operator;  
TERMINATE 1;  
START 1;
```

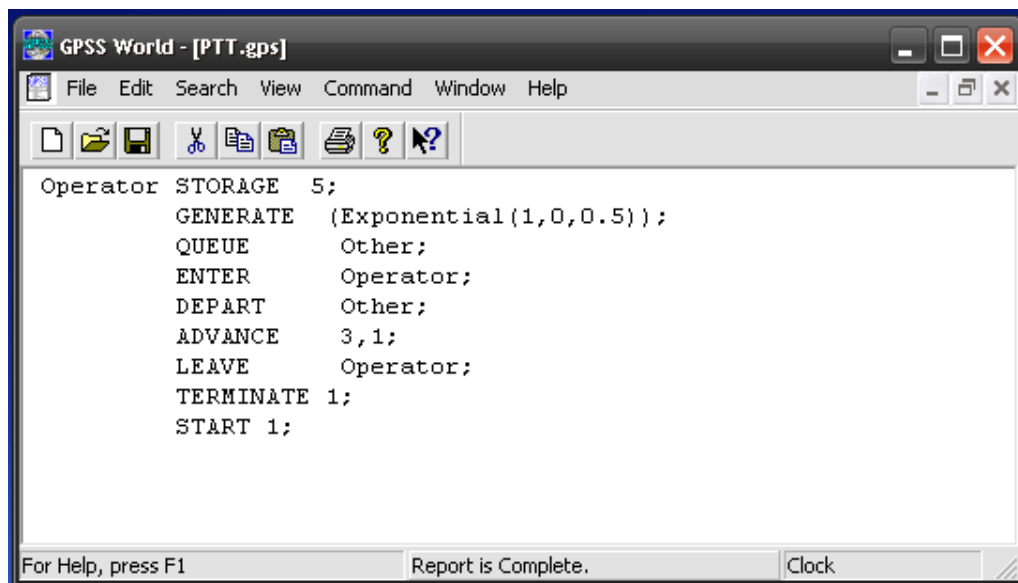


Рис. 14 Ввод программного кода

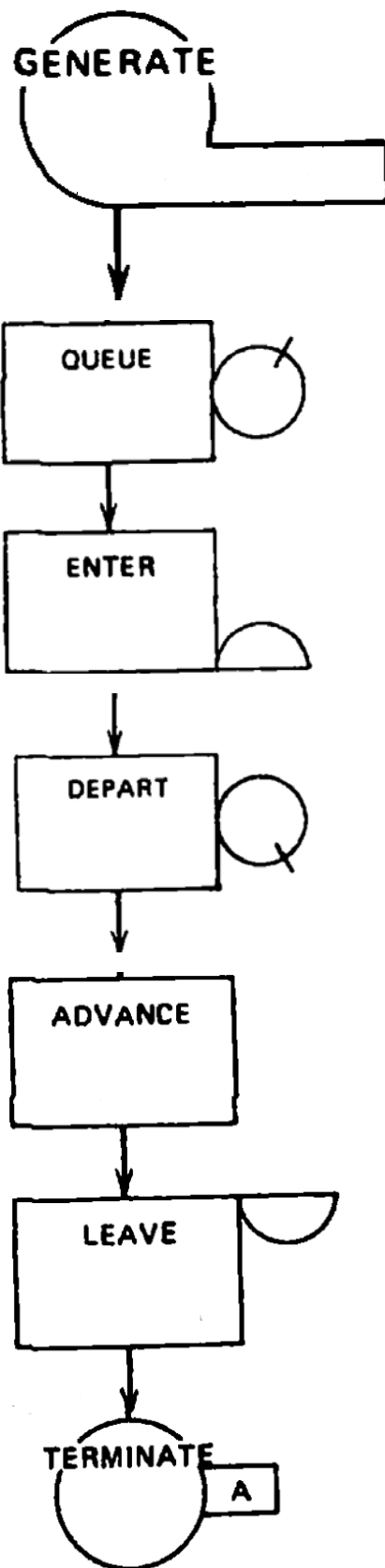


Рис. 15 Блок схема программы

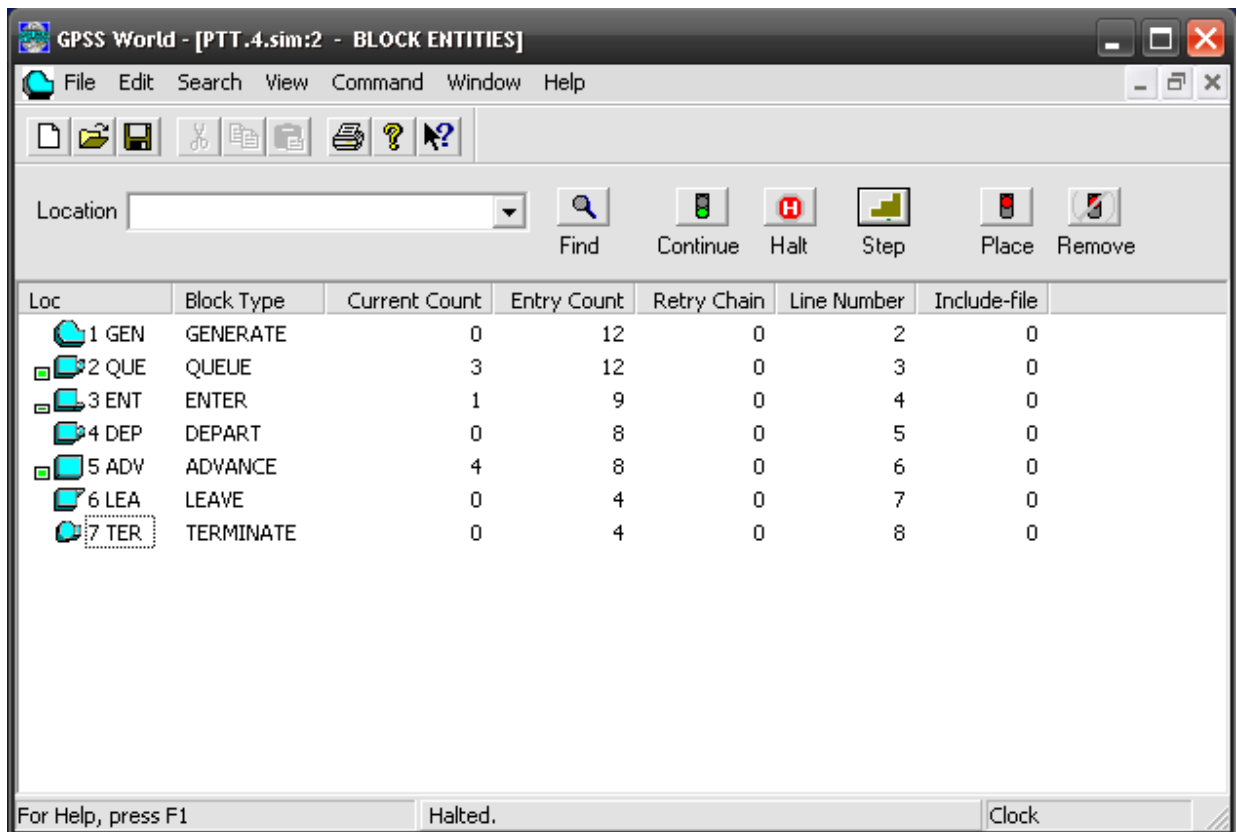
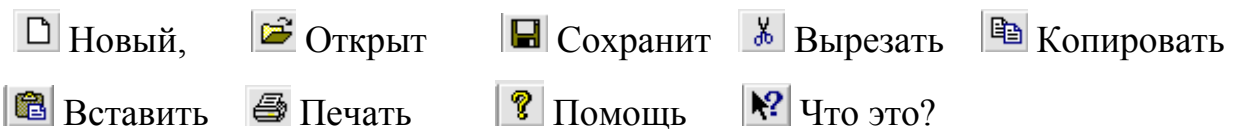





Рис. 16 Вид программы с блок схемами

В рисунке 12 приведен вид готовой программы, его интерфейс. Тут имеются стандартный набор кнопок:



Но кроме этих кнопок есть и функциональные кнопки нужные для управления процессом. Это кнопки:

	Поиск
	Остановить
	Продолжит

	По шаговое
	Поставить
	Удалить

«Поиск» - при нажатии кнопки, производится поиск по заданному параметру.

«Остановить» - при нажатии кнопки, производится остановка программы.

