

**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ
ИМЕНИ ПАТРИСА ЛУМУМБЫ**

На правах рукописи

Виана Карвалью Кравид Илкиаш

**Системы с пороговым управлением входящим
ПОТОКОМ**

Специальность 1.2.3. Теоретическая информатика, кибернетика

Диссертация на соискание учёной степени
кандидата физико-математических наук

Научный руководитель
к.ф.-м.н., доцент
И. С. Зарядов

Москва — 2024

Оглавление

Введение	4
1. Алгоритмы управления очередями и модели с обновлением	16
1.1. Алгоритмы управления очередями (на примере алгоритмов семейства RED)	16
1.2. Модели с полным и обобщенным обновлением	34
2. Анализ систем с пороговым механизмом обновления . .	50
2.1. Система массового обслуживания $G/M/1/\infty$ с одним порогом, безопасной зоной и обновлением.	50
2.2. Система массового обслуживания $G/M/1/\infty$ с одним порогом, обновлением и без безопасной зоны	68
3. Однопороговая система массового обслуживания с обобщенным обновлением и безопасной зоной в накопителе .	81
3.1. Описание системы, матрица переходных вероятностей вложенной цепи Маркова	81
3.2. Стационарное распределение по вложенной цепи Маркова числа заявок в системе	85
3.3. Вероятностные характеристики системы	87
3.4. Временные характеристики системы	91
Заключение	105

Список литературы	108
А. Результаты имитационного моделирования в среде GPSS	
для однопороговых моделей управления входящим тра-	
фиком на основе полного обновления	140
А.1. Имитационная модель системы массового обслуживания $G/M/1/\infty$	
с одним порогом, безопасной зоной и полным обновлением .	140
А.2. Имитационная модель системы массового обслуживания $G/M/1/\infty$	
с одним порогом, без безопасной зоны и полным обновлением	141
А.3. Сравнение результатов имитационного моделирования для	
однопороговых моделей с полным обновлением	142
В. Результаты имитационного моделирования в среде GPSS	146
В.1. Имитационная модель в среде GPSS однопороговой систе-	
мы массового обслуживания с обобщенным обновлением и	
безопасной зоной в накопителе	146
В.2. Результаты имитационного моделирования в среде GPSS для	
модели с обобщенным обновлением и безопасной зоной . . .	148
Список иллюстраций	150
Список таблиц	151

Введение

Актуальность темы исследования. Активное управление очередями (Active Queue Management — AQM), согласно рекомендации Инженерного совета Интернета (Internet Engineering Task Force RFC 7567) [17] считается передовой практикой предотвращения (уменьшения) и контроля сетевых перегрузок в маршрутизаторах посредством некоторых правил (алгоритмов), которые, будучи реализованными в маршрутизаторах, контролируют длину буферной очереди в целях эффективного управления перегрузками системы. Основная цель AQM алгоритмов — это предотвращение возможной перегрузки или, если она уже возникла, уменьшение ее воздействия на систему (маршрутизатор).

За последние тридцать лет было предложено множество различных AQM алгоритмов [2; 3; 6—9; 45; 47; 49; 58; 72—74; 95; 103; 106; 126] для надлежащего управления очередями для избежания таких нежелательных эффектов, как:

- переполнение буферной памяти маршрутизатора;
- избыточная или недостаточная загрузка канала передачи данных;
- сильные колебания задержки передачи данных и другие показатели.

Стоит заметить, что среди множества предложенных AQM алгоритмов нет ни одного «идеального» алгоритма, превосходящего всех остальных по ключевым параметрам качества функционирования (Quality of Service —

QoS), что обусловлено изменчивостью сетевого трафика и сетей, поэтому у каждого AQM алгоритма есть свои достоинства и недостатки.

Степень разработанности темы. Математические методы теории массового обслуживания (ТМО) и теории телетрафика позволяют как создавать вероятностные модели для новых AQM алгоритмов, так и оценивать эффективность функционирования уже существующих, обеспечивают возможность решения многочисленных задач по расчёту характеристик качества обслуживания (Quality of Service, QoS) и функционирования различных компонент сетей, включая оценку вероятностно-временных характеристик узлов коммутации и маршрутизации, анализ производительности сетей, управления потоками данных, расчёт потерь и загрузки цифровых линий связи при передаче данных, голоса и видеоинформации [41; 104; 189].

Значительный вклад в разработку и анализ классических моделей ТМО внесли А.К. Эрланг [42], А.Я. Хинчин [190], Б.В. Гнеденко [166], А.А. Боровков [157; 158], Д. Кендалл [76], Д. Литтл, Д. Кокс [178], В. Смит [178], Л. Клейнрок [80], Б.А. Севастьянов [188], Л. Такач [133], Ф. Поллачек, М. Ньюте [101; 102].

Среди работ современных авторов следует выделить работы Г.П. Башарина [153; 154; 156; 182], П.П. Бочарова [19; 20; 119; 159; 161], В.М. Вишневого [41; 165], А.Н. Дудина [11; 12; 40; 77; 81; 129], В.А. Наумова [20; 97–99; 183; 184], А.В. Печинкина [10; 111; 119; 185–187], К.Е. Самуйлова [5; 100; 112; 155; 184; 189], О.В. Семеновой [129], С.Н. Степанова, Э. Геленбе [54; 55; 57] и другие [84].

В представленной диссертации продолжено исследование моделей с обновлением и обобщенным обновлением, начатое в работах А.Я. Крейнина [87; 88], П.П. Бочарова [160], А.В. Печинкина [169] и других авторов [82; 144; 170; 172; 173]. Ключевое отличие от рассмотренных ранее моделей состоит в том, что введен однопороговый механизм управления вероятностным сбросом, что позволяет создавать аналитические модели, применимые для построения новых и анализа уже существующих механизмов активного управления очередью (Active Queue Management, AQM), в которых реализуется принцип числом заявок в системе путём их «случайного» удаления. Наиболее распространены алгоритмы типа RED (Random Early Detection), о которых на русском языке можно посмотреть в работах [179; 180] и которые более подробно рассматриваются в первой главе диссертации, а также алгоритм Drop Tail, алгоритм Blue, CHOKE, GREEN, YELLOW, BLACK и другие [6; 7; 44; 46].

Следует заметить, что изучение математических моделей, рассматривающих потерю (по той или иной причине) принятых в систему заявок, является актуальной задачей. Изначально потери заявок были связаны либо с либо с поступлением заявок особого типа: это или «катастрофы» (когда поступление особой заявки может привести к потере всех заявок либо в накопителе, либо в системе) [12; 40; 89; 96; 129], или отрицательные заявки (которые которые могут выбивать из накопителя одну (или несколько) обычных заявок, как правило, из числа последних пришедших) [15; 18; 23; 39; 55; 61; 78; 92–94; 110; 116; 117; 123; 141; 151; 159; 185; 186], либо и «катастрофы» и отрицательные заявки одновременно [141; 142]. Первая работа, в которой

была исследована система массового обслуживания с отрицательными заявками, принадлежит Е. Геленбе с соавторами [55], который в дальнейшем продолжил работать над этой темой [52; 54; 56]. Системы массового обслуживания с отрицательными заявками тесно связаны с G-сетями [51; 53; 57; 60; 75; 159] и до сих пор являются предметом интенсивных исследований [57; 123]. Также потеря данных в системе может быть связана с ненадежными приборами (снова используется понятие «катастрофы», приводящей к потере всех данных в системе) [77; 81; 137; 138; 187].

Основные отличия моделей с обобщённым обновлением от моделей с отрицательными заявками и моделей с катастрофами можно сформулировать следующим образом:

первое — в моделях с обобщённым обновлением сброс (потеря) поступивших в систему заявок из накопителя осуществляется в момент окончания обслуживания на приборе, а не в момент поступления в рассматриваемую систему (СМО с отрицательными заявками и (или) поступающим потоком катастроф);

второе — для моделей с обобщённым обновлением происходит вероятностный сброс произвольного числа принятых в систему заявок, что отличает от систем с отрицательными заявками (в которых, как правило, выбивается одна заявка) и от систем с катастрофами (как правило, теряются все заявки).

Стоит также отметить, что системы массового обслуживания с вероятностным сбросом заявок используются для анализа качества функционирования различных AQM алгоритмов [13; 25; 27–30; 113].

Цель и задачи работы. Целью диссертационной работы является разработка и анализ моделей для расчета показателей эффективности функционирования систем с пороговым управлением входящим потоком с помощью механизма обновления.

В рамках поставленной цели решаются следующие задачи:

1. проводится исследование эволюции алгоритмов активного управления на примере алгоритмов семейства RED, а также анализ уже полученных результатов для систем с обновлением.
2. Построение в виде систем массового обслуживания $GI/M/1/\infty$ с полным или обобщенным обновлением моделей порогового управления входящим потоком.
3. Вывод аналитических выражений для расчета таких показателей функционирования систем как среднее время пребывания сообщения в системе (задержка передачи сообщения), вероятность потери поступившего и вероятность обслуживания принятого в систему пакета, число потерянных пакетов с помощью систем массового обслуживания $GI/M/1/\infty$ с пороговым механизмом управления вероятностным сбросом принятых в систему заявок.

4. Построение в среде GPSS имитационных моделей, позволяющих численно оценить и сравнить показатели функционирования систем с различными вариантами дисциплин обслуживания и порогового механизма обновления.

Краткое содержание работы. Диссертация состоит из трех глав. Первая часть **Главы 1** диссертации посвящена развитию алгоритмов активного управления очередями (active queue management (AQM) algorithms) с описанием их основных характеристик. Вторая часть — это обзор основных результатов, полученным по моделям с полным и обобщенным обновлением.

Глава 2 посвящена изучению моделей порогового управления трафиком на базе системы массового обслуживания (СМО) $GI/M/1/\infty$ с обновлением (полным обновлением), когда заявка, находящаяся на приборе, в момент непосредственно перед уходом из системы, может либо полностью опустошить накопитель (с заданной вероятностью q), либо просто покидает систему (с вероятностью $p = 1 - q$). В отличие от предыдущих работ по данной тематике [87; 145; 160; 172] введено пороговое управление механизмом обновления.

В первом разделе главы рассмотрена однопороговая модель, в которой пороговое значение Q_1 определяет момент включения механизма вероятностного сброса поступивших в систему запросов (механизма обновления) — превышение текущей длины очереди значение порога Q_1 . Во втором разделе изучена однопороговая модель, в которой пороговое значение Q_1 не только определяет момент включения механизма вероятностного сброса

поступивших в систему запросов (механизма обновления), но и определяет границы области в накопителе, из которой находящиеся в ней заявки не могут быть сброшены (безопасная зона).

Для каждой модели выведены аналитические выражения для расчета основных вероятностно-временных характеристик.

В главе 3 продолжено изучение пороговых моделей управления входящим трафиком на базе СМО $GI/M/1/\infty$, но уже для случая обобщенного обновления, когда заявка непосредственно перед уходом из системы либо может с заданной вероятностью $q(i)$ или сбросить ровно i заявок из накопителя (если в нем было i и более заявок), или опустошить накопитель (если в нем было менее i заявок), либо просто покидает систему (с вероятностью $p = 1 - \sum_{i=1}^{\infty} q(i)$) [171; 177].

Рассмотрена однопороговая модель, в которой пороговое значение Q_1 не только определяет момент включения механизма вероятностного сброса поступивших в систему запросов (механизма обновления), но и определяет границы области в накопителе, из которой находящиеся в ней заявки не могут быть сброшены (безопасная зона), для следующих вариантов дисциплин обслуживания и обобщенного обновления: прямое обобщённое обновление (заявки сбрасываются из накопителя начиная с первой пришедшей) при обслуживании заявок в порядке поступления — FIFO/First и обратное обобщённое обновление при инверсионном обслуживании заявок — LIFO/Last.

Для каждого варианта выведены уравнения в терминах преобразований Лапласа-Стилтьеса (ПЛС) для функций распределения и средних значений времён пребывания в накопителе и системе обслуженной, сброшенной и

произвольной заявок, причём для прямого порядка обобщённого обновления при дисциплине обслуживания в порядке поступления (*FIFO*) в явном виде.

В **заключительном разделе** диссертации сформулированы основные результаты работы.

В **Приложении А** приведены результаты имитационного моделирования для рассмотренных во второй главе моделей. Для различных начальных значений таких параметров системы, как интенсивность обслуживания, интенсивность входящего потока, значение порога Q_1 получены численные значения следующих характеристик: вероятность сброса $p^{(\text{loss})}$ и вероятность обслуживания $p^{(\text{serv})}$ принятой в систему заявок, среднее время $w^{(\text{loss})}$ пребывания в накопителе сброшенной заявки и среднее время $w^{(\text{serv})}$ ожидания начала обслуживания, среднее число сброшенных и среднее число обслуженных заявок. Проведено сравнение вышеупомянутых характеристик для трех рассмотренных моделей.

В **Приложении В** представлены результаты имитационного моделирования для модели порогового управления входящим трафиком с безопасной зоной, рассмотренной в третьей главе. Для различных начальных значений таких параметров системы, как интенсивность обслуживания, интенсивность входящего потока, значение порога Q_1 получены численные значения следующих характеристик: вероятность сброса $p^{(\text{loss})}$ и вероятность обслуживания $p^{(\text{serv})}$ принятой в систему заявок, среднее время $w^{(\text{loss})}$ пребывания в накопителе сброшенной заявки и среднее время $w^{(\text{serv})}$ ожидания начала обслуживания, среднее число сброшенных и среднее число обслуженных заявок.

Научная новизна и основные результаты диссертации. Все основные результаты диссертации являются новыми. Отличие от предыдущих работ по системам с (полным) обновлением и обобщенным обновлением — введен пороговый механизм управления вероятностным сбросом, рассмотрены однопороговые модели с безопасной зоной в накопителе как для систем с полным обновлением, так и для систем с обобщенным обновлением. Также изучена модель с полным обновлением на примере системы без безопасной зоны в накопителе.

На защиту выносятся следующие результаты:

1. На основе СМО $G/M/1/\infty$ с пороговым механизмом полного обновления получены аналитические выражения и алгоритмы для расчета таких вероятностно-временных характеристик как: стационарные распределения числа заявок, вероятности сброса и обслуживания принятых в систему заявок, функции распределения времён пребывания в накопителе (системе) сброшенной, обслуженной и произвольной заявок для однопороговых и моделей.
2. На основе СМО $G/M/1/\infty$ с пороговым механизмом обобщенного обновления и безопасной зоной в накопителе получены аналитические выражения и алгоритмы для расчета таких вероятностно-временных характеристик как: стационарные распределения числа заявок, вероятности сброса и обслуживания принятых в систему заявок, функции

распределения времён пребывания в накопителе (системе) сброшенной, обслуженной и произвольной заявок для различных вариантов дисциплин обслуживания и обобщенного обновления.

3. Показано, что, выбирая тот или иной вид порогового механизма управления вероятностным сбросом заявок и меняя дисциплины обслуживания и обновления (обобщенного обновления) при неизменных начальных условиях, можно уменьшать (увеличивать) время пребывания в накопителе (системе) обслуженной (либо сброшенной) заявки, т.е. изменять значения задержки — одного из параметров AQM.

Методы исследования. В работе используются методы теории вероятностей, теории случайных процессов, теории массового обслуживания и математической теории телетрафика.

Теоретическая и практическая ценность. Разработанные математические модели и вычислительные алгоритмы, а также аналитические выражения, представленные в диссертации, могут использоваться для разработки новых алгоритмов активного управления очередями, а также для анализа и расчёта характеристик функционирования данных алгоритмов либо уже существующих. Полученные результаты ориентированы на применение в различных сетях и автоматизированных системах.

Реализация результатов работы. Результаты диссертации использовались в научно-исследовательских работах, проводимых в рамках гранта РФФИ № 19-07-00739 "Разработка аналитических методов расчета показателей качества функционирования узлов сетей последующих поколений,

использующих механизмы активного управления очередями, основанными на принципе обновления".

Достоверность и обоснованность. Обоснованность полученных результатов следует из того, что на всех этапах построения аналитических моделей, а также для аналитического и численного анализа полученных решений использовались строгие и проверенные математические методы теории вероятностей, теории случайных процессов и теории массового обслуживания.

Достоверность теоретических результатов диссертации подтверждена численными расчётами на основе программных модулей для анализа моделей СМО $GI/M/1/\infty$ с полным и обобщённым обновлением, а также соответствием полученных результатов (при ряде упрощающих модели предположений) уже известным результатам для систем с полным и обобщённым обновлением.

Соответствие паспорту специальности. Диссертационное исследование соответствует следующим разделам паспорта специальности 1.2.3. Теоретическая информатика, кибернетика, а именно: п.9 «Математическая теория исследования операций», п. 12 «Модели информационных процессов и структур»; п. 23 «Новые интернет - технологии, включая средства поиска, анализа и фильтрации информации».