«Продолжит» - при нажатии кнопки, производится возобновление программы

Если после нажатия кнопки «Остановить» нажать кнопку «Пошаговое», то можно просматривать ежесекундное изменение результатов.

Для вывода полных результатов в отдельном окне, нужно нажать кнопку «Поставить» и потом нажать кнопку «Продолжить».

И мы получим такие результаты. (Рис.17)

5.3 Результаты моделирования

GPSS World Simulation Report - PTT.4.2

Thursday, May 27, 2010 13:49:37

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	3.139	7	0	1

NAME	VALUE
OPERATOR	10000.000
OTHER	10001.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
	1	GENERATE	8	0	0
	2	QUEUE	8	1	0
	3	ENTER	7	1	0
	4	DEPART	6	0	0
	5	ADVANCE	6	4	0
	6	LEAVE	2	0	0
	7	TERMINATE	2	0	0

QUEUE	MAX	CONT.	ENTRY	ENTRY (0)	AVE. CONT.	AVE. TIME	AVE. (-0)
RETRY							
OTHER	3	2	8	5	0.633	0.248	0.662 0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE. C.	UTIL.	RETRY
DELAY									
OPERATOR	5	0	0	5	7	1	3.262	0.652	0 1

CEC XN	PRI	M1	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
7	0	2.423	7	3	4		

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
9	0	3.175	9	0	1		
1	0	3.848	1	5	6		
4	0	4.812	4	5	6		
5	0	5.726	5	5	6		
6	0	5.798	6	5	6		

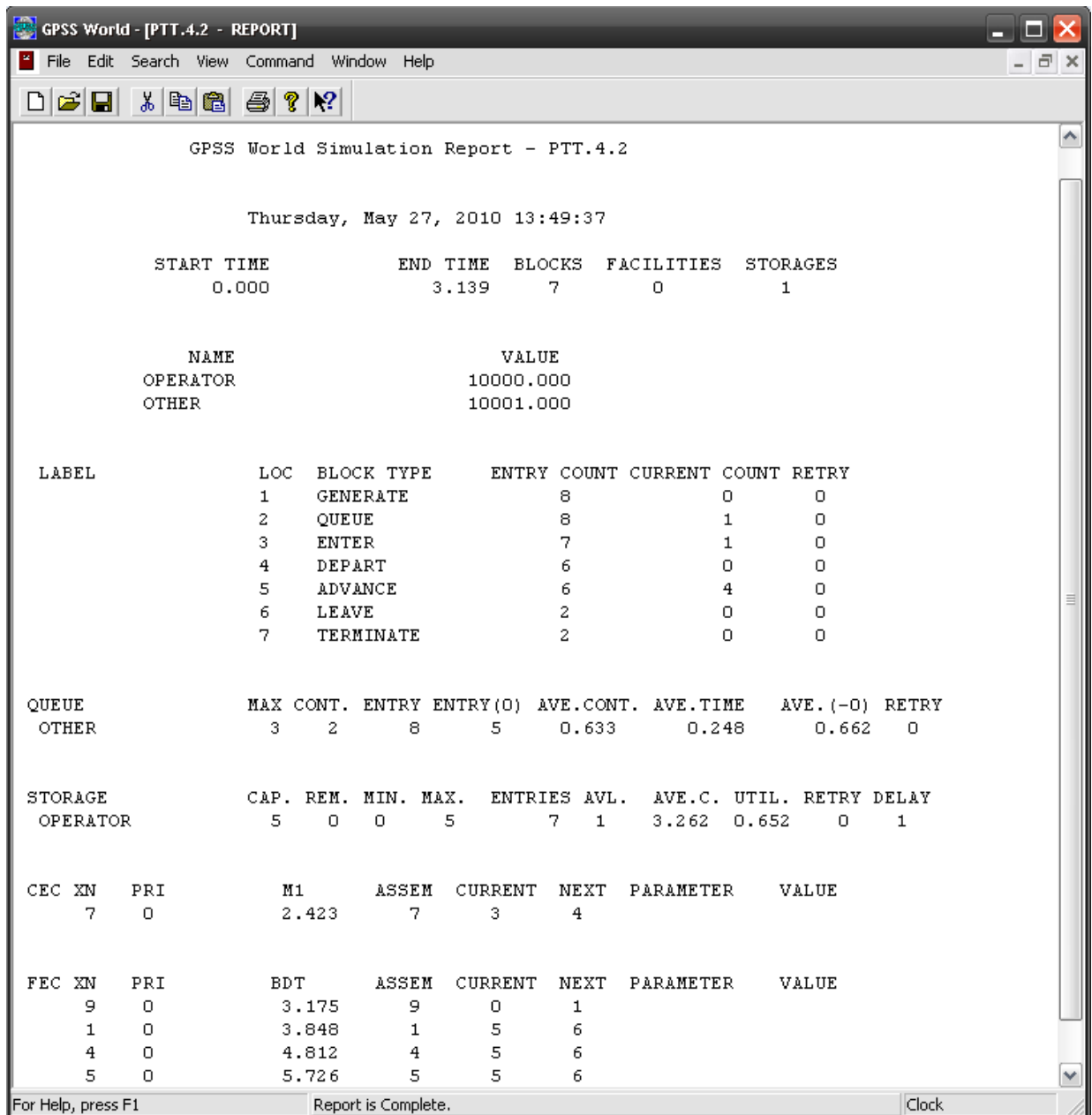


Рис. 17 Вывод результатов в программе

Таблица 2.1

λ	1	1.3	1.68	1.7	1.9	2
p	0,000028	0,00007	0,00019	0,00028	0,00038	0,00056

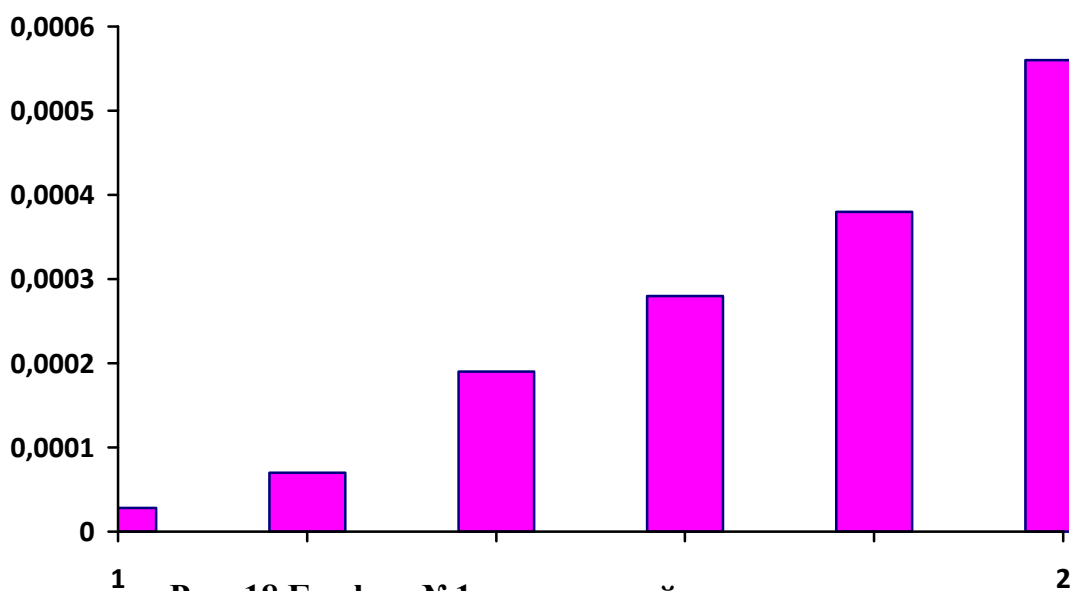


Рис. 18 График №1 полученный от моделирования

Таблица 2.2

λ	2.1	2.3	2.5	2.7	2.8	2.95
p	0,000595	0,00093	0,0013	0,00192	0,00223	0,00282

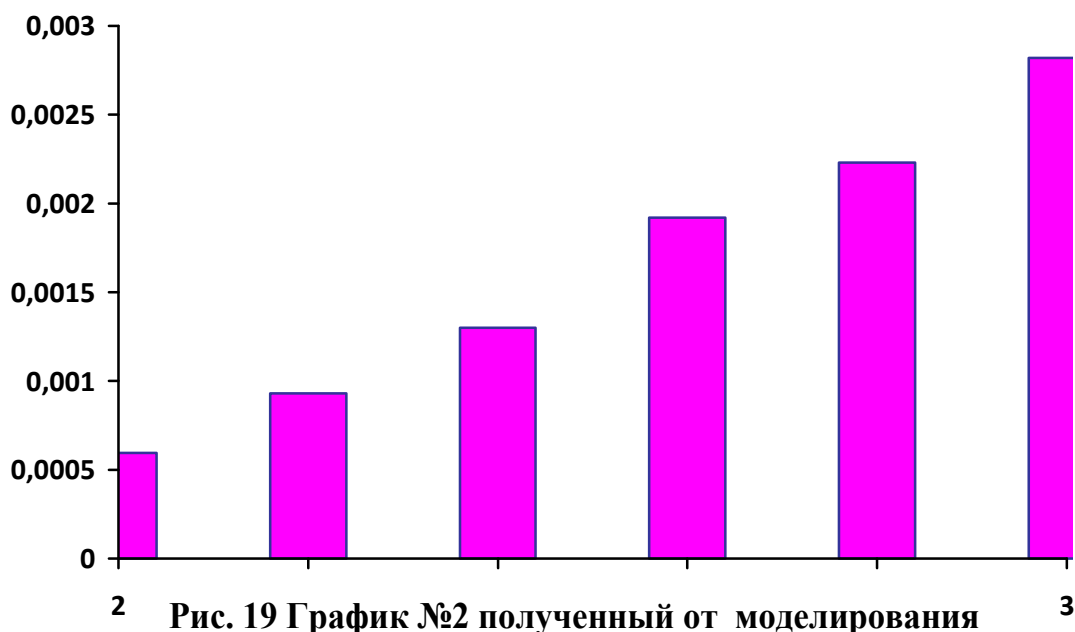


Рис. 19 График №2 полученный от моделирования

Таблица 2.3

λ	2.1	2.3	2.5	2.7	2.8	2.95
p	0,028	0,032	0,04	0,047	0,051	0,057

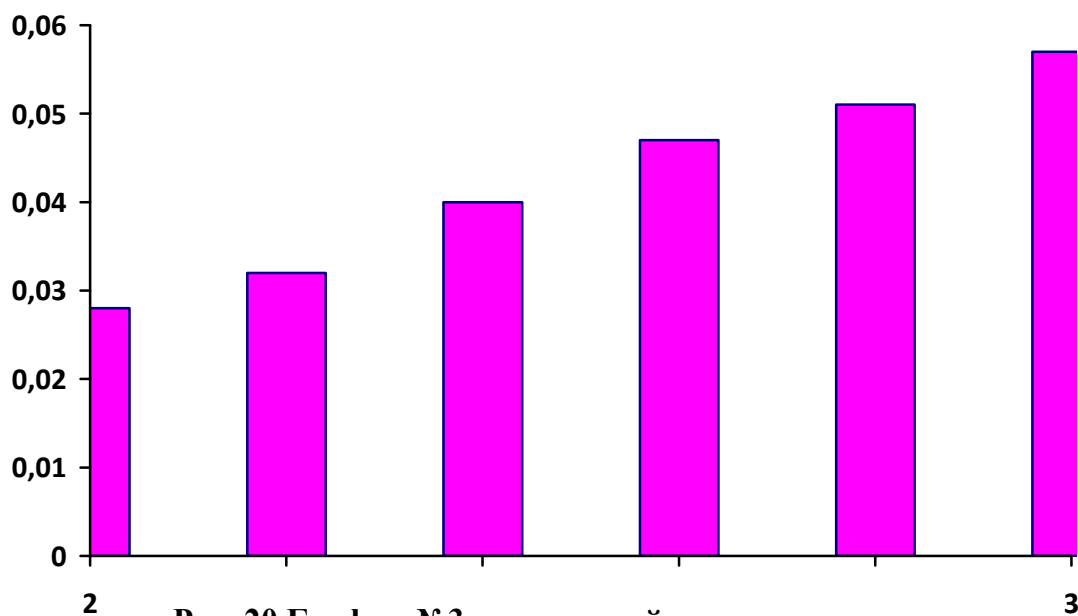


Рис. 20 График №3 полученный от моделирования

Все графики были получены методом моделирования на GPSS с теми же данными, которыми были получены графики ручных расчетов. Полученные значения и графики не значительно, но различаются от тех, которые мы получили методом ручных расчетов в предыдущей главе. И в следующей главе мы будем их сравнивать, что бы узнать величину погрешности.