Апробация работы. Результаты, полученные в ходе выполнения работы, были представлены на всероссийских конференциях с международным участием «Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем» (Российский

университет дружбы народов (РУДН), 2019, 2021-2023 гг.), международных конференциях «Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь. Distributed computer and communication networks: control, computation, communications (DCCN)» (Российский университет дружбы народов (РУДН), Институт проблем управления (ИПУ) РАН, Москва 2019, 2021-2022) и международной конференции «13th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, ICUMT 2021».

Публикации. По теме диссертации опубликовано 13 работ [62–67; 135; 136; 152; 162–164; 174] (из них 11 — тезисы и расширенные тезисы докладов на всероссийских и международных конференциях, 5 работ проиндексированы в международной базе цитирования Scopus [62; 65; 66; 135; 136], 2 — статьи, опубликованные в ведущих рецензируемых научных журналах и изданиях, определённых Высшей аттестационной комиссией [164; 174]).

Личный вклад. В работах, опубликованных в соавторстве, личный вклад соискателя состоит в проведении исследований, построении и анализе математической модели. Имитационные модели построены лично автором, также лично автором произведена интерпретация полученных результатов.

Структура и объем диссертации. Диссертация состоит из введения, трёх глав, разделённых на пункты, заключения, двух приложений и списка литературы. Текст диссертации изложен на 151 страницах (из них основной текст — 139 страниц), в приложении приведено 5 таблиц, также в тексте диссертации 3 рисунка. Список литературы содержит 190 наименований.

Глава 1. Алгоритмы управления очередями и модели с обновлением

1.1. Алгоритмы управления очередями (на примере алгоритмов семейства RED)

Активное управление очередями (Active Queue Management — AQM), согласно рекомендации Инженерного совета Интернета (Internet Engineering Task Force RFC 7567) [17] считается передовой практикой предотвращения (уменьшения) и контроля сетевых перегрузок в маршрутизаторах посредством некоторых правил (алгоритмов), которые, будучи реализованными в маршрутизаторах, контролируют длину буферной очереди в целях эффективного управления перегрузками системы. Основная цель AQM алгоритмов — это предотвращение возможной перегрузки или, если она уже возникла, уменьшение ее воздействия на систему (маршрутизатор).

За последние тридцать лет было предложено множество различных AQM алгоритмов [3; 6—9; 34; 45; 47; 49; 58; 72—74; 84; 95; 103; 106; 126] для надлежащего управления очередями для избежания таких нежелательных эффектов, как:

- переполнение буферной памяти маршрутизатора;
- избыточная или недостаточная загрузка канала передачи данных;
- сильные колебания задержки передачи данных;

- и другие показатели.

Стоит заметить, что среди множества предложенных AQM алгоритмов нет ни одного «идеального» алгоритма, превосходящего всех остальных по ключевым параметрам качества функционирования (Quality of Service — QoS), что обусловлено изменчивостью сетевого трафика и сетей, поэтому у каждого AQM алгоритма есть свои достоинства и недостатки.

AQM алгоритмы (такие, например, как алгоритмы семейства RED (Random Early Detection) [49], Explicit Congestion Notification (ECN) [121], Controlled Delay (CoDel) [36; 105]) основаны на использовании механизма «умного» вероятностного сброса пакетов, поступающих в систему, при переполнении буферной очереди (или при приближении некоторой усредненной длины очереди к заданному критическому значению).

Основное внимание будет уделено алгоритмам семейства RED, так как они обладают способностью сглаживать кратковременные всплески сетевого трафика и предупреждать длительные перегрузки сети за счет уведомления источников трафика о необходимости снижения интенсивности передачи информации [49; 86; 180]

Классический RED алгоритм [49] представляет из себя пороговый механизм вероятностного сброса поступающих пакетов с двумя заданными пороговыми значениями (минимальное — Q_{\min} и максимальное — Q_{\max}), функцией скользящей средней длины очереди \hat{Q} [49; 69; 180]:

$$\hat{Q}_{k+1} = (1 - w_q)\hat{Q}_k + w_q\hat{Q}_k, \quad k = 0, 1, 2, \dots,$$

где w_q , $0 < w_q < 1$ некоторый весовой коэффициент, определяемый пользователем, скользящей средней длины очереди. Согласно [49] алгоритм RED отслеживает среднюю длину очереди и сбрасывает (или маркирует, если совместно используется и ECN механизм) пакеты согласно функции сброса $p(\hat{Q})$ [49]:

$$p(\hat{Q}) = \begin{cases} 0, & 0 \leq \hat{Q} < Q_{\min}, \\ \frac{\hat{Q} - Q_{\min}}{Q_{\max} - Q_{\min}} p_{\max}, & Q_{\min} \leq \hat{Q} < Q_{\max}, \\ 1, & \hat{Q} \geq Q_{\max}, \end{cases}$$

здесь p_{\max} — максимальная доля сбрасываемых (маркируемых) пакетов при достижении верхнего порога Q_{\max} .

К достоинствам классического алгоритма RED можно отнести то, что сброс поступающих пакетов происходит не в случае переполнения очереди (отличие от механизма Tail Drop), также алгоритм RED способен сглаживать кратковременные «всплески» трафика, кроме того обеспечивается снижение расходов при обработке трафика. Однако же, как показано в [73], у классического алгоритма RED есть и ряд недостатков, например:

- способность обрабатывать лишь один поток трафика;
- необходимость периодической перенастройки;
- отсутствие достаточной информации относительно оптимальной настройки ключевых параметров;

- невозможность разделения по классам качества обслуживания (QoS).

Поэтому было предложено большое число модификаций алгоритма RED [8; 14; 16; 33; 34; 47; 50; 59; 90; 103; 108; 120; 124—126], многие из которых рассмотрены в обзорах [6; 7; 45; 180], а также представлены и в данной главе.

Наиболее простой вариант алгоритмов семейства RED — это EASY RED [59], для которого вероятность сброса зависит не от средневзвешенной длины очереди, а от текущей длины очереди. Основная задача алгоритма — как можно скорее уведомить отправителя о переполнении очереди. Параметры алгоритма: минимальное пороговое значение Q_{\min} и вероятность сброса p_{drop} , равная некоторой константе ($0 < p_{drop} < 1$), если текущая длина очереди превысила пороговое значение Q_{\min} .

Семейство адаптивных RED алгоритмов. Классический адаптивный RED (Adaptive RED или Active RED — ARED) [4; 50; 134; 180] алгоритм был разработан для того, поддерживать среднюю длину очереди в заранее определенных границах, путем более/менее активного сброса пакетов. Если экспоненциально взвешенная скользящая средняя длина очереди близка или больше максимального порога Q_{\max} , то поступающие пакеты будут сбрасываться с большей вероятностью, или, если экспоненциально взвешенная скользящая средняя длина очереди близка к минимальному порогу Q_{\min} , то вероятность сброса пакета будет снижаться. Таким образом, реализованы механизмы, позволяющие динамически изменять вероятность сброса (маркировки) входящих пакетов (с помощью двух вспомогательных параметров, отвечающих за увеличение (параметр α) и уменьшение

(параметр β) вероятности сброса) в зависимости от характера поведения трафика так, чтобы удерживать экспоненциально взвешенную скользящую среднюю длину очереди \hat{Q} между заданными минимальным и максимальным порогами ($Q_{\min} < \hat{Q} \leq Q_{\max}$). Итоговый вариант классической версии сформулирован в [50]. Дальнейшие разновидности — это алгоритмы RED-Worcester [7; 114] (поддерживается в заданных границах не только средняя длина очереди, но среднее значение задержки передачи данных), Refined ARED (RARED или RE-ARED) [79] (возможность изменения параметров α и β в зависимости от взвешенной средней длины очереди \hat{Q} для быстрого возвращения системы в стационарный режим функционирования), POWARED [115] (динамичное управление максимальной вероятностью (долей) сброса p_{\max} путем разбиения интервала $[Q_{\min}; Q_{\max}]$ на три части, для каждой из которых задается свое значение p_{\max} с учетом понижающего δ_1 или повышающего δ_1 коэффициентов), осторожный адаптивный RED (Cautious Adaptive Random Early Detection — CARED) [132] (динамичное изменение параметра p_{\max} в зависимости от уровня загрузки системы без введения дополнительных параметров), быстрый адаптивный RED (Fast Adapting RED — FARED) [131] (модификация алгоритма RE-ARED, улучшено поведение алгоритма RE-ARED при средней и, что более важно, высокой загрузке системы с помощью задания фиксированной нижней границы коэффициента β и зависящей от p_{\max} верхней границы коэффициента α). В алгоритме QARED [140] вместо линейной функции сброса используется квадратичная функция.

В представленных далее алгоритмах для более плавного и менее агрессивного сброса пакетов вводятся дополнительные пороговые значения.

В алгоритме Double Slope RED (DSRED) [148; 149; 180] в отличие от классического RED помимо пороговых Q_{\min} и Q_{\max} вводится еще одно пороговое значение Q_{mid} , причем $Q_{mid} = \frac{Q_{\max} + Q_{\min}}{2}$, что позволяет динамически изменять угол наклона кривой вероятности сброса, являющейся функцией уровня загруженности буфера. Вид вероятности $p(\hat{Q})$ определяется формулой:

$$p(\hat{Q}) = \begin{cases} 0, & 0 \leq \hat{Q} < Q_{\min}, \\ \alpha (\hat{Q} - Q_{\min}), & Q_{\min} \leq \hat{Q} < Q_{mid}, \\ 1 - \gamma + \beta (\hat{Q} - Q_{mid}), & Q_{mid} \leq \hat{Q} < Q_{\max}, \\ 1, & \hat{Q} \geq Q_{\max}, \end{cases}$$

где

$$\alpha = \frac{2(1 - \gamma)}{Q_{\max} - Q_{\min}}, \quad \beta = \frac{2\gamma}{Q_{\max} - Q_{\min}},$$

а γ — параметр, отвечающий за выбор угла наклона кривой функции сброса, \hat{Q} определяется как и в случае классического RED (1.1).

В алгоритме Gentle RED (GRED) [7; 48; 124] добавлен еще один параметр $2Q_{\max}$, что позволяет обеспечить более плавный рост значений функции

сброса от p_{\max} до 1 за пороговым значением Q_{\max} , то есть:

$$p(\hat{Q}) = \begin{cases} 0, & 0 \leq \hat{Q} < Q_{\min}, \\ \frac{\hat{Q} - Q_{\min}}{Q_{\max} - Q_{\min}} p_{\max}, & Q_{\min} \geq \hat{Q} < Q_{\max}, \\ \frac{\hat{Q} - Q_{\max}}{Q_{\max}} (1 - p_{\max}) + p_{\max}, & Q_{\max} \geq \hat{Q} < 2Q_{\max}, \\ 1, & \hat{Q} \geq 2Q_{\max}. \end{cases}$$

Более современная модификация алгоритма Gentle RED (GRED) — это IGRED [70] (уменьшено число параметров, определяющих значение функции сброса).

Для того, чтобы алгоритм RED стал более устойчивым к изменениям интенсивности трафика в работе [91] было предложено учитывать текущую длину очереди, а также долю сброшенных пакетов к поступившим за определенное число временных интервалов при определении вероятности сброса — Loss Ratio Based RED.

Устойчивый RED алгоритм (Robust RED — RRED) [125] был разработан для улучшения работы TCP протокола в случае DoS (Denial-of-Service) атак, в частности LDoS (Low-rate Denial-of-Service) атак. Таким образом, в RRED алгоритме реализована возможность обнаружения и фильтрации LDoS пакетов от остальных водящих пакетов данных до того, как они переполнят систему. Если отправитель (источник) обнаружил потерю (получил сообщение о сбросе) отправленного пакета данных, то устанавливается задержка

в передаче сообщений от данного источника, и все поступающие в систему пакеты (в течение короткого периода времени с момента обнаружения источником факта потери отправленного пакета) будут рассматриваться как LDoS пакеты.

Отличие алгоритма SDRED (State Dependent RED)[7; 127; 180] от классического алгоритма RED состоит в том, что очередь делится на семь частей, для каждой из которой устанавливается свой весовой коэффициент при вычислении средней длины очереди:

$$\hat{Q}_{i+1} = \begin{cases} (1 - w)\hat{Q}_i + wQ_i, & \hat{Q}_i < Q_{\max}, \quad 0,9R \leq \hat{Q}_i \leq R, \\ (1 - kw)\hat{Q}_i + kwQ_i, & Q_{\max} \leq \hat{Q}_i < 0,7R, \\ (1 - k^2w)\hat{Q}_i + k^2wQ_i, & 0,7R \leq \hat{Q}_i < 0,8R, \\ (1 - k^3w)\hat{Q}_i + k^3wQ_i, & 0,8R \leq \hat{Q}_i < 0,9R, \\ 1, & 0,9R \leq \hat{Q}_i \leq R, \end{cases}$$

здесь k — это константа, используемая для изменения параметра w в зависимости от загруженности очереди.

Функция (вероятность) сброса остается линейной в каждой из областей и определяется по формуле:

$$p(\hat{Q}) = \begin{cases} 0, & 0 \leq \hat{Q} < Q_{\min}, \\ \frac{\hat{Q} - Q_{\min}}{Q_{\max} - Q_{\min}} p_{\max}, & Q_{\min} \leq \hat{Q} < Q_{\max}, \\ \frac{\hat{Q} - Q_{\min}}{(Q_{\max} + 0,1R) - Q_{\min}} p_{\max}, & Q_{\max} \leq \hat{Q} < 0,7R, \\ \frac{\hat{Q} - Q_{\min}}{(Q_{\max} + 0,2R) - Q_{\min}} p_{\max}, & 0,7R \leq \hat{Q} < 0,8R, \\ \frac{\hat{Q} - Q_{\min}}{(Q_{\max} + 0,3R) - Q_{\min}} p_{\max}, & 0,8R \leq \hat{Q} < 0,9R, \\ 1, & 0,9R \leq \hat{Q} \leq R, \end{cases}$$

где R — максимальная длина очереди.

Отличие алгоритма Enhanced Random Early Detection (ERED) [7; 118; 180] от классического алгоритма заключается в совместном использовании экспоненциально взвешенной скользящей средней длины очереди \hat{Q} и текущей длины очереди Q , а также двух коэффициентов α и β ($\alpha > 1, \beta > 1$), что позволяет управлять экспоненциально взвешенной скользящей средней длиной очереди, уменьшая задержку реагирования на изменения текущей длины очереди:

$$\hat{Q}_{i+1} = \begin{cases} \frac{(1-w)}{\alpha} \hat{Q}_i + wQ_i, & Q < Q_{\min}, \\ (1-w)\hat{Q}_i + wQ_i, & Q_{\min} \leq Q < Q_{\max}, \\ (1-w)\hat{Q}_i + \frac{wQ_i}{\beta}, & Q_{\max} \leq Q. \end{cases}$$

Вид функции сброса остается таким же, как и в классическом RED алгоритме.

Совместное использование средней и текущей длин очереди реализовано и в алгоритме DyRED [37], но для динамического изменения верхнего порога Q_{\max} для раннего предотвращения перегрузки системы:

$$Q_{\max} = \begin{cases} Q_{\max}, & Q < Q_{\min}, \\ \frac{\hat{Q}}{\alpha}, & Q_{\min} \leq Q < Q_{\max}, \\ Q_{\max}^*, & Q_{\max} \leq Q. \end{cases}$$

Если текущая длина очереди Q меньше минимального порога Q_{\min} , то используется предустановленное значение верхнего порога Q_{\max} . При преодолении текущей длиной очереди минимального порога значение верхнего порога постоянно переопределяется на основе средней длины очереди и нормирующего параметра α (задается таким образом, чтобы избежать высоких значений задержки обслуживания). Если текущая длина очереди становится больше Q_{\max} , то значение верхнего порога задается максимально большим (для оптимального функционирования системы). Динамическое изменение верхнего порога позволяет уменьшить вероятность сброса и увеличить пропускную способность системы при допустимом уровне задержки обслуживания.

Алгоритм URED (Upper threshold RED)[109] аналогично алгоритму GRED вводит дополнительный верхний порог для более плавного роста вероятности сброса пакетов, если превышен порог Q_{\max} до максимального значения 1. При этом вероятность сброса остается линейной функцией.

В алгоритме Three-section RED (TRED) [35] область значений средней длины очереди между минимальным Q_{\min} и максимальным Q_{\max} порогами делится поровну на три части: $(Q_{\min}; Q_{\min} + \Delta)$, $[Q_{\min} + \Delta; Q_{\min} + 2\Delta)$ и $[Q_{\min} + 2\Delta; Q_{\max})$, где $\Delta = \frac{Q_{\max} - Q_{\min}}{3}$. Каждая из областей соответствует случаю легкой, средней или высокой загрузки системы. Вероятность сброса определяется следующим образом:

$$p(\hat{Q}) = \begin{cases} 0, & 0 \leq \hat{Q} < Q_{\min}, \\ 9 \left(\frac{\hat{Q} - Q_{\min}}{Q_{\max} - Q_{\min}} \right)^3 p_{\max}, & Q_{\min} \leq \hat{Q} < Q_{\min} + \Delta, \\ \frac{\hat{Q} - Q_{\min}}{Q_{\max} - Q_{\min}} p_{\max}, & Q_{\min} + \Delta \leq \hat{Q} < Q_{\min} + 2\Delta, \\ 9 \left(\frac{\hat{Q} - Q_{\max}}{Q_{\max} - Q_{\min}} \right)^3 p_{\max} + p_{\max}, & Q_{\min} + 2\Delta \leq \hat{Q} < Q_{\max}, \\ 1, & Q_{\max} R \leq \hat{Q}. \end{cases}$$

При этом функция сброса в двух из трех областей является нелинейной функцией.