ГЛАВА 6. СРАВНЕНИЕ РЕЗУЛЬТАТОВ РАСЧЕТА И МОДЕЛИРОВАНИЯ УСЛУГИ PUSH-TO-TALK.

Результаты, полученные методом математических расчетов и результаты полученные методом моделирования на GPSS почти не отличаются от друг- друга.

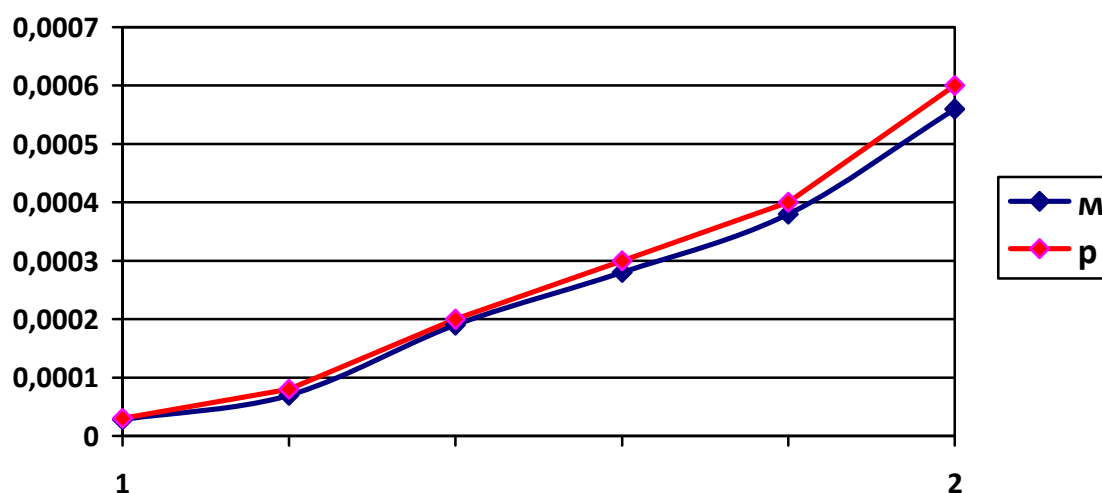


Рис. 21 Сравнение графиков №1 ручных(p) и с моделированных(m)

Можно заметить, что график почти не отличается от графика полученного методом ручных расчетов, погрешность составила 5-7%. Точно так же и остальные графики полученные методом моделирования в GPSS имеют не значительные отклонения от тех графиков, которые получены методом ручных расчетов.

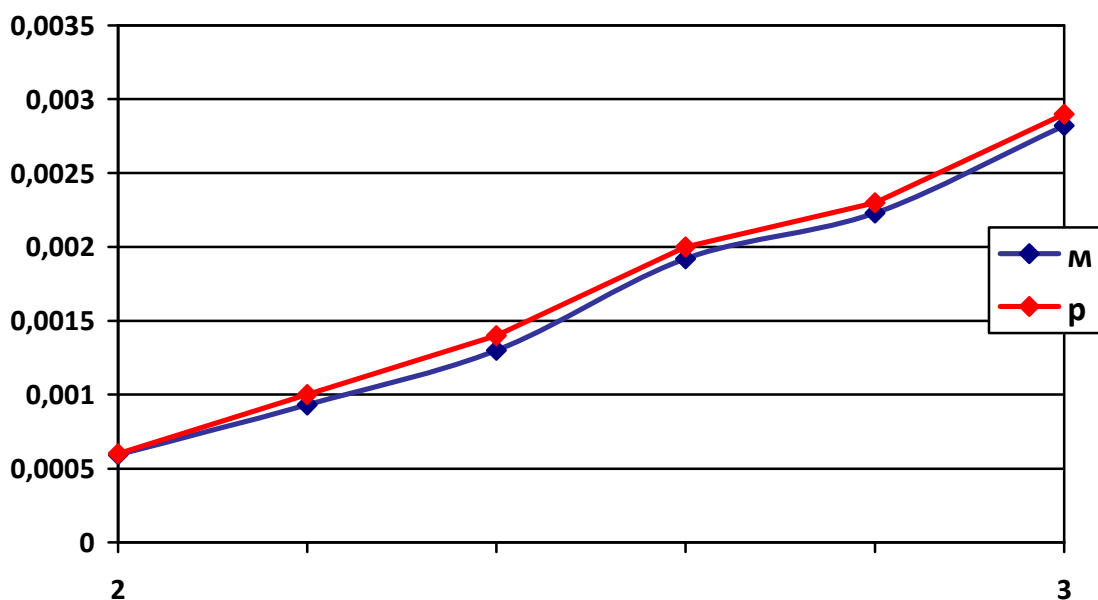


Рис. 22 Сравнение графиков №2 ручных(p) и с моделированных(m)

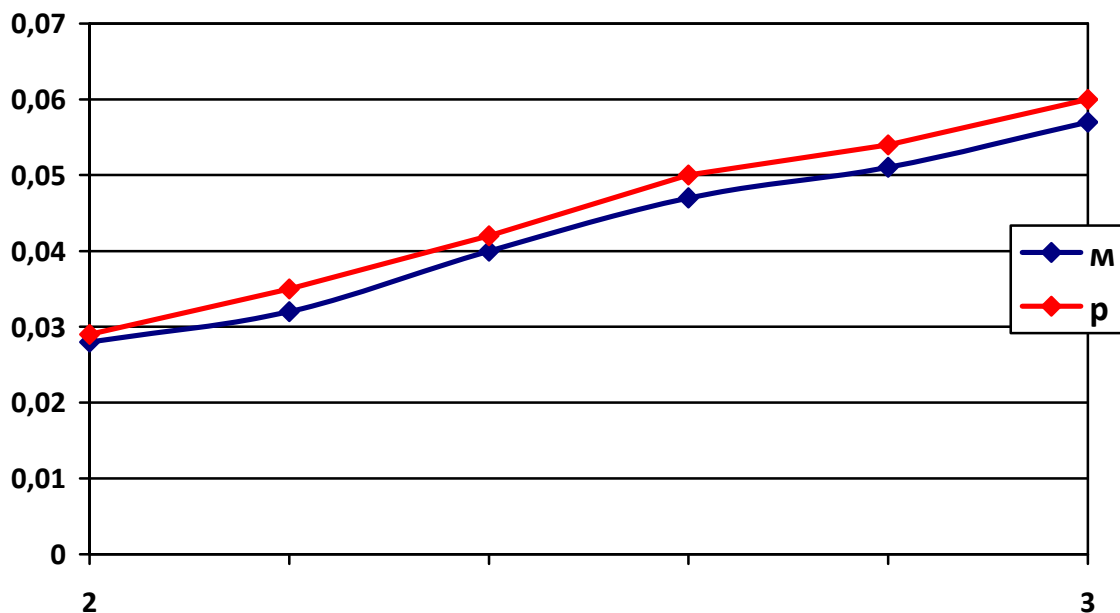


Рис. 23 Сравнение графиков №3 ручных(p) и с моделированных(m)

Беря во внимания то что результаты расчетов сделанных в моделирование(Таблица 2.1) имеют погрешность 5-7% от результатов ручных расчетов(Таблица 1.1), а это не превышает допустимой нормы в 10%. То можно сделать вывод что, программная модель, сделанная на GPSS отвечает всем требованиям.

ЗАКЛЮЧЕНИЕ

Актуальность рассматриваемой темы заключается в том, что услуги Push-to-Talk (РТТ) уже завоевали популярность в США и в Европе. Российские операторы сотовой связи пока с осторожностью смотрят в сторону данной новации. Ниша для этого продукта (точнее, продукта с подобными потребительскими свойствами) на рынке есть.

Исследования показывают, что услуги Push-to-Talk (РТТ) имеют достаточный спрос у определенного сегмента рынка. Например, такие сервисы актуальны для курьерских служб, компаний, занимающихся перевозками, или строительных компаний.

В данной дипломной работе была рассмотрена программная модель услуги реального времени Push-to-Talk. Были сделаны сравнения результатов полученных от программной модели с результатами математических расчетов. Проведенные работы показали, что программа отвечает поставленной к ней требованиям, и она может быть эффективно использована в работе.

СПИСОК СОКРАЩЕНИЙ

2G	Second generation
3GPP	Third Generation Partnership Project
3GPP2	Third Generation Partnership Project 2
ARPU	Average Revenue Per User
AS	Application Server
BGCF	Breakout Gateway Control Function
BICC	Bearer Independent Call Control
CDMA	Code Division Multiple Access
CSCF	Call Session Control Function
EDGE	Enhanced Data rates for GSM Evolution
EIT	Ericsson Instant Talk
ETSI	European Telecommunications Standards Institute
GGSN	General GPRS Support Node
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
HLR	Home Location Register
HSPDA	High-Speed Downlink Packet Access
HSS	Home Subscriber Server
I-CSCF	Interrogating CSCF
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ISC	IMS Service Control
ITU	International Telecommunication Union
MGCF	Media Gateway Control Function
MGW	Media Gateway Function
MMS	Multimedia Messaging Service
MRF	Media Resource Function

MRFC	Multimedia Resource Function Controller
MRFP	Multimedia Resource Function Processor
MTP	Message Transfer Part
NGN	Next Generation Network
OMA	Open Mobile Alliance
P-CSCF	Proxy CSCF
PDF	Policy Decision Function
PoC	Push to Talk over Cellular
PTT	Push to Talk
QoS	Quality Of Service
S-CSCF	Serving CSCF
SCTP	Stream Control Transmission Protocol
SDP	Service Description Protocol
SEG,	Signaling Gateway
SGW	Signaling Gateway
SLF	Subscriber Location Function
SMS	Short Message Service
SIP	Session Initiation Protocol
THIG	Topology Hiding Inter-Network Gateway
UMTS	Universal Mobile Telecommunication Sytem
VAS	Value Added Services
W-CDMA	Wideband Code Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
СМО	Событийный метод обработки
ММ	Математическая модель
ИМ	Имитационная модель
ТфОП	Телефонная сеть общего пользования
ЭВМ	Электронная вычислительная машина

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Syed A. Ahson, Mohammad Ilyas «IP Multimedia Subsystem (IMS)» Handbook. 2009 by Taylor & Francis Group, London New York
- [2] «Сети связи: Учебник для ВУЗов» Гольдштейн Б.С. Соколов Н.А. Яновский Г.Г. 2010
- [3] Т. Дж. Шрайбер «Моделирование на GPSS»
- [4] «GPSS World. Основы имитационного моделирования различных систем» Кудрявцев Е. М. ДМК Пресс2004
- [5] «Имитационное моделирование в среде GPSS» В. Томашевский, Е. Жданова Москва 2003
- [6] Travis Russell «The IP Multimedia Subsystem (IMS): Session Control and Other Network Operations» 2007 McGraw-Hill Osborne Media
- [7] «Моделирование систем. Инструментальные средства GPSS World» Боев В.Д 2004
- [8] Д. Н. Шевченко, И. Н. Кравченя «Имитационное моделирование на GPSS» Гомель 2007

ЛИСТИНГ ПРОГРАММ

* Модель Push-to-Talk.