Примеры алгоритмов семейства RED, где функция сброса является нелинейной: PRED (parabola RED) [73], HRED (hyperbola RED) [68].

Для поддержки загрузки системы в заданных границах (чтобы не было излишнего сброса поступающих пакетов при кратковременном росте трафика) были предложены две разновидности классического алгоритма RED. Первый алгоритм — это динамический RED (Dynamic RED — DRED) [7; 16; 120], в котором функция (вероятность) сброса зависит от загрузки системы, что позволяет поддерживать среднюю длину очереди в заданных границах. Вторым алгоритмом — это SRED (Stabilized RED) [108; 180] в котором функция сброса зависит от числа активных соединений и текущей длины очереди.

Для работы с различными типами трафика также были разработаны разновидности алгоритма RED. Первая такая разновидность — это алгоритм Weighted RED (WRED) [6; 7; 33; 45; 180], разработанный для обработки трафика различных приоритетов. В данном алгоритме для каждого типа трафика задаются свои наборы (совпадающие или не совпадающие) управляющих параметров (пороговые значения Q_{\min} и Q_{\max} , параметр p_{\max}). Дальнейшей модификацией WRED является алгоритм Distributed Weighted RED (DWRED)[31]. Поточковый RED (Flow RED — FRED) [6; 7; 45; 90; 120] основан на том, что для каждого типа трафика задается свои минимальные и максимальные границы в общей очереди (сколько от общей очереди пакеты каждого вида могут занимать) и для каждого потока устанавливается своя вероятность сброса (зависимости от занятия буфера пакетами этого потока). При этом для потоков, превышающих свою верхнюю границу, предусмотрен более агрессивный (по сравнению с предыдущим алгоритмом) сброс. Сбалансированный (Balanced RED — BRED) RED [6; 7; 14; 45] отличается от FRED тем, что вероятность сброса для каждого типа

трафика определяется не только числом пакетов этого типа трафика в очереди, но и числом принятых пакетов данного типа трафика с момента последнего сброса пакета данного трафика. Алгоритм SF-RED [6; 7; 139; 180] также предназначен для работы с различными типами трафика, но в отличие от предыдущих алгоритмов помимо единой физической очереди используются еще и виртуальные очереди, каждая из которых соответствует определенному классу входящего трафика. Для каждой виртуальной очереди задаются свои пороговые значения и определяется своя вероятность сброса (как функция экспоненциально взвешенной скользящей средней длины соответствующей виртуальной очереди). При этом минимальный порог для каждой виртуальной очереди может изменяться (увеличиваться или уменьшаться) на коэффициент α в зависимости от соотношения экспоненциально взвешенной скользящей средней длины соответствующей виртуальной очереди и некоторой области для целевого значения экспоненциально взвешенной скользящей средней длины соответствующей физической очереди. В алгоритме Light-weight Flow information RED (LwFRED) [71], предназначенном для обработки разных типов входящего трафика, реализована идея объединения нескольких потоков в группы с последующим определением вероятности сброса для каждой группы.

Отличным от рассмотренных ранее можно считать алгоритм Rate-Based n-RED [7; 38], использующий в качестве аргумента функции сброса не среднюю длину очереди, а интенсивность поступления пакетов. Интенсивность поступления сравнивается с интенсивностью обслуживания (пропускной

способностью канала) и в случае превышения принимается решение о сбросе. Для динамического изменения вероятности сброса введен коэффициент занятости очереди.

К семейству алгоритмов RIO (RED with In|Out) относят первоначальный алгоритм RIO [7; 32; 180] и его разновидности RIO-C (RED with In|Out and Coupled Virtual Queues) [43], RIO-DC (RED with In|Out and Decoupled Virtual Queues) [7; 43; 180], адаптивный RIO (ARIO)-[1; 7; 107] и Rate-based RIO (Rb-RIO) [7; 26]. Идея алгоритма RIO состоит в том, что поступающие пакеты делятся на типа: приоритетные (профильные) In-пакеты и неприоритетные (непрофильные) — Out-пакеты. Для пакетов каждого типа задается свой набор RED-Параметров (пороговые значения Q_{\max} и Q_{\min} , максимальная вероятность сброса p_{\max}). Разница состоит только в том, что: во, первых, для Out-пакетов пороговые значения меньше, чем для In-пакетов, и, во-вторых, решение о сбросе Out-пакетов основывается на значениях средней длины общей очереди, а решение о сбросе In-пакетов — на значениях средней длины виртуальной очереди, состоящей только из профильных пакетов.

В алгоритме RIO-C [7; 43; 180] увеличено число приоритетных классов для поступающих пакетов. Для каждого i -го класса приоритета определяются свои пороговые значения Q_{\max_i} и Q_{\min_i} , максимальная вероятность сброса p_{\max_i} (снова пороговые значения для менее приоритетных классов устанавливаются ниже, чем для более приоритетных классов), но отличие от RIO состоит в том, что средняя длина очереди для пакета класса i определяется числом пакетов данного класса, а также числом пакетов более приоритетных классов. Тем самым, реализован механизм изменения

вероятности сброса пакета при росте интенсивности более приоритетных пакетов. Для алгоритма RIO-DC [7; 43; 180] для пакетов i -го класса приоритета средняя длина очереди вычисляется только по числу пакетов данного класса в очереди (отличие от алгоритма WRED [33]).

Адаптивный RIO [1; 7; 107] берет за основу идеи алгоритма RIO-C [7; 43; 180] (работа с трафиком различных приоритетов), только для каждого типа трафика вместо классического RED используется алгоритм ARED [7; 50], позволяющий менять пороговые значения в зависимости от интенсивности трафика.

Алгоритм Rb-RIO [7; 26] реализован для трафика трех приоритетов, причем вероятность сброса как и в алгоритме Rate-Based n-RED [7; 38]: 1) если суммарная интенсивность поступления трех классов трафика $\lambda = \lambda_1 + \lambda_2 + \lambda_3$ меньше интенсивности обслуживания μ , то все пакеты принимаются в систему; 2) если $\lambda > \mu$, но суммарная интенсивность поступления двух более приоритетных классов меньше интенсивности обслуживания $\lambda_2 + \lambda_3 < \mu$, то пакеты двух более приоритетных классов принимаются в систему, а менее приоритетного сбрасываются с вероятностью $\frac{\lambda - \mu}{\lambda_1}$; 3) если $\lambda_2 + \lambda_3 > \mu$, но $\lambda_3 < \mu$, то пакеты с самым высоким приоритетом принимаются в систему, менее приоритетные пакеты сбрасываются с вероятностью $\frac{\lambda_2 + \lambda_3 - \mu}{\lambda_2}$, все неприоритетные (с самым низким приоритетом) пакеты сбрасываются из системы; 4) если $\lambda_3 > \mu$, то пакеты наивысшего приоритета сбрасываются с вероятностью $\frac{\lambda_3 - \mu}{\lambda_3}$, пакеты остальных классов сбрасываются с вероятностью равной 1.

В работе [2] предложена разновидность алгоритма RED — Average Queue Length and Change Rate-RED (AC-RED), в которой в качестве параметров, определяющих поведение функции сброса, используется не только средняя длина очереди, но и интенсивность изменения средней длины очереди:

$$\hat{Q}(t+1) = (1 - w_q)\hat{Q}(t) + w_qQ(t+1),$$

$$d\hat{Q}(t+1) = (1 - w_q)d\hat{Q}(t) + w_q(Q(t+1) - Q(t)).$$

Здесь w_q — весовой параметр, $\hat{Q}(t)$ — значение средней длины очереди в момент времени t , $Q(t)$ — значение текущей длины очереди в момент времени t , $d\hat{Q}(t)$ — интенсивность изменения средней длины очереди в момент времени t . Помимо пороговых значений Q_{\min} и Q_{\max} вводится еще одно пороговое значение Q_{mid} ($Q_{\min} < Q_{mid} < Q_{\max}$) аналогично алгоритму DSRED [148; 149; 180]. Но в отличие от [148; 149; 180] значение порога Q_{mid} динамично изменяется:

$$Q_{mid} = \begin{cases} Q_{mid} - \frac{2}{3}, & d\hat{Q}(t) > 0.005, \\ Q_{mid}, & 0 \leq d\hat{Q}(t) \leq 0.005, \\ Q_{mid} + \frac{3}{2}, & d\hat{Q}(t) < 0. \end{cases}$$

В зависимости от области значений параметра $d\hat{Q}(t)$ по разному определяется вероятность сброса. Если $d\hat{Q}(t) > 0.005$, то

$$p(\hat{Q}) = \begin{cases} 0, & \hat{Q} < Q_{\min}, \\ \left(\frac{2(\hat{Q} - Q_{\min})}{Q_{mid} - Q_{\min}} \right)^2 \frac{2}{3} p_{\max}, & Q_{\min} \leq \hat{Q} < Q_1, \\ \left(\frac{2\hat{Q} - (Q_{mid} + Q_{\min})}{Q_{\max} - Q_{\min}} \right)^{\frac{1}{3}} 2p_{\max} + \frac{2}{3} p_{\max}, & Q_1 \leq \hat{Q} \leq Q_2, \\ 1, & \hat{Q} > Q_2. \end{cases}$$

Если $0 \leq d\hat{Q}(t) \leq 0.005$, то

$$p(\hat{Q}) = \begin{cases} 0, & \hat{Q} < Q_{\min}, \\ \left(\frac{\hat{Q} - Q_{\min}}{Q_{mid} - Q_{\min}} \right)^3 \frac{1}{2} p_{\max}, & Q_{\min} \leq \hat{Q} < Q_{mid}, \\ \left(1 + \left(\frac{\hat{Q} - Q_{mid}}{Q_{\max} - Q_{mid}} \right)^{\frac{1}{3}} \right) \frac{1}{2} p_{\max}, & Q_{mid} \leq \hat{Q} < Q_{\max}, \\ \left(1 + \left(\frac{\hat{Q} - Q_{AC}}{Q_{AC} - Q_{\max}} \right)^{\frac{3}{2}} \right) (1 - p_{\max}), & Q_{\max} \leq \hat{Q} \leq Q_{AC}, \\ 1, & \hat{Q} > Q_{AC}. \end{cases}$$

Если $d\hat{Q}(t) < 0$, то

$$p(\hat{Q}) = \begin{cases} 0, & \hat{Q} < Q_{\min}, \\ \left(\frac{\hat{Q} - Q_{\min}}{Q_{mid} - Q_{\min}} \right)^3 \frac{1}{2} p_{\max}, & Q_{\min} \leq \hat{Q} < Q_{mid}, \\ \left(1 + \left(\frac{\hat{Q} - Q_{mid}}{Q_{\max} - Q_{mid}} \right)^{\frac{1}{2}} \right) \frac{1}{2} p_{\max}, & Q_{mid} \leq \hat{Q} < Q_{\max}, \\ 1, & \hat{Q} \geq Q_{\max}. \end{cases}$$

Здесь $Q_1 = \frac{Q_{mid} + Q_{\min}}{2}$, $Q_2 = \frac{Q_{mid} + Q_{\max}}{2}$, а Q_{AC} — дополнительный верхний порог, как и в алгоритме URED [109].

Один из недавно предложенных вариантов алгоритма RED — это Beta RED [103] алгоритм, продолжающий заложенную в работах [2; 8; 35; 47; 122; 147; 150] идею использования нелинейной функции (вероятности) сброса поступающих в систему пакетов. Предложено использовать плотность бета-распределения в качестве коэффициента, на который умножается максимальная вероятность сброса p_{\max} . При этом параметры бета-распределения зависят от предопределенного целевого значения длины очереди. На основе алгоритма BetaRED созданы два алгоритма — Adaptive Beta RED (ABetaRED [103], реализована возможность менять значения параметра p_{\max} в зависимости от соотношения средней длины очереди и целевого значения длины очереди) и Dynamic Beta RED (DBetaRED [103],

введение нового параметра — виртуальное целевое значение длины очереди — с последующим его изменением в зависимости от значений средней длины очереди).

Другой из недавно предложенных — это алгоритм Aggressive Random Early Detection (AgRED) [74], разработанный для сетей 5G. Вероятность сброса поступающих пакетов также как и в [2; 8; 35; 47; 103; 122; 147; 150] является нелинейной и определяется по формуле (для $Q_{\min} \leq \hat{Q} \leq Q_{\max}$):

$$p(\hat{Q}) = \frac{1}{1 + P_d}, \quad P_d = \frac{e^{-\hat{Q}} - e^{-Q_{\min}}}{e^{-Q_{\max}} - e^{-Q_{\min}}} \cdot e^{-p_{\max}},$$

где \hat{Q} — средняя длина очереди, Q_{\min} и Q_{\max} — пороговые значения, p_{\max} — максимальная вероятность сброса.

Более подробно ознакомиться с вариантами AQM алгоритмов, их описаниями, анализом производительности, сравнением можно ознакомиться на сайте [128], а также в работах [6; 7; 44; 46].

1.2. Модели с полным и обобщенным обновлением

Впервые термин «обновление» был введен в работе [88] как своеобразный вариант дисциплины обслуживания заявок, согласно которой заявка, находящаяся на приборе, в момент окончания своего обслуживания могла либо сбросить все другие заявки из накопителя с некоторой ненулевой вероятностью q (вероятность обновления) и уйти из системы, либо с дополнительной (обратной) вероятностью $p = 1 - q$ покинуть систему массового

обслуживания (СМО), не сбрасывая заявки из накопителя. В работе доказано, что условием существования стационарного распределения в системах с бесконечным объемом накопителя является отличие от нуля вероятностей p и q , а также рассмотрены одноканальные системы массового обслуживания с дисциплиной обслуживания FIFO (First In First Out), для которых получен ряд результатов:

- Система с пуассоновским входящим потоком и накопителем ограниченного размера $M|M^R|1|N$ (в классификации Башарина-Кендалла и обозначениях автора), для которой выведена система уравнений равновесия, с помощью производящей функции получены аналитические выражения для вероятности простоя системы (π_0 - в обозначениях автора) и вероятности переполнения системы π_N , а так же формула для расчета стационарного распределения числа заявок в системе.
- Система с групповым пуассоновским входящим потоком и накопителем бесконечной емкости $M^X|M^R|1|\infty$. Выведена система уравнений равновесия, построена производящая функция числа заявок в системе, из которой найдена вероятность простоя системы.
- Система с рекуррентным входящим потоком и накопителем конечной емкости $GI|M^R|1|N$. С помощью дополнительной переменной (время, прошедшее с момента прихода последней заявки) строится марковский процесс и получены выражения для расчета вероятностного распределения по времени.

- Система с рекуррентным входящим потоком и накопителем бесконечной емкости $GI|M^R|1|\infty$. С помощью дополнительной переменной (время, прошедшее с момента прихода последней заявки) строится марковский процесс и получены выражения для расчета вероятностного распределения по времени.

По предложению П.П. Бочарова был рассмотрен вариант обновления с повторным обслуживанием, когда заявка, опустошившая накопитель, остается в системе. Система подобного типа была рассмотрена в работах [88; 160]. В работе [160] исследовалась система массового обслуживания с n приборами, накопителем ограниченной емкости, пуассоновским входящим потоком и экспоненциальным распределением времени обслуживания заявки на приборе. Для вывода стационарного распределения числа заявок в системе было предложено использовать алгоритмы из работы [183], разработанные для обобщенного процесса рождения и гибели (также подробно представлены в [19; 119; 161]). Показано, что матричные алгоритмы из [19; 119; 161; 183]) применимы как для систем массового обслуживания с обновлением, так и для систем массового обслуживания с обновлением и повторным обслуживанием — получены в явном виде выражения для расчета стационарного распределения (по времени) числа заявок в системе. В качестве частных случаев рассмотрены марковские системы $M|M^R|n|r$ ($r < \infty$) и $M|M^{RR}|n|r$ (здесь верхний индекс R (renovation) обозначает обновление без повторного обслуживания, а RR (renovation with reservice) — обновление с повторным

обслуживанием). Для каждого из частных случаев представлены стационарные распределения (по времени) числа заявок в системе, формулы для расчета среднего числа заявок в системе и накопителе.

Если в системе накопитель неограниченного размера, то предложенные в [160] алгоритмы расчета стационарного распределения числа заявок для марковских систем ([19; 119; 161; 183]) могут быть переформулированы как для полного обновления, так и для полного обновления с повторным обслуживанием.

Профессором А.В. Печинкиным понятие обновления было развито — введен термин «обобщенное обновление»: заявка в момент окончания своего обслуживания и ухода из системы может сбросить из накопителя не все заявки, а некоторое произвольное число заявок k с заданной вероятностью $q(k)$ (при этом, как правило, сбрасывается группа из k последовательно расположенных заявок), если в накопителе находится более k заявок, либо может полностью опустошить накопитель (если в нем было менее k заявок), либо сбросить все заявки из накопителя *вне зависимости от их числа), либо просто покинуть систему, не сбрасывая ни одной заявки из накопителя, с вероятностью $p = 1 - q - \sum_{i=0}^{\infty} q(i)$ (для системы с неограниченным накопителем) или $p = 1 - \sum_{i=1}^r q(i)$ (для системы с ограниченным накопителем).

В работе [172] исследуется многолинейная система массового обслуживания с рекуррентным входящим потоком заявок, экспоненциальным обслуживанием заявок на приборах, накопителем конечной емкости и обобщенным обновлением $GI|M|n|r$. По моментам поступления заявок в систему

строится вложенная цепь Маркова, выведена система уравнений равновесия и показано, как с помощью метода уменьшения числа состояний [19; 119; 161]) вычислять стационарное распределение вероятностей по построенной вложенной цепи Маркова. Осуществлен переход от стационарного распределения вероятностей по вложенной цепи Маркова к стационарному по времени распределению вероятностей числа заявок в системе.

Так как потери заявок могут наступать не только из-за переполнения системы, но и из-за механизма обобщенного обновления, то дополнительно вычислены вероятность потери принятой в систему заявки, вероятность сброса заявки из накопителя первой обслужившейся, а также общая вероятность потери заявки (учитывающая как переполнение системы, так и обобщенное обновление).

Особый интерес в подобных системах представляют временные характеристики, а именно:

- стационарное распределение времени ожидания начала обслуживания (только для несброшенных заявок)
- стационарное распределение времени пребывания в накопителе сброшенной заявки (время ожидания сброса/потери);
- среднее время ожидания начала обслуживания;
- среднее время пребывания в накопителе до сброса.

Для моделей с обобщенным обновлением в зависимости от того, в каком порядке поступившие в систему заявки обслуживаются и сбрасываются,

рассматриваются несколько вариантов комбинаций обслуживания и сброса (обобщенного обновления):

- Либо принятые в систему заявки обслуживаются начиная с первой пришедшей (дисциплина обслуживания FIFO — First In First Out) и сбрасываются группой (группа начинается с первой заявки в накопителе) согласно порядку поступления (First) — вариант FIFO/First;
- Либо принятые в систему заявки обслуживаются согласно порядку поступления (FIFO), но сбрасываются последовательной группой, начиная с последней пришедшей заявки (Last) — комбинация FIFO/Last;
- Либо поступившие в систему заявки обслуживаются начиная с последней пришедшей (дисциплина обслуживания LIFO — Last In First Out), но сбрасываются последовательной группой, начиная с первой заявки в накопителе (First) — комбинация LIFO/First;
- Либо поступившие в систему как обслуживаются, так и сбрасываются (последовательной группой), начиная с последней пришедшей — вариант LIFO/Last;
- Либо заявки обслуживаются согласно дисциплине FIFO, но сбрасывается группа заявок с любого места в накопителе (Random) — вариант FIFO/Random;
- Либо заявки обслуживаются согласно дисциплине LIFO, а сброс осуществляется с произвольного места в накопителе — LIFO/Random.

Первый из представленных вариантов — FIFO/First, был рассмотрен в работе [172] и для него было доказано, что как стационарная функция распределения времени ожидания начала обслуживания, так и стационарная функция распределения времени пребывания в накопителе сброшенной заявки представляют взвешенные по вероятности суммы функций распределения Эрланга, параметрами которого задаются числом обслуженных заявок в системе и интенсивностью обслуживания.

Затем, в работе [173] были рассмотрены такие комбинации дисциплин обслуживания и обобщенного обновления, как FIFO/Last, LIFO/First и LIFO/Last. Для каждого из указанных вариантов были получены в терминах преобразований Лапласа-Стилтьеса (ПЛС) стационарная функция распределения времени ожидания начала обслуживания и стационарная функция распределения времени пребывания в накопителе сброшенной заявки. В работе [169] для комбинаций FIFO/Last, LIFO/First и LIFO/Last дисциплин обслуживания и обобщенного обновления были получены средние характеристики.

Следующая система, результаты исследования которой представлены в работах [143; 168; 171; 177], была система вида $GI|M|n|\infty$, исследование которой также проводилось с помощью вложенной по моментам непосредственно перед поступлением новой заявки в систему цепи Маркова.

В работе [171] в явном виде было найдено стационарное по вложенной цепи Маркова распределение числа заявок в системе, для которого было доказано, что данное распределение представимо в геометрическом виде, если все приборы в системе заняты. Осуществлен переход к стационарному по

времени распределению числа заявок в системе, представлены формулы для расчета стационарного по времени распределения числа заявок в системе.

Также в работах [143; 171], как и в [172] введены и найдены вероятности того, что поступающая в систему заявка либо в дальнейшем будет обслужена, либо обслужена. Но в отличие от [172] вероятности получены в явном виде.

Особое внимание уделено временным характеристикам системы. Для варианта обслуживания и обновления FIFO/First в работе [171] доказано, что как время пребывания в накопителе сброшенной заявки, так и время ожидания начала обслуживания имеют экспоненциальные распределения (для обслуженной заявки — со скачком в момент времени ноль). Были выведены формулы для расчета средних временных характеристик как обслуженной, так и сброшенной заявок. В [143] представлены основные результаты [181].

Для вариантов дисциплин обслуживания и обобщенного обновления FIFO/Last, LIFO/First и LIFO/Last в работе [171] с помощью ПЛС и двойной производящей функции выведены временные характеристики системы $GI|M|n|\infty$ были получены аналитические выражения как для стационарных функций распределения времен пребывания в накопителе сброшенной и обслуженной заявок, так и для средних временных характеристик.

Варианты FIFO/Random и FIFO/Random были рассмотрены в работе [146] и был представлен аналитический метод расчета среднего времени ожидания начала обслуживания принятой и несброшенной заявки, а также среднее время пребывания в системе принятой, но в дальнейшем сброшенной заявки.

В работах [168; 170] для вариантов FIFO/First, FIFO/Last, LIFO/First и LIFO/Last дисциплин обслуживания и обобщенного обновления была исследована зависимость временных характеристик (для обслуженной и сброшенной заявок) системы $GI|M|n|\infty$ от вероятностного распределения обобщенного обновления. Это позволяет выбрать наиболее оптимальную стратегию, минимизирующую время ожидания. В качестве распределения вероятностей обобщенного обновления были выбраны геометрическое и пуассоновское распределения. Для вариантов обслуживания и обновления FIFO/First, LIFO/First и LIFO/Last для различных случаев входящего потока (пуассоновский, эрланговский, поступление подчиняется гамма-распределению) было исследовано поведение средних времен пребывания в накопителе обслуженной и сброшенной заявок относительно средних времен для варианта FIFO/Last, выбранного в качестве базового как наиболее типичного для алгоритмов семейства RED активного управления очередями.

В частности, было показано [168; 170], что для варианта FIFO/Last характерна наибольшее среднее время пребывания в накопителе обслуженной заявки в системе по сравнению с другими рассмотренными вариантами обслуживания и обновления. Также в [168; 170] было показано, что если вероятности обновления подчиняются геометрическому распределению, то:

- для вариантов FIFO/First и LIFO/Last обслуживания и обобщенного обновления для обслуженной и сброшенной заявок средние временные характеристики вне зависимости от загрузки системы;

- для варианта (FIFO/Last) обслуживания и обобщенного обновления среднее время пребывания в накопителе сброшенной («убитой» в терминах работ [168; 170]) заявки принимает наименьшие значения, а для варианта LIFO/First обслуживания и обобщенного обновления — наибольшие значения;
- для обслуженной заявки среднее время ожидания начала обслуживания принимает наименьшие значения при варианте LIFO/First обслуживания и обобщенного обновления, а наибольшие — для варианта FIFO/Last.

Если вероятности обобщенного обновления подчиняются распределению Пуассона, то:

- поведение средних времен зависит как от типа входящего потока, так и от загрузки системы (особенно это видно в случае гамма распределения входящего потока);
- наибольшие значения среднее время пребывания в накопителе обслуженной заявки принимает либо в случае комбинации FIFO/Last обслуживания и обновления (эрланговский или пуассоновский входящий поток), либо, если интервалы времени между последовательными моментами поступления подчинены гамма распределению, для дисциплины LIFO/Last обслуживания и обновления, а наименьшие значения (при любых рассмотренных вариантах входящего потока) — для дисциплины LIFO/First обслуживания и обновления.

Работа [176] посвящена применению модели обобщенного обновления для анализа RED-подобного алгоритма с помощью системы массового обслуживания $G|M|n|\infty$. Для соответствия классическому алгоритму RED [49] в накопителе определяются два пороговых значения q_{min} и q_{max} , $q_{min} < q_{max}$. В зависимости от уровня заполнения накопителя поступающими в систем пакетами в момент передачи из системы пакета принимается решение о сбросе принятых в накопитель пакетов, причем вероятность сброса (функция сброса, согласно терминологии работы [176]) $p_d(\hat{q})$ определяется формулой:

$$p_d(\hat{q}) = \begin{cases} 0, & 0 \leq \hat{q} \leq q_{min}, \\ p(\hat{q}), & q_{min} < \hat{q} \leq q_{max}, \\ 1, & \hat{q} > q_{max}. \end{cases}$$

Исследование рассматриваемой в работе [176] системы также (как и в других работах по системам с обобщенным обновлением) проводилось с помощью вложенной по моментам поступления в систему заявок цепи Маркова, для которой было получено стационарное распределения числа заявок, вероятности обслуживания или сброса поступающей заявки, временные характеристики (в терминах ПЛС и средние значения) для обслуженной заявки. Была проанализирована зависимость стационарного распределения по вложенной цепи Маркова числа заявок в системе, а также вероятности сброса принятой в систему заявки, среднего времени ожидания начала обслуживания от интенсивности входящего потока.

В дальнейшем, использование систем массового обслуживания с обновлением или обобщенным обновлением для анализа характеристик систем активного управления очередями типа Random Early Detection (RED) рассматривалось в работах [82; 83; 85]. Для системы $M/D/1/N$ были получены аналитические выражения для таких стационарных характеристик, как распределение числа заявок в системе и вероятность потери поступившей в систему заявки. В результате численного моделирования было доказано, что при определенных начальных параметрах модели с обобщенным обновлением, несмотря на коренное различие в механизмах вероятностного сброса как алгоритма RED, а так же меняя значение вероятностей обновления $q(i)$, можно обеспечить необходимый уровень эффективности функционирования рассматриваемой модели и получить результаты, сопоставимые с результатами модели алгоритма RED (с теми же исходными параметрами). При этом отмечено, что основная проблема при реализации моделей с обновлением (обобщенным обновлением) связана с определением наиболее оптимальных значений вероятностей обновления. То есть, внедрение механизма обновления в качестве вероятностного механизма сброса поступающих пакетов данных требует предварительной и (или) оперативной настройки параметров обновления. авторы в работах [82; 83; 85] подчеркивают необходимость дальнейших исследований моделей с обновлением для случаев нескольких входящих потоком (в том числе и с различными приоритетами), а также построение моделей на основе реальных данных сетевых узлов (маршрутизаторов).

В работе [83] предложен новый вариант механизма обновления — обновление с задержкой, смысл которого заключается в том, сброс поступивших заявок может произойти не в каждый момент окончания обслуживания на приборе (приборах), а только после некоторого заданного числа окончаний обслуживания. С помощью численного эксперимента показано, что поведение исследуемой системы с новым алгоритмом обновления аналогично поведению системы с классическим RED алгоритмом (при достаточно долгом функционировании системы). Представлены выражения для расчета таких характеристик как вероятность потери поступающего в систему пакета (заявки) и среднее число заявок в системе.

Исследование систем с обновлением для случая нескольких входящих потоков различных приоритетов было предпринято в работах [144; 167; 175]. В работе [175] была осуществлена попытка исследования однолинейной системы массового обслуживания с накопителем неограниченной емкости, в которую поступают два пуассоновских потока различной интенсивности, причем заявки одного потока (приоритетные заявки) в момент окончания обслуживания на приборе с некоторой отличной от нуля вероятностью q сбросить все заявки второго типа (неприоритетные) из накопителя и покинуть систему, либо с дополнительной вероятностью $p = 1 - q$ просто покинуть систему. Время обслуживания на приборе имеет экспоненциальное распределение, параметр μ которого зависит от типа обслуживаемой заявки (ее приоритета). Для исследования системы был построен трехмерный марковский процесс, учитывающий не только число заявок каждого типа в накопителе, но и тип обслуживаемой на приборе заявки. Доказано, что

маргинальное распределение числа приоритетных заявок в накопителе, если на приборе находится неприоритетная заявка, имеет геометрический вид и зависит от вероятности того, что система пуста (вероятность простоя). Вероятность простоя системы зависит от маргинальной вероятности $p_{1,1}$, того, что в накопителе одна приоритетная заявка и на приборе также находится приоритетная заявка. Также, в [175] приведены формулы для расчета вероятности обслуживания неприоритетной заявки, вероятности ее сброса из накопителя приоритетной заявкой, стационарного распределения времени пребывания в накопителе приоритетных заявок. Но, так как не найдено в явном виде совместное стационарное по времени распределение числа заявок обоих типов, то вышеупомянутые вероятностно-временные характеристики в окончательном виде получить авторам не удалось.

Исследование модели [175] было продолжено в работе [167]. Получено стационарное распределение вероятностей системы, в терминах производящей функции выведена формула для расчета вероятности обслуживания неприоритетных заявок, а также формула для расчета среднего числа неприоритетных заявок. Кроме того получена такая характеристика как вероятность простоя системы.

В работе [144] была построена с помощью двумерного марковского процесса упрощенная модель, в которой приоритетная заявка, заканчивая обслуживание, сбрасывает из накопителя неприоритетную заявку с вероятностью, равной единице. Для рассмотренной упрощенной системы в явном виде были получены такие стационарные характеристики, как число приоритетных заявок в системе, вероятность простоя системы, вероятность

обслуживания и вероятность потери неприоритетной заявки, стационарное распределение (в терминах ПЛС) времени ожидания начала обслуживания и среднее время ожидания начала обслуживания неприоритетной заявки. Но даже для упрощенной модели авторам не удалось вывести в явном виде выражения для таких характеристик, как стационарное распределение числа неприоритетных заявок в системе и стационарное распределение времени ожидания начала обслуживания приоритетной заявки.

Исследование систем с обновлением и повторным обслуживанием (дообслуживанием), начатое в работе [160], было продолжено в работах [145; 181] на примере системы массового обслуживания $GI|M^{RR}|1|\infty$. Исследование снова проводилось с помощью вложенной по моментам поступления цепи Маркова, и в рамках проводимого исследования авторами были получены следующие результаты:

- стационарное по вложенной цепи Маркова распределение числа заявок в системе, которое имеет геометрический вид [181];
- вероятность обслуживания поступившей заявки (в терминах статьи — вероятность передачи поступивших в систему пакетов) [181],
- показано, что для несброшенной заявки время ожидания начала обслуживания имеет экспоненциальное распределение с скачком в нулевой момент времени [181], в работе [145] представлено выражение для среднего времени ожидания начала обслуживания;

- показано, что время пребывания в накопителе сброшенной заявки также подчинено экспоненциальному распределению. Кроме того для сброшенной заявки получено выражение для расчета среднего времени пребывания в накопителе [145];
- стационарное распределение времени пребывания в накопителе произвольной заявки и среднее время пребывания в накопителе произвольной заявки [145].

Исследование систем с обновлением и повторным обслуживанием было продолжено в [24; 130], где были изучены модели с обобщенным обновлением, для которых получены аналитические выражения для расчета стационарного распределения числа заявок в системе, вероятности обслуживания принятой в систему заявки, вероятности сброса принятой в систему заявки, временные характеристики как для обслуженных, так и для сброшенных заявок.

Также была рассмотрена система вида $M|G|1|\infty$ с обновлением в работах [21; 22], которая оказалась более сложной для исследования и для которой были получены только производящая функция числа заявок в системе, вероятность простоя системы, среднее число заявок в системе, вероятность отсутствия потерь, распределение времени ожидания начала обслуживания и среднее время ожидания обслуживания для несброшенных заявок.

Глава 2. Анализ систем с пороговым механизмом обновления

2.1. Система массового обслуживания $G/M/1/\infty$ с одним порогом, безопасной зоной и обновлением.