*

* Сбор статистики о времени пребывания заявок в очереди.

* Интенсивности поступления заявок

Lambda1 EQU 100

* Длительность процесса моделирования

Timer EQU 1000000

* Просто так псевдонимы

Free EQU 0

Busy EQU 1

* Интенсивность обслуживания заявок

* Первое число -- номер потока

* Второе число -- интенсивность обслуживания заявок

Betas FUNCTION P\$STRM,D8

11,5

* Параметром функции является атрибут GRP каждой заявки

* Результат -- имя таблицы, в которую заносятся данные по этой заявке

TableNum FUNCTION P\$STRM,D9

11,Table1

AbsPrt VARIABLE 100-(P\$STRM/10)

RelPrt FVARIABLE 100-P\$STRM
SeTi FVARIABLE P\$OSTIME-P\$STIME

* Таблицы для статистики по заявкам

* Формат:

* ИМЯ TABLE

параметр,первый_промежуток,ширина_промежутков,число_промежутков

* Вот это набор таблиц, которые надо использовать, чтобы получить время

* заявки в системе вообще

*

Table11 TABLE M1,1000000,1,1

* А вот эти таблицы -- для получения статистики о времени задержки заявок

*

*Table11 TABLE P\$QTIME,100000,1,1

*

* Стандартный блок генерирования заявок

** Вот от сюда

* Генерируем заявки

L11 GENERATE (Exponential(1,0,Lambda11)),,,,99

* Устанавливаем значение номера потока

ASSIGN STRM,11

ASSIGN ABSA,99

ASSIGN RELA,99

ASSIGN STIME,(Exponential(1,0,FN\$Betas))

ASSIGN OSTIME,(Exponential(1,0,FN\$Betas))

* Передаём на групповую обработку (Group Processing)

TRANSFER ,GR_PROC

** и вот до сюда.

** Этот блок можно копировать и множить столько раз, сколько нужно.

** Метка в начале вовсе не обязательна, просто она помогает сориентироваться.

** При добавлении ещё одного подобного блока нужно выставить выбранные значения

** интенсивности поступления заявок (вместо Lambda11)

** и номер потока -- в блоке ASSIGN.

** Далее заявки с любого группового блока попадают на групповую обработку, которая правильно

** обрабатывает данные с любого количества групповых блоков.

* Блок групповой обработки

* Данный блок может обрабатывать любые заявки, имеющие параметр STRM(номер потока)

* Здесь надо организовать заявки

* Транслятор упорядочивает заявки автоматически.

GR_PROC SAVEVALUE Relative,PR

 SAVEVALUE Absolute,PR

MAIN_PROC PRIORITY P\$ABSA
 TRANSFER BOTH,INTRPT,WAIT

* Попытка прервать обработку заявки

* Прерываемое устройство -- PROCESSOR

* Условие прерывания -- наличие у заявки более высокого уровня

приоритета

INTRPT PREEMPT PROCESSOR,PR,WAIT,OSTIME,RE

* аналогично. Если произведён захват устройства,

* то записываем текущее время транзакта в параметр QTIME

ASSIGN QTIME,M1

* задержим заявку на время обработки

ADVANCE (P\$OSTIME)

* для разрешения временных узлов

* на логику моделирования никак не влияет

PRIORITY PR,BU

* освободим процессор

RETURN PROCESSOR

* выведем первую заявку из очереди на вход блока основной

обработки

UNLINK MAINQ,MAIN_PROC,1

* Занесём информацию о времени пребывания заявки в таблицу

QUIT TABULATE FN\$TableNum

* Вывести заявку из модели

TERMINATE

WAIT PRIORITY P\$RELA

LINK MAINQ,PR

* Блок таймера

GENERATE Timer

TERMINATE 1

РЕЗУЛЬТАТЫ

GPSS World Simulation Report - !forDisserText.174.1

Tuesday, June 01, 2010 16:18:18

START TIME	END TIME	BLOCKS	FACILITIES	STORAGES
0.000	1000000.000	72	1	0

NAME	VALUE
ABSA	10025.000
ABSOLUTE	10030.000
ABSPRT	10013.000
BETAS	10011.000
BUSY	1.000
FREE	0
GR_PROC	57.000
INTRPT	61.000
L11	1.000
L12	8.000
L13	15.000