В данной главе будут рассмотрены модели с полным обновлением [87; 88; 145; 160; 170] (когда заявка, закончившая обслуживание и покидающая систему, либо полностью опустошает накопитель и покидает систему с заданной вероятностью q , либо с дополнительной вероятностью $p = 1 - q$ покидает систему, не опустошая накопитель), в которые введен пороговый механизм включения вероятностного сброса.

2.1.1. Описание системы, матрица переходных вероятностей вложенной цепи Маркова

Рассмотрим систему массового обслуживания (представлена на рис. 2.1), состоящую из одного обслуживающего устройства (время обслуживания на котором подчинено экспоненциальному распределению с параметром μ) и накопителя неограниченной емкости, в котором определено пороговое значение Q_1 . В систему поступает рекуррентный поток заявок, функция распределения времени между последовательными моментами поступления — $A(x)$.

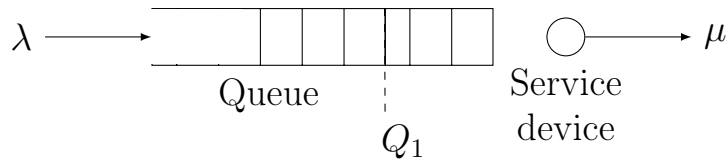


Рис. 2.1.. Однопороговая система с безопасной зоной в накопителе

Определим механизм обновления следующим образом: если число заявок в накопителе меньше или равно пороговому значению Q_1 , то никакого сброса заявок нет. Если же число заявок i в накопителе становится больше порогового значения Q_1 , то либо с вероятностью p заявка, находящаяся на приборе, после окончания обслуживания просто покидает систему, либо с вероятностью $q = 1 - p$ заявка в момент ухода из системы сбрасывает все заявки из накопителя, начиная с $Q_1 + 1$ от начала накопителя (таким образом, первые Q_1 находящиеся в накопителе заявки не могут быть сброшены — порог Q_1 определяет безопасную от сброса зону).

Исследование рассматриваемой системы будем проводить с помощью вложенной по моментам поступления цепи Маркова, образованной числом $\nu(\tau_n - 0)$ заявок в системе в моменты времени $(\tau_n - 0)$, где τ_n — момент поступления n -й заявки. Множество состояний построенной вложенной цепи Маркова имеет вид $\mathcal{X} = \{0, 1, \dots\}$.

Для построения матрицы переходных вероятностей вложенной цепи Маркова определим вспомогательные вероятности:

- A_k — вероятность того, что с момента прихода предыдущей заявки и до прихода следующей систему покинет ровно k ($k \geq 0$) заявок.

- A_0 — вероятность того, что за время между последовательными приходами заявок в систему, заявка на приборе не закончила обслуживание.

Тогда элементы $p_{i,j}$ (вероятности того, что в момент прихода заявки в системе $0j$ заявок ($j = \overline{0, i+1}$), если в момент прихода предыдущей заявки было i ($i \geq 0$) заявок) матрицы переходных вероятностей вложенной цепи Маркова определяются по приведенным далее формулам.

Если к приходу следующей заявки на приборе не закончилось обслуживание (в момент прихода текущей заявки в системе было i и стало $i+1$ к моменту прихода следующей), то вероятность того, что систему не покинула ни одна заявка, определяется по формуле (2.1).

$$p_{i,i+1} = A_0 = \int_0^{\infty} e^{-\mu e} dA(x) = \alpha(\mu), \quad (2.1)$$

где $\alpha(s)$ — преобразование Лапласа-Стилтьеса (ПЛС) функции $A(x)$.

Вероятность того, что в момент прихода новой заявки в системе будет 0 заявок, если в момент прихода предыдущей заявки также было 0 заявок определяется по формуле (2.2).

$$p_{0,0} = \int_0^{\infty} (1 - e^{-\mu e}) dA(x) = 1 - \alpha(\mu). \quad (2.2)$$

Пусть $0 \leq i \leq Q_1 + 1$, где i — число заявок в системе, пороговое значение в накопителе не превышено, механизм обновления не включен. Тогда можно

записать следующие уравнения для переходных вероятностей (2.3) и (2.4):

$$p_{i,i+1} = A_0 = \alpha(\mu), \quad (2.3)$$

$$p_{i,j} = A_{i+1-j} = \int_0^{\infty} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x). \quad (2.4)$$

Для перехода в нулевое состояние верна формула (2.5)

$$p_{i,0} = 1 - \sum_{j=1}^{i-1} p_{i,j}, \quad (2.5)$$

Пусть $i > Q_1 + 1$ и $j \geq Q_1 + 1$ — порог Q_1 был преодолен к приходу предыдущей заявки и остался превышенным к моменту поступления следующей. Тогда будем иметь следующие выражения (2.6) и (2.7) для расчета вероятностей:

$$p_{i,i+1} = A_0 = \alpha(\mu), \quad (2.6)$$

$$p_{i,j} = A_{i+1-j} = \int_0^{\infty} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} p^{i+1-j} dA(x), \quad (2.7)$$

где $i+1-j$ — это число заявок, которое должно покинуть систему только за счет обслуживания.

Рассмотрим случай, когда $i > Q_1 + 1, j = Q_1$. Определим вероятность $p_{i,Q_1} = P\{\text{либо } i+1-Q_1 \text{ обслужились и ушли}\} + P\{k \text{ заявок из } i+1-Q_1 \text{ обслужились и последняя } k\text{-ая заявка сбросила заявки из накопителя (из расположенных после порога)}\}$. Тогда справедливо выражение (2.8)

$$p_{i,Q_1} = \int_0^{\infty} \frac{(\mu x)^{i+1-Q_1} e^{-\mu x}}{(i+1-Q_1)!} p^{i+1-Q_1} dA(x) + \sum_{k=1}^{i+1-Q_1} \int_0^{\infty} \frac{(\mu x)^k e^{-\mu x}}{k!} p^{k-1} q dA(x). \quad (2.8)$$

Рассмотрим случай, когда $i > Q_1 + 1$, а $0 < j < Q_1$ (в момент прихода предыдущей заявки пороговое значение Q_1 в накопителе было преодолено и работал механизм вероятностного сброса, но к приходу следующей заявки механизм обновления отключился).

Вспомогательная вероятность $A_{i,j}(x)$ того, что цепь Маркова перейдет за время x из состояния i в состояние j определяется формулой (2.9):

$$p_{i,j} = \int_0^{\infty} \int_0^x A_{i+1,Q_1}(y) dy A_{Q_1,j}^*(x-y) dA(x), \quad (2.9)$$

где $A_{i+1,Q_1}(y)$ — переход из состояния i в j за время y , $A_{Q_1,j}^*(x-y)$ — переход из состояния i в j за оставшееся время $x-y$.

$$A_{i,Q-1}(y) = \frac{(\mu y)^{i+1-Q_i}}{(i+1-Q_i)!} e^{-\mu y} p^{i+1-Q_i} + \sum_{k=1}^{i+1-Q_i} \frac{(\mu y)^k}{k!} e^{-\mu y} p^{k-1} q,$$

$$A_{Q_1,j}^*(x-y) = \frac{(\mu(x-y))^{Q_1-j}}{(Q_1-j)!} e^{-\mu(x-y)}.$$

2.1.2. Стационарное по вложенной цепи Маркова распределение вероятностей числа заявок в системе

Для стационарных по вложенной цепи Маркова π_i ($i \geq 0$) вероятностей того, что в системе ровно i ($i \geq 0$) заявок справедлива система уравнений (2.10):

$$\pi_0 = \sum_{i=0}^{\infty} \pi_i p_{i,0}, \quad \pi_i = \sum_{j=i-1}^{\infty} \pi_j p_{j,i}, \quad i \geq 1, \quad (2.10)$$

где вероятности $p_{i,j}$ ($j = \overline{0, i+1}$), i ($i \geq 0$) определяются формулами (2.1), (2.2), (2.3), (2.4), (2.5), (2.6), (2.7), (2.8), (2.9).

Утверждение 2.1. При $i \geq Q_1 + 1$ стационарные вероятности π_i представимы в следующем виде:

$$\pi_i = \pi_{Q_1+1} \cdot g^{i-Q_1-1}, \quad g = \alpha(\mu(1 - pg)), \quad g \in (0, 1). \quad (2.11)$$

Доказательство: Докажем, что как и в работах [87; 88; 145; 160; 170], при $i \geq Q_1 + 1$ для стационарных вероятностей π_i верны формулы (2.11) и (2.12).

Согласно (2.10)

$$\pi_{Q_1+i+1} = \sum_{j=Q_1+i}^{\infty} \pi_j \cdot p_{j, Q_1+i+1}, \quad i \geq 0$$

Подставим вместо переходной вероятности p_{j, Q_1+i+1} ее выражение из формулы (2.7) и, предполагая, что формула (2.11) верна, упростим данное выражение: :

$$\begin{aligned}
\pi_{Q_1+i+1} &= \sum_{j=Q_1+i}^{\infty} \pi_j \int_0^{\infty} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} p^{i+1-j} dA(x) = \\
&= \sum_{j=Q_1+i}^{\infty} \pi_{Q_1+1} g^{j-Q_1-1} \int_0^{\infty} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} p^{i+1-j} dA(x) = \\
&= \pi_{Q_1+1} \sum_{j=0}^{\infty} g^{j-1+i} \int_0^{\infty} \frac{(\mu x)^j}{(j)!} e^{-\mu x} p^j dA(x) = \\
&= \pi_{Q_1+1} g^{i-1} \int_0^{\infty} \left(\sum_{j=0}^{\infty} \frac{(pg\mu x)^j}{(j)!} \right) e^{-\mu x} dA(x) = \pi_{Q_1+1} g^{i-1} \int_0^{\infty} e^{\mu pgx} e^{-\mu x} dA(x) = \\
&= \pi_{Q_1+1} g^{i-1} \int_0^{\infty} e^{-\mu(1-pg)x} dA(x) = \pi_{Q_1+1} g^{i-1} \alpha(\mu(1-pg)).
\end{aligned}$$

Таким образом:

$$\pi_{Q_1+i+1} = \pi_{Q_1+1} g^{i-1} \alpha(\mu(1-pg)),$$

и, исходя из (2.11), получаем, что параметр g является решением уравнения

$$g = \alpha(\mu(1-pg)), \quad g \in (0, 1). \quad (2.12)$$

Доказательство завершено.

Для стационарных вероятностей по вложенной цепи Маркова верно:

$$1 = \sum_{i=0}^{\infty} \pi_i = \sum_{i=0}^{Q_1} \pi_i + \sum_{i=Q_1+1}^{\infty} \pi_i. \quad (2.13)$$

Используя формулу (2.11) получаем, что (2.13) представима в виде (2.14):

$$1 = \sum_{i=0}^{Q_1} \pi_i + \sum_{i=Q_1+1}^{\infty} \pi_{Q_1+1} g^{i-Q_1-1} = \sum_{i=0}^{Q_1} \pi_i + \pi_{Q_1+1} \frac{1}{1-g}. \quad (2.14)$$

Таким образом, вероятность того, что механизм обновления не включен, определяется по формуле:

$$\sum_{i=0}^{Q_1} \pi_i = 1 - \frac{\pi_{Q_1+1}}{1-g}. \quad (2.15)$$

2.1.3. Вероятность обслуживания и вероятность потери поступающей в систему заявки

Обозначим через $p^{(serv)}$ вероятность того, что поступившая в систему заявка будет обслужена, а через $p^{(loss)}$ — вероятность того, что поступившая в систему заявка будет сброшена из накопителя (не будет обслужена).

Утверждение 2.2. Вероятности $p^{(serv)}$ и $p^{(loss)}$ определяются по формулам:

$$p^{(serv)} = 1 - \pi_{Q_1+1} \cdot \frac{q}{(1-g)(1-gp)}, \quad p^{(loss)} = \pi_{Q_1+1} \frac{q}{(1-g)(1-gp)}.$$

Доказательство: Введем вспомогательную вероятность $p_i^{(serv)}$ — условная вероятность того, что заявка поступившая в систему и заставшая в ней i других заявок перейдет на обслуживание.

Если $0 \leq i \leq Q_1$ (порог в накопителе не достигнут), $p_i^{(serv)} = 1$ (заявка не будет сброшена).

Если $i \geq Q_1 + 1$ (порог в накопителе (с учетом поступившей заявки) преодолен), то заявки, за счет которых в накопителе было преодолено пороговое значение Q_1 , должны обслуживаться и с вероятностью p просто уйти. Тогда

$$p_i^{(serv)} = p^{i-Q_1}.$$

Используя формулы (2.11) и (2.15), получим

$$p^{(serv)} = \sum_{i=0}^{Q_1} \pi_i + \sum_{i=Q_1+1}^{\infty} \pi_i p^{i-Q_1} = 1 - \pi_{Q_1+1} \cdot \frac{q}{(1-g)(1-pg)},$$

где q — вероятность сброса всех заявок из накопителя.

Используя следующее условие

$$p^{(loss)} + p^{(serv)} = 1, \Rightarrow p^{(loss)} = 1 - p^{(serv)},$$

получим, что

$$p^{(loss)} = \pi_{Q_1+1} \frac{q}{(1-g)(1-pg)}.$$

Доказательство завершено.

2.1.4. Временные характеристики системы

Предположим, что поступающие в систему заявки обслуживаются согласно порядку поступления — дисциплина обслуживания FCFS (First Come First Served) или FIFO (First In First Out).

Обозначим через $W^{(serv)}(x)$ стационарную функцию распределения времени ожидания начала обслуживания поступившей в систему заявки, а через $W^{(loss)}(x)$ стационарную функцию распределения времени пребывания в накопителе сброшенной заявки.

Обозначим через $w^{(serv)}(x)$ и $w^{(loss)}(x)$ — плотности распределения, а через $\omega^{(serv)}(x)$ и $\omega^{(loss)}(x)$ преобразования Лапласа-Стилтьеса функций $W^{(serv)}(x)$ и $W^{(loss)}(x)$.

Введем следующие условные вероятности:

- $W_i^{(serv)}(x)$ — условная вероятность того, что за время меньше x заявка, поступившая в систему и заставшая в ней i других заявок, перейдет на прибор.
- $W_i^{(loss)}(x)$ — условная вероятность того, что за время меньше x заявка, поступившая в систему и заставшая в ней i других заявок, будет сброшена,

и обозначим через $w_i^{(serv)}(x)$ и $w_i^{(loss)}(x)$ соответствующие условные плотности.

Тогда

$$W^{(serv)}(x) = \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} W_i^{(serv)}(x) \pi_i, \quad w^{(serv)}(x) = \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} w_i^{(serv)}(x) \pi_i,$$

$$W^{(loss)}(x) = \frac{1}{p^{(loss)}} \sum_{i=0}^{\infty} W_i^{(loss)}(x) \pi_i, \quad w^{(loss)}(x) = \frac{1}{p^{(loss)}} \sum_{i=0}^{\infty} w_i^{(loss)}(x) \pi_i.$$

Временные характеристики для обслуженной заявки

Рассмотрим сначала обслуженную заявку (то есть ту заявку, которая, поступив в систему, перешла на обслуживание).

Утверждение 2.3. Преобразование Лапласа-Стилтьеса функции распределения времени ожидания начала обслуживания имеет вид (2.16)

$$\omega^{(serv)}(s) = \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \left(\frac{\mu}{\mu + s} \right)^i \pi_i + p \left(\frac{\mu}{\mu + s} \right)^{Q_1+1} \pi_{Q_1+1} \frac{\mu + s}{\mu + s - p\mu g} \right). \quad (2.16)$$

Доказательство: Если $i = 0$ (система пустая), то $W_0^{(serv)}(x) = 1$.

Используя введенные плотности $w^{(serv)}(x)$ и условные плотности $w_i^{(serv)}(x)$, можно записать, что

$$w^{(serv)}(x) = \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} w_i^{(serv)}(x) \pi_i + \sum_{i=Q_1+1}^{\infty} w_i^{(serv)}(x) \pi_i \right),$$

где

$$w_0^{(serv)}(x) = 0, \quad w_i^{(serv)}(x) = \frac{\mu^i x^{i-1}}{(i-1)!} e^{-\mu x}, \quad i = \overline{1, Q_1},$$

$$w_i^{(serv)}(x) = p^i \frac{\mu^i x^{i-1}}{(i-1)!} e^{-\mu x}, \quad i \geq Q_1 + 1.$$

И тогда

$$w^{(serv)}(x) = \frac{1}{p^{(serv)}} \left(\sum_{i=1}^{Q_1} \frac{\mu^i x^{i-1}}{(i-1)!} e^{-\mu x} \pi_i + \sum_{i=Q_1+1}^{\infty} p^{i-Q_1} \frac{\mu^i x^{i-1}}{(i-1)!} e^{-\mu x} \pi_i \right).$$

Переходя к преобразованиям Лапласа-Стилтьеса функций $W^{(serv)}(x)$ и $W_i^{(serv)}(x)$ ($i \geq 0$), получим:

$$\omega_i^{(serv)}(s) = \left(\frac{\mu}{\mu + s} \right)^i, \quad i = \overline{0, Q_1}, \quad \omega_i^{(serv)}(s) = p^{i-Q_1} \left(\frac{\mu}{\mu + s} \right)^i, \quad i > Q_1,$$

Суммируя, окончательно получим:

$$\begin{aligned} \omega^{(serv)}(s) &= \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} \omega_i^{(serv)}(s) \pi_i = \\ &= \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \left(\frac{\mu}{\mu + s} \right)^i \pi_i + p \left(\frac{\mu}{\mu + s} \right)^{Q_1+1} \pi_{Q_1+1} \frac{\mu + s}{\mu + s - p\mu g} \right). \end{aligned}$$

Доказательство завершено.

Временные характеристики для сброшенной заявки

Перейдем теперь к сброшенной (заявке, принятой в систему и в дальнейшем сброшенной из накопителя) или «убитой» заявке (в терминах [87; 88; 160; 168; 170; 171]).

Утверждение 2.4. Преобразование Лапласа-Стилтьеса функции распределения времени пребывания в накопителе сброшенной заявки имеет

вид (2.17)

$$\omega^{(loss)}(s) = \frac{1}{p^{(loss)}} \cdot \frac{q\pi_{Q_1+1}}{1-g} \cdot \frac{\mu}{\mu+s-\mu pg}. \quad (2.17)$$

Доказательство: Стоит отметить, что если поступающая в систему заявка застает в ней $0 \leq i \leq Q_1$ других заявок, то она не может быть сброшена, так как будет находиться в безопасной зоне.

$$w^{(loss)}(x) = \frac{1}{p^{(loss)}} \sum_{i=Q_1+1}^{\infty} w_i^{(loss)}(x) \pi_i = \frac{1}{p^{(loss)}} \sum_{i=1}^{\infty} w_{Q_1+i}^{(loss)}(x) \pi_{Q_1+i},$$

где

$$w_i^{(loss)}(x) = \sum_{j=1}^i qp^{j-1} \frac{\mu^j x^{j-1}}{(j-1)!} e^{-\mu x}, \quad i \geq Q_1 + 1.$$

В терминах ПЛС:

$$\omega_{Q_1+i}^{(loss)}(s) = \sum_{j=1}^i qp^{j-1} \left(\frac{\mu}{\mu+s} \right)^j, \quad i \geq Q_1 + 1.$$

И тогда

$$\omega^{(loss)}(s) = \frac{1}{p^{(loss)}} \sum_{i=0}^{\infty} \omega_i^{(loss)}(s) \pi_i = \frac{1}{p^{(loss)}} \cdot \frac{q\pi_{Q_1+1}}{1-g} \cdot \frac{\mu}{\mu+s-\mu pg}.$$

Доказательство завершено.

Обозначим через $w^{(loss)}$ — среднее время пребывания в накопителе для сброшенной заявки.

$$\begin{aligned} w^{(loss)} &= - \left(\omega^{(loss)}(s) \right)'_{s=0} = - \left(\frac{1}{p^{(loss)}} \cdot \frac{q\pi_{Q_1+1}}{1-g} \cdot \frac{\mu}{\mu+s-\mu pg} \right)'_{s=0} = \\ &= \frac{1}{p^{(loss)}} \cdot \frac{q\pi_{Q_1+1}}{1-g} \cdot \frac{1}{\mu(1-pg)}. \end{aligned}$$

2.1.5. Временные характеристики для случая инверсионного обслуживания заявок

Воспользуемся введенными ранее обозначениями.

Если система в момент поступления рассматриваемой заявки пуста ($i = 0$), то заявка сразу переходит на обслуживание.

Если же в момент поступления заявки прибор занят, но имеется хотя бы одно свободное место в безопасной зоне ($0 < i \leq Q_1$), то, так как обслуживание в инверсионном порядке, приходится учитывать заявки, поступившие в систему после рассматриваемой. Первый вариант, если с учетом вновь пришедших заявок порог Q_1 не преодолен ($0 < i + j \leq Q_1$) (заявки не могут быть сброшены):

$$w_{i,j}^{(serv)}(x) = \frac{\mu^{j+1} x^j}{j!} e^{-\mu x} \bar{A}(x) + \int_0^x \sum_{k=0}^j \frac{(\mu y)^k}{k!} e^{-\mu y} w_{i,j-k+1}^{(serv)}(x-y).$$

Второй вариант, когда за счет вновь поступивших в систему заявок превышено в накопителе пороговое значение ($i + j > Q_1$) (рассматриваемая заявка не может быть сброшена, так как находится в безопасной зоне,

но время ожидания начала обслуживания может быть сокращено за счет сброса пришедших после заявок):

$$\begin{aligned}
w_{i,j}^{(serv)}(x) = & \sum_{k=1}^{i+j-Q_1} p^{k-1} q \frac{\mu^{Q_1-i+k} x^{Q_1-i+k-1}}{(Q_1-i+k-1)!} e^{-\mu x} \bar{A} x + \\
& + \int_0^x \sum_{k=0}^{i+j-Q_1} p^k \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) w_{i,j-k+1}^{(serv)}(x-y) + \\
& + \int_0^x \sum_{k=1}^{i+j-Q_1} p^{k-1} q \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) w_{i,Q_1-i}^{(serv)}(x-y) + \\
& + \int_0^x \int_0^y \sum_{k=1}^{i+j-Q_1} p^{k-1} q \frac{(\mu z)^k}{k!} e^{-\mu z} dz \sum_{k=0}^{Q_1-i-1} \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) \cdot w_{i,Q_1-i-k+i}^{(serv)}(x).
\end{aligned}$$

Если же в момент поступления заявки в накопителе было i других заявок, причем $i \geq Q_1$, то превышает значение порогового механизма и включается вероятностный сброс заявок:

$$\begin{aligned}
w_{i,j}^{(serv)}(x) = & p^{j+1} \cdot \frac{\mu^{j+i} x^j}{j!} \cdot e^{-\mu x} \bar{A}(x) + \\
& + \int_0^x \sum_{k=0}^j p^k \cdot \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) \cdot w_{i,j-k+1}^{(serv)}(x-y).
\end{aligned}$$

Утверждение 2.5. Преобразование Лапласа-Стилтьеса функции распределения времени пребывания в накопителе сброшенной заявки имеет вид (2.18)

$$\omega^{(loss)}(s) = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{q\mu\bar{\alpha}(\mu+s-\mu p u(s))}{(1-g)(1-u(s))}. \quad (2.18)$$

Доказательство: Если в момент поступления заявки в накопителе $i \geq Q_1$ других заявок, то из-за включившегося механизма обновления поступающая в систему заявка может быть сброшена (либо до прихода следующей заявки, либо после прихода в систему других заявок):

$$w_{i,j}^{(loss)}(x) = \sum_{k=1}^{j+1} p^{k-1} q \cdot \frac{\mu^k x^{k-1}}{(k-1)!} \cdot e^{-\mu x} \bar{A}(x) + \\ + \int_0^x \sum_{k=0}^j p^k \cdot \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) w_{i,j-k+1}^{(loss)}(x-y).$$

Рассмотрим ПЛС для сброшенной заявки при $i \geq Q_1$:

$$\omega_{i,j}^{(loss)}(s) = \sum_{k=1}^{j+1} p^{k-1} q \cdot \frac{(-1)^{k-1} \mu^k}{(k-1)!} \cdot \bar{\alpha}^{(K-1)}(\mu+s) + \\ + \sum_{k=0}^j p^k \cdot \frac{(-\mu)^k}{k!} \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k+1}^{(loss)}(s).$$

Тогда верно, что:

$$w^{(loss)}(x) = \frac{1}{p^{(loss)}} \sum_{i=Q_1+1}^{\infty} w_{i,0}^{(loss)}(x) \pi_i, \quad \omega^{(loss)}(s) = \frac{1}{p^{(loss)}} \sum_{i=Q_1+1}^{\infty} \omega_{i,0}^{(loss)}(s) \pi_i.$$

Как и в [170; 177] рассмотрим производящую функцию:

$$\omega_i^{(loss)}(s, u) = \sum_{j=0}^{\infty} \omega_{i,j}^{(loss)}(s) \cdot u^j = \sum_{j=0}^{\infty} u^j \left(\sum_{k=1}^{j+1} p^{k-1} q \frac{(-1)^{k-1} \mu^k}{(k-1)!} \bar{\alpha}^{(k-1)}(\mu+s) + \right. \\ \left. + \sum_{k=0}^j p^k \frac{(-\mu)^k}{k!} \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k+1}^{(loss)}(s) \right) =$$

$$= \frac{q \cdot \mu}{1 - u} \bar{\alpha}(\mu + s - \mu p u) + \frac{1}{u} \alpha(\mu + s - \mu p u) \left(\omega_i^{(loss)}(u, s) - \omega_{i,0}^{(loss)}(s) \right),$$

и выразим $\omega_i^{(loss)}(u, s)$.

$$\omega_i^{(loss)}(s, u) = \frac{u q \mu \bar{\alpha}(\mu + s - \mu p u) - (1 - u) \alpha(\mu + s - \mu p u) \cdot \omega_{i,0}^{(loss)}(s)}{(1 - u)(u - \alpha(\mu + s - \mu p u))}. \quad (2.19)$$

Обозначим знаменатель выражения (2.19) через $f(u) = f(u, s)$, то есть $f(u) = f(u, s) = u - \alpha(s + \mu - p \mu u)$ и рассмотрим уравнение $f(u) = 0$.

Аналогично [168; 177], поскольку при всех $s \geq 0$ выполнены неравенства $f(0) = -\alpha(s + \mu) < 0$, $f(1) = 1 - \alpha(s + \mu(1 - p)) \geq 0$, и, кроме того, при всех $0 < u < 1$ и $s \geq 0$ справедливо неравенство $f''(u) = -(\mu p)^2 \alpha''(s + \mu - p \mu u) \leq 0$, то функция $f(u)$ является на интервале $(0, 1)$ при всех $s \geq 0$ выпуклой (вверх) функцией и уравнение $f(u) = 0$ имеет единственное решение $u(s)$.

Используя свойство непрерывности функции $\omega_i^{(loss)}(s, u)$ и учитывая равенство нулю знаменателя правой части формулы (2.19) в точке $u(s)$, делаем вывод, что в этой точке равен нулю и числитель правой части этой формулы

$$u q \mu \bar{\alpha}(\mu + s - \mu p u) - (1 - u) \cdot u \cdot \omega_{i,0}^{(loss)}(s) = 0,$$

что приводит к равенству

$$\omega_{i,0}^{(loss)}(s) = \frac{q \mu \cdot \bar{\alpha}(\mu + s - \mu p u)}{1 - u}.$$

Таким образом, стационарное распределение времени пребывания в системе сброшенной заявки имеет ПЛС:

$$\omega^{(loss)}(s) = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{q\mu\bar{\alpha}(\mu + s - \mu p u(s))}{(1-g)(1-u(s))}.$$

Доказательство завершено.

2.1.6. Среднее число обслуженных и сброшенных заявок

Пусть случайная величина $\xi^{(loss)}$ — число сброшенных из поступивших в систему заявок, а случайная величина $\xi^{(serv)}$ — число обслуженных из поступивших в систему заявок.

Пусть $\pi_i^{(loss)} = P\{\xi^{(loss)} = i\}$ — вероятность того, что будет сброшено ровно i заявок из поступивших, $i \geq 0$.

Вероятности $\pi_i^{(loss)}$ определяются по следующей формуле:

$$\begin{cases} \pi_0^{(loss)} = \sum_{i=0}^{Q_1+1} \pi_i + \sum_{i=Q_1+2}^{\infty} \pi_i p^{i-Q_1-1}, \\ \pi_k^{(loss)} = \sum_{i=Q_1+1+k}^{\infty} \pi_i q p^{i-Q_1-k}. \quad k \geq 1. \end{cases}$$

Среднее число сброшенных заявок $N^{(loss)}$ определяется по формуле (2.20)

$$N^{(loss)} = \sum_{i=0}^{\infty} i \pi_i^{(loss)} = \frac{q \cdot \pi_{Q_1+1}}{(1-pg)(1-g)^2}. \quad (2.20)$$

Пусть $\pi_i^{(serv)} = P\{\xi^{(serv)} = i\}$ — вероятность того, что будет обслужено ровно i заявок из поступивших, $i > 0$.

Вероятности $\pi_i^{(serv)}$ определяются по следующей формуле:

$$\begin{cases} \pi_i^{(serv)} = \pi_{i-1} & 1 \leq i \leq Q_1 \\ \pi_{Q_1+i}^{(serv)} = \pi_{Q_1+i-1} p^{i-1} + \sum_{k=Q_1+i}^{\infty} \pi_k q p^{k-1}, & i \geq 1 \end{cases}$$

Среднее число обслуженных заявок определяется по формуле (2.21)

$$N^{(serv)} = \sum_{i=1}^{\infty} i \cdot \pi_i^{(serv)} = \sum_{i=1}^{Q_1+1} i \pi_{Q_1-1} + \pi_{Q_1+1} \cdot \frac{p + (Q_1 + 1)(1 - pg)}{(1 - g)(1 - pg)}, \quad (2.21)$$

где $\sum_{i=1}^{Q_1+1} i \pi_{Q_1-1}$ — среднее число обслуженных заявок из безопасной зоны,

а $\pi_{Q_1+1} \cdot \frac{p + (Q_1 + 1)(1 - pg)}{(1 - g)(1 - pg)}$ — среднее число обслуженных заявок из тех,

что были в накопителе вне безопасной зоны.

2.2. Система массового обслуживания $G/M/1/\infty$ с одним порогом, обновлением и без безопасной зоны

2.2.1. Описание системы, матрица переходных вероятностей вложенной цепи Маркова

В данном разделе рассмотрим систему массового обслуживания $G/M/1/\infty$, аналогичную рассмотренной в предыдущем разделе, только без безопасной зоны.

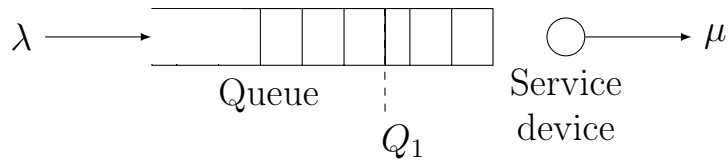


Рис. 2.2.. Однопороговая система без безопасной зоны в накопителе

Механизм обновления аналогичен предыдущей модели, за исключением того, что отсутствует безопасная зона.

Аналогично предыдущей модели исследование будем проводить с помощью вложенной по моментам поступления цепи Маркова $v(t_i - 0) = v_i$, где t_i — момент поступления в систему i -й заявки. Множество состояний цепи Маркова: $X = \{0, 1, 2, \dots\}$.

Введем вероятности A_i — вероятность того, что систему покидает ровно i заявок между последовательными моментами поступления в систему.

Обозначим через $p_{i,j}$ переходную для вложенной цепи Маркова вероятность того, что к моменту поступления новой заявки в системе будет ровно j заявок, если непосредственно перед приходом предыдущей в системе было i заявок, $i \geq 0, j = \overline{0, i+1}$.

Распишем вероятности $p_{i,j}$ ($i \geq 0, j = \overline{0, i+1}$) и определим их связи с вероятностями A_i как $p_{i,j} = A_{i-j+1}$, где $p_{i,i+1} = A_0$ — вероятность того, что систему не покинет ни одна заявка. Это возможно только в том случае, если на приборе не закончилось обслуживание.

$$p_{i,i+1} = A_0 = \int_0^{\infty} e^{-\mu x} dA(x) = \alpha(\mu).$$

Пусть с учетом пришедшей и заставшей в системе i заявок в накопителе не преодолен порог Q_1 ($1 < i \leq Q_1$), то есть, заявки из накопителя не сбрасываются. В этом случае заявки могут уходить из системы только за счет окончания обслуживания.

Тогда вероятность того, что в системе обслужится ровно $i + 1 - j$ заявок определяется по формуле (2.22):

$$p_{i,j} = A_{i+1-j} = \int_0^{\infty} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x), \quad i \geq 0, \quad j = \overline{0, i}, \quad (2.22)$$

где $\alpha^{(k)}(\mu)$ — производная порядка k преобразования Лапласа-Стилтьеса $\alpha(\mu)$.

Вероятность того, что систему покинут все заявки определяется по формуле (2.23):

$$p_{i,0} = 1 - \sum_{j=1}^{i+1} P_{ij} = A_{i+1}^* = 1 - \sum_{j=0}^{i+1} A_j, \quad i \geq 0. \quad (2.23)$$

Рассмотрим случай, когда $i \geq Q_1 + 1$ (то есть, в момент поступления новой заявки пороговое значение в накопителе достигнуто) и возможен сброс заявок из накопителя.