L21	22.000
L22	29.000
L23	36.000
L31	43.000
L32	50.000
LAMBDA11	100.000
LAMBDA12	100.000
LAMBDA13	100.000

LAMBDA21	100.000
LAMBDA22	100.000
LAMBDA23	100.000
LAMBDA31	100.000
LAMBDA32	100.000
MAINQ	10033.000
MAIN_PROC	59.000
OSTIME	10028.000
PROCESSOR	10031.000
QTIME	10032.000
QUIT	67.000
RELA	10026.000
RELATIVE	10029.000
RELPR	10014.000
SETI	10015.000
STIME	10027.000
STRM	10024.000
TABLE11	10016.000
TABLE12	10017.000
TABLE13	10018.000
TABLE21	10019.000
TABLE22	10020.000
TABLE23	10021.000
TABLE31	10022.000
TABLE32	10023.000
TABLE33	UNSPECIFIED
TABLENUM	10012.000
TIMER	1000000.000
WAIT	69.000

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT	COUNT
RETRY					
L11	1	GENERATE	9866	0	0
	2	ASSIGN	9866	0	0
	3	ASSIGN	9866	0	0
	4	ASSIGN	9866	0	0
	5	ASSIGN	9866	0	0
	6	ASSIGN	9866	0	0
	7	TRANSFER	9866	0	0
L12	8	GENERATE	9912	0	0
	9	ASSIGN	9912	0	0
	10	ASSIGN	9912	0	0
	11	ASSIGN	9912	0	0
	12	ASSIGN	9912	0	0
	13	ASSIGN	9912	0	0
	14	TRANSFER	9912	0	0
L13	15	GENERATE	9831	0	0
	16	ASSIGN	9831	0	0
	17	ASSIGN	9831	0	0
	18	ASSIGN	9831	0	0
	19	ASSIGN	9831	0	0
	20	ASSIGN	9831	0	0
	21	TRANSFER	9831	0	0
L21	22	GENERATE	9980	0	0
	23	ASSIGN	9980	0	0

	24	ASSIGN	9980	0	0
	25	ASSIGN	9980	0	0
	26	ASSIGN	9980	0	0
	27	ASSIGN	9980	0	0
	28	TRANSFER	9980	0	0
L22	29	GENERATE	10016	0	0
	30	ASSIGN	10016	0	0
	31	ASSIGN	10016	0	0
	32	ASSIGN	10016	0	0
	33	ASSIGN	10016	0	0
	34	ASSIGN	10016	0	0
	35	TRANSFER	10016	0	0
L23	36	GENERATE	10067	0	0
	37	ASSIGN	10067	0	0
	38	ASSIGN	10067	0	0
	39	ASSIGN	10067	0	0
	40	ASSIGN	10067	0	0
	41	ASSIGN	10067	0	0
	42	TRANSFER	10067	0	0
L31	43	GENERATE	9863	0	0
	44	ASSIGN	9863	0	0
	45	ASSIGN	9863	0	0
	46	ASSIGN	9863	0	0
	47	ASSIGN	9863	0	0
	48	ASSIGN	9863	0	0
	49	TRANSFER	9863	0	0
L32	50	GENERATE	9949	0	0
	51	ASSIGN	9949	0	0
	52	ASSIGN	9949	0	0
	53	ASSIGN	9949	0	0
	54	ASSIGN	9949	0	0
	55	ASSIGN	9949	0	0
	56	TRANSFER	9949	0	0
GR_PROC	57	SAVEVALUE	79484	0	0
	58	SAVEVALUE	79484	0	0
MAIN_PROC	59	PRIORITY	110888	0	0
	60	TRANSFER	110888	0	0
INTRPT	61	PREEMPT	89744	0	0
	62	ASSIGN	89744	0	0
	63	ADVANCE	89744	0	0
	64	PRIORITY	79484	0	0
	65	RETURN	79484	0	0
	66	UNLINK	79484	0	0
QUIT	67	TABULATE	79484	0	0
	68	TERMINATE	79484	0	0
WAIT	69	PRIORITY	31404	0	0
	70	LINK	31404	0	0
	71	GENERATE	1	0	0
	72	TERMINATE	1	0	0

FACILITY	ENTRIES	UTIL.	AVE. TIME	AVAIL.	OWNER	PEND	INTER
RETRY DELAY							
PROCESSOR	89744	0.398	4.440	1	0	0	0
0							
0							

TABLE	MEAN	STD.DEV.	RANGE	RETRY
FREQUENCY CUM.% TABLE11	5.743	5.692	— - 1000000.000	0
100.00				9866
TABLE12	5.843	5.766	— - 1000000.000	0
100.00				9912
TABLE13	5.899	6.031	— - 1000000.000	0
100.00				9831
TABLE21	7.942	8.639	— - 1000000.000	0
100.00				9980
TABLE22	8.284	9.458	— - 1000000.000	0
100.00				10016
TABLE23	8.895	10.610	— - 1000000.000	0
100.00				10067
TABLE31	11.089	13.946	— - 1000000.000	0
100.00				9863
TABLE32	12.638	16.646	— - 1000000.000	0
100.00				9949

USER CHAIN	SIZE	RETRY	AVE.CONT	ENTRIES	MAX	AVE.TIME
MAINQ	0	0	0.261	31404	11	8.311

SAVEVALUE	RETRY	VALUE
RELATIVE	0	88.000
ABSOLUTE	0	88.000

FEC XN	PRI	BDT	ASSEM	CURRENT	NEXT	PARAMETER	VALUE
79493	88	1000006.757	79493	0	29		
79488	99	1000023.377	79488	0	1		
79484	89	1000077.399	79484	0	22		
79481	87	1000109.662	79481	0	36		
79491	97	1000110.394	79491	0	15		
79482	98	1000115.835	79482	0	8		
79487	78	1000119.125	79487	0	50		
79485	79	1000141.706	79485	0	43		
79494	0	2000000.000	79494	0	71		