Если $0 < j \leq Q_1$ (система не станет пустой после ухода обслужившейся заявки и механизм обновления отключится), то переходная вероятность $p_{i,j} = A_{i+1-j}$ — это вероятность того, что ровно $i + 1 - j$ заявок обслужилось и покинуло систему (сброса из накопителя не было) и определяется по

формуле (2.24).

$$p_{i,j} = A_{i+1-j} = \int_0^{\infty} p^{i-Q_1} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x), \quad 0 < j \leq Q_1. \quad (2.24)$$

Если $Q_1 < j \leq i$ (порог остался преодоленным и механизм вероятностного сброса функционирует), то вероятность перехода из состояния i в состояние j определяется по формуле (2.25):

$$p_{i,j} = A_{i+1-j} = \int_0^{\infty} \frac{(p\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x). \quad (2.25)$$

2.2.2. Стационарное по вложенной цепи Маркова распределение числа заявок в системе

Пусть π_i — стационарная вероятность (по вложенной цепи Маркова того), что поступающая в систему заявка застанет в ней ровно i других заявок.

$$\pi_0 = \sum_{i=0}^{\infty} \pi_i p_{i,0}, \quad \pi_i = \sum_{k=i-1}^{\infty} \pi_k p_{k,i}, \quad i \geq 1. \quad (2.26)$$

Утверждение 2.6. При $i \geq Q_1 + 1$ для вероятностей из (2.26) справедлива формула (2.27):

$$\pi_i = \pi_{Q_1+1} \cdot g^{i-Q_1-1}, \quad i \geq Q_1 + 1, \quad (2.27)$$

где параметр g есть решение уравнения (2.29):

$$g = \alpha(\mu(1 - pg)), \text{quad} g \in (0, 1). \quad (2.28)$$

Доказательство. Докажем, что как и в работах [87; 160; 170] и как в предыдущей модели при $i \geq Q_1 + 1$ для вероятностей из (2.26) справедлива формула (2.27).

$$\begin{aligned} \pi_i &= \sum_{j=i-1}^{\infty} \pi_j A_{j-i+1} = \sum_{j=i-1}^{\infty} \pi_{Q_1+1} \cdot g^{j-Q_1-1} \int_0^{\infty} \frac{(p\mu x)^{j-i+1}}{(j-i+1)!} e^{-\mu x} dA(x) = \\ &= \int_0^{\infty} e^{-\mu x} \left(\sum_{j=i-1}^{\infty} \pi_{Q_1+1} \cdot g^{j-Q_1-1} \cdot \frac{(p\mu x)^{j-i+1}}{(j-i+1)!} \right) dA(x) = \\ &= \pi_{Q_1+1} \int_0^{\infty} e^{-\mu x} \left(\sum_{k=0}^{\infty} g^{k+i-1-Q_1-1} \cdot \frac{(p\mu x)^k}{k!} \right) dA(x) = \\ &= \pi_{Q_1+1} g^{i-Q_1-2} \int_0^{\infty} e^{-\mu x} \left(\sum_{k=0}^{\infty} \frac{(pg\mu x)^k}{k!} \right) dA(x) = \\ &= \pi_{Q_1+1} \left(\int_0^{\infty} e^{-\mu x} e^{pg\mu x} dA(x) \right) g^{i-Q_1-2} = \pi_{Q_1+1} g^{i-Q_1-2} \int_0^{\infty} e^{-\mu x(1-pg)} dA(x) = \\ &= \pi_{Q_1+1} \cdot g^{i-Q_1-2} \cdot \alpha(\mu(1 - pg)). \end{aligned}$$

Из нашего предположения (2.27) получаем

$$\pi_{Q_1+1} \cdot g^{i-Q_1-1} = \pi_i = \pi_{Q_1+1} \cdot g^{i-Q_1-2} \cdot \alpha(\mu(1 - pg)),$$

то есть параметр g есть решение уравнения

$$g = \alpha(\mu(1 - pg)), \text{quad} g \in (0, 1). \quad (2.29)$$

Доказательство завершено.

2.2.3. Вероятность обслуживания и вероятность потери заявки

Обозначим через $p^{(serv)}$ вероятность обслуживания поступившей в систему заявки, а через $p^{(loss)}$ — вероятность потери принятой в систему заявки.

Введем вспомогательную вероятность $p_i^{(serv)}(x)$ — вероятность того, что заявка, заставшая в системе i других заявок ($i \geq 0$), в дальнейшем за время x перейдет на прибор (не будет сброшена).

$$p_0^{(serv)} = 1.$$

Пусть $p_{i,j}^{(serv)}(x)$ — вероятность того, что за время меньше чем x , заявка, перед которой в накопителе (прибор занят) ровно i , а за которой в накопителе — ровно j заявок, перейдет на обслуживание. Тогда справедлива формула

$$p_{i,j}^{(serv)}(x) = \frac{(\mu x)^{i+1}}{(i+1)!} e^{-\mu x} \bar{A}(x) + \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) p_{i-k,j+1}^{(serv)}(x, y), \quad i + j + 1 \leq Q_1,$$

где $\frac{(\mu x)^{i+1}}{(i+1)!}e^{-\mu x}$ — обслуживание всех $i+1$ заявок, $\bar{A}(x)$ — новая заявка не пришла, $\frac{(\mu y)^k}{k!}e^{-\mu y}$ — обслуживание ровно k из $i+1$ заявок и $dA(y)$ — в момент y поступление новой заявки.

Для случая, когда $i+j+1 > Q_1+1$ — (суммарное число заявок в системе больше порога), получаем выражение:

$$p_{i,j}^{(serv)}(x) = \bar{A}(x) \cdot \frac{(\mu x)^{i+1}}{(i+1)!} e^{-\mu x} \cdot p^{\min(i+1, i+j+1-Q_1)} + \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} \cdot p_{k,j}^{(serv)} dA(y) p_{i-k, j+1}^{(serv)}(x-y),$$

где

$$p_{i+1, j}^{(serv)} = p^{\min(i+1, i+j+1-Q_1)} = \begin{cases} p^{i+1}, & i+1 < i+j+1-Q_1, \\ p^{i+j+1-Q_1}, & i+1 \geq i+j+1-Q_1. \end{cases}$$

Тогда

$$p^{(serv)} = \sum_{i=0}^{\infty} p_i^{(serv)} \pi_i, \quad p_{i+1}^{(serv)} = \int_0^{\infty} p_{i,0}^{(serv)}(x) dx, \quad i \geq 0,$$

и как результат

$$p^{(serv)} = \pi_0 + \sum_{i=0}^{\infty} \pi_{i+1} \int_0^{\infty} p_{i,0}^{(serv)}(x) dx.$$

Пусть $p^{(loss)}$ — вероятность того, что поступившая в систему заявка будет сброшена, $p_i^{(loss)}$ — вероятность того, что поступившая заявка будет

сброшена, если в момент поступления она застала в системе ровно i других заявок. Тогда

$$p_0^{(loss)} = 0.$$

$p_{i,j}^{(loss)}(x)$ — вероятность того, что за время x заявка, перед которой в накопителе ровно i заявок, за которой в накопителе ровно j заявок и прибор занят, будет сброшена.

$$p_{0,0}^{(loss)}(x) = \int_0^x e^{-\mu y} dA(y) p_{0,1}^{(loss)}(x-y),$$

$$p_{0,j}^{(loss)}(x) = \int_0^x e^{-\mu y} dA(y) p_{0,j+1}^{(loss)}(x-y), \quad 0 < j \leq Q_1 - 1,$$

$$p_{0,j}^{(loss)}(x) = q \frac{\mu x}{1!} e^{-\mu x} \bar{A}(x) + \int_0^\infty e^{-\mu y} dA(y) P_{0,j+1}^{(loss)}(x-y), \quad j \geq Q_1,$$

$$p_{i,j}^{(loss)}(x) = \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) p_{i-k,j+1}^{(loss)}(x-y), \quad i+j+1 \leq Q_1,$$

$$p_{i,j}^{(loss)}(x) = \sum_{k=1}^{i+1+j-Q_1} q p^{k-1} \frac{(-\mu x)^k}{k!} e^{\mu x} \bar{A}(x) + \int_0^\infty \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} \cdot p_{k,j}^{(serv)} dA(y) p_{i-k,j+1}^{(loss)}(x), \quad i+1+j > Q_1,$$

В результате

$$p^{(loss)} = \sum_{i=1}^{\infty} p_i^{(loss)} \cdot \pi_i = \sum_{i=1}^{\infty} \pi_i \int_0^{\infty} p_{i-1,0}^{(loss)}(x) dx.$$

2.2.4. Временные характеристики обслуженной заявки

Будем предполагать, что заявки в системе обслуживаются в порядке поступления, начиная с первой пришедшей.

Обозначим через $W^{(serv)}(x)$ стационарную функции распределения времени ожидания начала обслуживания для несброшенной заявки.

$$W^{(serv)}(x) = \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} W_i^{(serv)}(x) \pi_i,$$

где $p^{(serv)}$ — вероятность обслуживания поступившей в систему заявки, π_i ($i \geq 0$) — стационарная по вложенной цепи Маркова вероятность того, что в момент прихода заявки в системе ровно i других заявок, а $W_i^{(serv)}(x)$ ($i \geq 0$) вспомогательная вероятность того, что поступившая в систему заявка перейдет на обслуживание за время меньше x , если в момент поступления в системе было i других заявок (если система пуста, то заявка сразу переходит на обслуживание).

Обозначим через $W_{i,j}^{(serv)}(x)$ ($i \geq 0, j \geq 0$) вспомогательную вероятность того, что заявка, перед которой в накопителе находится ровно i заявок, а за которой в накопитель пришло еще ровно j заявок, перейдет на обслуживание за время меньше чем x (прибор занят). Тогда $W_i^{(serv)}(x) = W_{i-1,0}^{(serv)}(x)$.

Введем $w_{i,j}^{(serv)}(x)$ — плотность функции $W_{i,j}^{(serv)}(x)$

Рассмотрим случай, когда $i+j+1 \leq Q_1$ (порог в накопителе не преодолен).

$$w_{i,j}^{(serv)}(x) = \bar{A}(x) \cdot \frac{\mu^{i+1} x^i}{i!} e^{-\mu x} + \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) \cdot w_{i-k,j+1}^{(serv)}(x-y).$$

Перейдем к случаю, когда $i+j+1 > Q_1$ (порог в накопителе преодолен).

$$w_{i,j}^{(serv)}(x) = \bar{A}(x) \cdot \frac{\mu^{i+1} x^i}{i!} e^{-\mu x} p_{i+1,j}^{(serv)} + \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} p_{k,j}^{(serv)} dA(y) \cdot w_{i-k,j+1}^{(serv)}(x-y).$$

Пусть $p_{i+1,j}^{(serv)}$ — вероятность того, что заявки, заканчивающие обслуживание на приборе, не будут опустошать накопитель (то есть, просто покидают систему с вероятностью p), пока пороговое значение в накопителе превышено:

$$p_{i+1,j}^{(serv)} = \begin{cases} p^{i+1}, & i+j+1 - Q_1 \geq i+1 \quad (j \geq Q_1), \\ p^{i+j+1-Q_1}, & i+j+1 - Q_1 < i+1 \quad (j < Q_1). \end{cases}$$

Первый вариант соответствует случаю, когда число заявок, на которое превышено пороговое значение, больше числа заявок, находящихся в системе (в накопителе и на приборе) перед "выделенной". Второй вариант — число заявок, на которое превышено пороговое значение, меньше числа заявок в системе перед выделенной (в накопителе и на приборе).

Перейдем к преобразованию Лапласа-Стилтьеса функции $W_{i,j}^{(serv)}(x)$.

Если $i + j + 1 \leq Q_1$, то

$$w_{i,j}^{(serv)}(s) = \frac{(-1)^i \mu^{i+1} \bar{\alpha}^{(i)}(s + \mu)}{i!} + \sum_{k=0}^i \frac{(-1)^k \mu^k \alpha^k(s + \mu)}{k!} \cdot w_{i-k,j+1}^{(serv)}(s).$$

Здесь $\alpha(s)$ — ПЛС функции $A(x)$, $\bar{\alpha}(s)$ — преобразование Лапласа (ПЛ) функции $\bar{A}(x)$.

Если $i + j + 1 > Q_1$, то

$$w_{i,j}^{(serv)}(s) = \frac{(-1)^i \mu^{i+1} \bar{\alpha}^{(i)}(s + \mu)}{i!} p_{i+1,j}^{(serv)} + \sum_{k=0}^i \frac{(-1)^k \mu^k \alpha^k(s + \mu)}{k!} p_{k,j}^{(serv)} \cdot w_{i-k,j+1}^{(serv)}(s).$$

2.2.5. Временные характеристики для сброшенной из накопителя заявки

Будем считать, что заявки обслуживаются в порядке поступления. Обозначим через $W^{(loss)}(x)$ — вероятность того, что заявка будет сброшена за время, меньшее x , а через $W_i^{(loss)}(x)$ — условную вероятность того, что заявка будет сброшена из накопителя за время меньше x , если в момент ее поступления в системе ровно i ($i \geq 0$) других заявок, а $W_{i,j}^{(loss)}(x)$ — условная вероятность того, что заявка, перед которой в накопителе ровно i ($i \geq 0$) заявок и за которой ровно j ($j \geq 0$) других заявок, будет сброшена из накопителя за время меньше x (прибор занят). Тогда

$$W_i^{(loss)}(x) = W_{i-1,0}^{(loss)}(x), \quad i \geq 1,$$

$$W^{(loss)}(x) = \frac{1}{p^{(loss)}} \sum_{i=0}^{\infty} W_i^{(loss)}(x) \pi_i = \frac{1}{p^{(loss)}} \sum_{i=1}^{\infty} W_{i,0}^{(loss)}(x).$$

Обозначим через $w_{i,j}^{(loss)}(x)$ и $\omega_{i,j}^{(loss)}(s)$ ($i \geq 0, j \geq 0$) плотности и преобразования Лапласа-Стилтьеса функций $W_i^{(loss)}(x)$.

Рассмотрим следующие варианты.

Если $i + j + 1 \leq Q_1$, порог не преодолен, то верно выражение

$$w_{i,j}^{(loss)}(x) = \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) w_{i-k,j+1}^{(loss)}(x-y).$$

Если порог преодолен — $i + j + 1 \geq Q_1$ — (порог преодолен), то возможны два варианта. Первый, когда $0 \leq i \leq Q_1 - 1$ — выделенная заявка находится в накопителе до порога либо на приборе. Тогда нужно учесть, на сколько $i + j + 1 - Q_1$ заявок в накопителе преодолен порог, и:

$$w_{i,j}^{(loss)}(x) = \sum_{k=1}^{\min(i, i+1+j-Q_1)} \frac{(\mu)^k x^{k-1}}{(k-1)!} e^{-\mu x} \cdot p^{k-1} \cdot q \cdot \bar{A}(x) + \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} \cdot p^{\min(k, i+j+1-Q_1)} dA(y) w_{i-k,j+1}^{(loss)}(x-y),$$

где $p^{\min(k, i+j+1-Q_1)}$ — вероятность того, что выделенная заявка будет сброшена. Если $i > i + j + 1 - Q_1$, то обслуживание $i + j + 1 - Q_1$ заявки из i приведет к сбросу выделенной. Если $i \leq i + j + 1 - Q_1$, то обслуживание i заявок может привести к сбросу выделенной заявки.

Второй вариант, если $i \geq Q_1$ — рассматриваемая заявка находится после порогового значения. Сброс заявок из накопителя происходит за счет преодоления порога на $i + j + 1 - Q_1$ заявку, но обслужиться могут не более i заявок. Тогда

$$w_{i,j}^{(loss)}(x) = \sum_{k=1}^{\min(i, i+1+j-Q_1)} \frac{(\mu)^k x^{k-1}}{(k-1)!} e^{-\mu x} \cdot p^{k-1} \cdot q \cdot \bar{A}(x) + \int_0^x \sum_{k=0}^i \frac{(\mu y)^k}{k!} e^{-\mu y} \cdot p^{\min(k, i+j+1-Q_1)} dA(y) w_{i-k, j+1}^{(loss)}(x-y)$$

.

Запишем выражения для плотностей в терминах преобразований Лапласа-Стилтьеса.

Для случая, когда $i + 1 + j \leq Q_1$:

$$\omega_{i,j}^{(loss)}(s) = \sum_{k=0}^i \frac{(-1)^k \mu^k}{k!} \alpha^{(k)}(\mu + s) \cdot \omega_{i-k, j+1}^{(loss)}(s).$$

Для случая, когда $i + 1 + j > Q_1$:

$$\omega_{i,j}^{(loss)}(s) = \sum_{k=1}^{\min(i, i+1+j-Q_1)} \frac{(-1)^{k-1} \mu^k}{(k-1)!} \bar{\alpha}^{(k-1)}(\mu + s) \cdot p^{k-1} \cdot q + \sum_{k=0}^i \frac{(-1)^k \mu^k}{k!} p^{\min(k, i+j+1-Q_1)} \alpha^{(k)}(\mu + s) \cdot \omega_{i-k, j+1}^{(loss)}(s)$$

Тогда

$$\omega^{(loss)}(s) = \frac{1}{p^{(loss)}} \sum_{i=1}^{\infty} \pi_i \cdot \omega_{i-1, 0}^{(loss)}(s).$$

Глава 3. Однопороговая система массового обслуживания с обобщенным обновлением и безопасной зоной в накопителе

3.1. Описание системы, матрица переходных вероятностей вложенной цепи Маркова

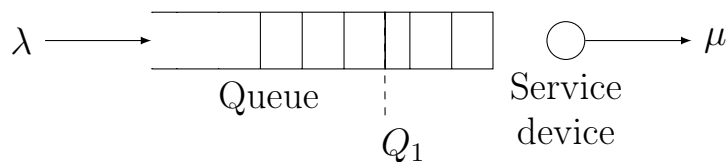


Рис. 3.1.. Система массового обслуживания $G/M/1/\infty$.

Механизм обновления:

- если число i заявок в накопителе меньше порогового значения Q_1 , то сброс заявок не происходит (механизм обновления не включен);
- если пороговое значение преодолено, то есть $i > Q_1$, то включается механизм обобщенного обновления [168; 170; 171], когда в момент окончания своего обслуживания на приборе заявка может либо сбросить с вероятностью $q(k)$ ($q \geq 1$) ровно k ($k \geq 1$) заявок из тех, что находятся в накопителе вне безопасной зоны (начиная с $Q_1 + 1$ заявки в накопителе), либо с вероятностью $p = q(0)$ просто покинуть систему ($\sum_{k=0}^{\infty} q(k) = 1$).

Исследование рассматриваемой системы (как и в предыдущей главе) будем проводить с помощью вложенной по моментам поступления цепи Маркова, образованной числом $\nu(\tau_n - 0)$ заявок в системе в моменты времени $(\tau_n - 0)$, где τ_n — момент поступления n -й заявки. Множество состояний вложенной цепи Маркова имеет вид $\mathcal{X} = \{0, 1, \dots\}$.

Вероятность $p_{i,j}$ — это переходная вероятность вложенной по моментам поступления цепи Маркова того, что в момент поступления заявка застанет в системе j заявок, если предыдущая поступившая заявка застала в системе i заявок ($i \geq 0, j \geq 0$).

Введем вспомогательные вероятности A_{i+j+1} — вероятность того, что между последовательными поступлениями заявок в систему из системы уйдут (обслужатся и/или будут сброшены) $i + j + 1$ заявок ($i \geq 0, j \geq 0$).

Тогда верно, что

$$p_{i,j} = A_{i+1-j}, \quad p_{i,i+1} = A_0 = \int_0^{\infty} e^{-\mu x} dA(x) = \alpha(\mu), \quad i \geq 0, j = \overline{0, i+1}$$

Если в момент поступления заявки механизм обновления не был включен ($0 \leq i \leq Q_1$), то уход из системы возможно только за счет обслуживания и, следовательно,

$$p_{i,j} = A_{i+1-j}^{(1)} = \int_0^{\infty} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x). \quad (3.1)$$

Вероятность того, в системе останется 0 заявок к приходу следующей заявки определяется по формуле:

$$p_{i,0} = 1 - \sum_{j=1}^{i+1} p_{i,j} = 1 - \sum_{j=1}^{i+1} A_{i+1-j}^{(1)}. \quad (3.2)$$

Для вывода дальнейших выражений воспользуемся вспомогательными вероятностями $\pi_m(k)$ — вероятностями того, что в результате обслуживания ровно m заявок систему покинет ровно k заявок:

$$\pi_0(0) = 1, \quad \pi_0(k) = 0, \quad k > 0, \quad \pi_1(k) = q(k-1), \quad k \geq 1, \quad (3.3)$$

$$\pi_m(k) = \sum_{l=m-1}^{k-1} \pi_{m-l}(l)\pi_1(k-l) = \sum_{l=1}^{k-m+1} \pi_{m-1}(l)\pi_1(k-l), \quad m \geq 2, k \geq m \quad (3.4)$$

Если к моменту прихода заявки механизм обновления был включен ($i > Q_1$) и к моменту прихода новой заявки останется включенным ($j > Q_1 + 1$), то

$$p_{i,j} = A_{i+1-j}^{(2)} = \int_0^{\infty} \sum_{k=1}^{i+1-j} \pi_k(i+1-j-k) \frac{(\mu x)^k}{k!} e^{-\mu x} dA(x),$$

$$j \neq i+1, \quad Q_1 + 1 < j < i+1, \quad (3.5)$$

Случай, когда к моменту прихода новой заявки механизм обновления отключен $j = Q_1$, но изначально был включен.

$$p_{i,Q_1} = A_{i+1-Q_1}^{(3)} = \int_0^\infty \sum_{k=1}^{i-Q_1} \pi_k(i-Q_1-k) \cdot \frac{(\mu x)^{k+1}}{(k+1)!} e^{-\mu x} dA(x) + \int_0^\infty \sum_{k=1}^{i-Q_1} \tilde{\pi}_k(i-Q_1-k) \cdot \frac{(\mu x)^k}{k!} dA(x), \quad j \neq i+1, \quad (3.6)$$

где вспомогательная вероятность $\tilde{\pi}_k(i)$ — это вероятность того, что k обслужившихся заявок сбросят из накопителя не менее i заявок, которая определяется по формулам (3.7) и (3.8):

$$\tilde{\pi}_1(i) = \sum_{k=i}^\infty q(k), \quad \tilde{\pi}_1(0) = \sum_{k=0}^\infty q(k) = 1; \quad (3.7)$$

$$\tilde{\pi}_k(i) = \sum_{l=0}^{i-1} \pi_1(l) \tilde{\pi}_{k-1}(l-1) = \sum_{l=0}^{i-1} q(l) \tilde{\pi}_{k-1}(l-1). \quad (3.8)$$

Далее, если изначально механизм обновления был включен, но к приходу следующей заявки в системе останется ровно $0 < j < Q_1$ заявок:

$$p_{i,j} = A_{i+1-j}^{(4)} = \int_0^\infty \int_0^x A_{i-1-Q_1}^{(3)}(y) dy A_{Q_1-j}^{(1)}(x-y) dA(x), \quad (3.9)$$

где

$$A_{i-1-Q_1}^{(3)}(y) = \sum_{k=1}^{i-Q_1} \pi_k(i-Q_1-k) \cdot \frac{(\mu y)^{k-1}}{(k-1)!} e^{-\mu y} + \sum_{k=1}^{i-Q_1-1} \tilde{\pi}_k(i-Q_1-k) \cdot \frac{(\mu y)^k}{k!} e^{-\mu y},$$

$$A_{Q_1-j}^{(1)}(x-y) = \frac{(\mu(x-y))^{Q_1-j}}{(Q_1-j)!} e^{\mu(x-y)}.$$

Вероятности перехода в нулевое состояние:

$$p_{i,0} = 1 - \sum_{j=1}^{i+1} p_{i,j} = 1 - \left(\sum_{j=1}^{Q_1-1} A_{i+1-j}^{(4)} + A_{i+1-Q_1}^{(3)} \sum_{j=Q_1+1}^{i+1} A_{i-Q_1}^{(2)} \right).$$

3.2. Стационарное распределение по вложенной цепи Маркова числа заявок в системе

Пусть π_i — стационарная по вложенной цепи Маркова вероятность того, что поступившая в систему заявка застанет в ней ровно i ($i \geq 0$) других заявок. Тогда система уравнений для вероятностей π_i имеет вид:

$$\pi_0 = \sum_{i=0}^{\infty} p_{i,0} \cdot \pi_i, \quad \pi_k = \sum_{i=0}^{\infty} p_{i,k} \cdot \pi_i, \quad i \geq 1. \quad (3.10)$$

В сделанных обозначениях (3.1), (3.2), (3.5), (3.6) и (3.9) система уравнений (3.10) представима в виде (3.11):

$$\left\{ \begin{array}{l} \pi_0 = \sum_{i=0}^{\infty} A_i^{(5)} \cdot \pi_i, \\ \pi_k = \sum_{i=k-1}^{Q_1-1} \pi_i A_{i+1-k}^{(1)} + \sum_{i=Q_1}^{\infty} \pi_i A_{i+1-Q_1}^{(3)}, \quad 0 < k < Q_1, \\ \pi_{Q_1} = \pi_{Q_1-1} A_0 + \sum_{i=Q_1}^{\infty} \pi_i A_{i+1-Q_1}^{(3)}, \\ \pi_k = \sum_{i=k-1}^{\infty} \pi_i \cdot A_{i+1-k}^{(2)}, \quad k > Q_1. \end{array} \right. \quad (3.11)$$

Утверждение 3.1. При $i \geq Q_1 + 1$ стационарная вероятность π_i представима в виде (3.12):

$$\pi_i = \pi_{Q_1+1} \cdot g^{i-Q_1-1}, \quad i \geq Q_1 + 1 \quad \text{или} \quad \pi_{Q_1+1+i} = \pi_{Q_1+1} g^i, \quad i \geq 0, \quad (3.12)$$

где параметр g является решением уравнения

$$g = \alpha(\mu(1 - g\hat{\pi}(g))), \quad 0 < g < 1. \quad (3.13)$$

Доказательство: Докажем верность сделанного предположения и выведем формулу для расчета константы g .

Подставим в формулу

$$\pi_k = \sum_{i=k-1}^{\infty} \pi_i \cdot A_{i+1-k}^{(2)}, \quad k > Q_1.$$

выражение для переходных вероятностей из (3.5):

$$\begin{aligned} \pi_{Q_1+1+j} &= \sum_{i=j}^{\infty} \pi_{Q_1+1} \cdot A_{Q_1+i+1}^{(2)} = \sum_{i=j}^{\infty} \pi_{Q_1+1} \cdot g^{i-1} \cdot A_{i-j}^{(2)} = \\ &= \pi_{Q_1+1} \sum_{k=0}^{\infty} g^{k+j-1} A_k^{(2)} = \pi_{Q_1+1} g^{j-1} \sum_{k=0}^{\infty} g^k \int_0^{\infty} \sum_{i=1}^k \pi_i (k-i) \cdot \frac{(\mu x)^i}{i!} e^{-\mu x} dA(x) = \\ &= \pi_{Q_1+1} g^{j-1} \left(\int_0^{\infty} e^{-\mu x} dA(x) + \int_0^{\infty} \sum_{k=1}^{\infty} g^k \pi_k (k-1) \cdot \frac{(\mu x)^i}{i!} e^{-\mu x} dA(x) \right) = \\ &= \pi_{Q_1+1} g^{j-1} \left(\int_0^{\infty} \sum_{k=0}^{\infty} g^k \left(\sum_{i=0}^k \pi_i (k-1) \frac{(\mu x)^i}{i!} \right) e^{-\mu x} dA(x) \right) = \end{aligned}$$

$$\begin{aligned}
&= \pi_{Q_1+1} g^{j-1} \left(\int_0^\infty e^{-\mu x} dA(x) + \int_0^\infty \sum_{k=1}^\infty g^k \sum_{i=1}^k \pi_i(k-1) \frac{(\mu x)^i}{i!} e^{-\mu x} dA(x) \right) = \\
&= \pi_{Q_1+1} g^{j-1} \int_0^\infty e^{-\mu x} \sum_{i=0}^\infty \frac{(g \hat{\pi}(g) \cdot \mu x)^i}{i!} dA(x) = \\
&= \pi_{Q_1+1} g^{j-1} \int_0^\infty e^{-\mu x} \cdot e^{g \hat{\pi}(g) \cdot \mu x} dA(x) = \\
&= \pi_{Q_1+1} g^{j-1} \int_0^\infty e^{-\mu(1-g \hat{\pi}(g))x} dA(x) = \pi_{Q_1+1} g^{j-1} \alpha(\mu(1-g \hat{\pi}(g))).
\end{aligned}$$

Здесь $\hat{\pi}(g) = \sum_{k=0}^\infty g^k \cdot q(k) = \hat{\pi}(g) = \sum_{k=0}^\infty g^k \cdot \pi_1(1+k)$ — производящая функция для вероятностей обновления, а $\sum_{n=0}^\infty \pi_m(m+n)g^n = (\hat{\pi}(g))^m$ — это производящая функция вспомогательных вероятностей $\pi_m(m+n)$, $n \geq 0$.

В итоге получили, что

$$\pi_{Q_1+1+j} = \pi_{Q_1+1} g^{j-1} \alpha(\mu(1-g \hat{\pi}(g))),$$

и таким образом, параметр g определяется из уравнения (3.13).

Доказательство завершено.

3.3. Вероятностные характеристики системы

Обозначим через p^{save} вероятность того, что поступающая в систему заявка попадет в безопасную зону или сразу на обслуживание. Тогда справедлива формула (3.14):

$$p^{\text{save}} = \sum_{i=0}^{Q_1} \pi_i = 1 - \sum_{i=Q_1+1}^{\infty} \pi_i = 1 - \sum_{i=Q_1+1}^{\infty} g^{i-Q_1-1} \cdot \pi_{Q_1+1} = 1 - \frac{\pi_{Q_1+1}}{1-g}. \quad (3.14)$$

Пусть, как и в предыдущих моделях, $p^{(\text{serv})}$ — это вероятность того, что поступившая в систему заявка будет обслужена, а $p^{(\text{loss})}$ — вероятность того, что поступившая в систему заявка будет сброшена.

$$p^{(\text{serv})} + p^{(\text{loss})} = 1$$

Утверждение 3.2. Вероятность обслуживания поступившей в систему заявки $p^{(\text{serv})}$ вычисляется по формуле (3.15):

$$p^{(\text{serv})} = \sum_{i=0}^{\infty} p_i^{(\text{serv})} \pi_i = 1 - \pi_{Q_1+1} \cdot \frac{1 - \hat{\pi}(g)}{(1-g)(1-g\hat{\pi}(g))}. \quad (3.15)$$

Вероятность потери поступившей в систему заявки $p^{(\text{loss})}$ вычисляется по формуле (3.16):

$$p^{(\text{loss})} = \sum_{i=0}^{\infty} p_i^{(\text{loss})} \pi_i = \pi_{Q_1+1} \cdot \frac{1 - \hat{\pi}(g)}{(1-g)(1-g\hat{\pi}(g))}. \quad (3.16)$$

Доказательство: Введем вспомогательные вероятности того, что заявка, заставшая в системе в момент поступления i , ($i \geq 0$) других заявок, либо

будет обслужена — $p_i^{(serv)}$, либо будет сброшена — $p_i^{(loss)}$. Тогда:

$$p^{(serv)} = \sum_{i=0}^{\infty} p_i^{(serv)} \pi_i, \quad p^{(loss)} = \sum_{i=0}^{\infty} p_i^{(loss)} \pi_i, \quad (3.17)$$

где π_i — стационарные по вложенной цепи Маркова вероятности, определяемые из системы (3.11) и уравнения (3.12).

Снова воспользуемся вспомогательными вероятностями $\pi_m(k)$, определенными формулами (3.3) и (3.4), а также вероятностями $\tilde{\pi}_k(i)$ ($k \geq 1, i \geq 0$) того, что k обслужившихся заявок сбросят из накопителя не менее i заявок, определяемыми по формулам (3.7) и (3.8).

Обозначим через $Q(g) = \sum_{l=0}^{\infty} \tilde{\pi}_1(l) \cdot g^l$ производящую функцию для вероятностей $\tilde{\pi}_1(l)$, а через $Q_k(g) = \sum_{l=0}^{\infty} \tilde{\pi}_k(l) \cdot g^l$ производящую функцию вероятностей $\tilde{\pi}_k(l)$:

$$Q(g) = \sum_{l=0}^{\infty} \tilde{\pi}_1(l) \cdot g^l = \frac{1 - g\hat{\pi}(g)}{1 - g}, \quad (3.18)$$

где $\hat{\pi}(g)$ — производящая функция для вероятностей $\pi_1(k)$, $k \geq 1$.

$$Q_k(g) = \sum_{l=0}^{\infty} \tilde{\pi}_k(l) \cdot g^l = (\tilde{\pi}(g))^{k-1} \cdot Q(g). \quad (3.19)$$

Будем предполагать, что поступившие в систему заявки обслуживаются и сбрасываются согласно порядку прихода в систему.

Если в момент поступления рассматриваемой заявки пороговое значение Q_1 не было преодолено (т. е. механизм обновления не был включен), то

заявка либо переходит на обслуживание ($i = 0$), либо в безопасную зону ($0 < i \leq Q_1$):

$$p_i^{(serv)} = 1, \quad p_i^{(loss)} = 0.$$

Если $i \geq Q_1 + 1$, то в момент поступления заявки безопасная зона заполнена полностью и пришедшая заявка в дальнейшем может быть сброшена, так как включен механизм обновления. И тогда

$$p_i^{(serv)} = \sum_{k=1}^{i-Q_1} \pi_k(i - Q_1 - k), \quad (3.20)$$

$$p_i^{(loss)} = \tilde{\pi}_1(i - Q_1) + \sum_{l=1}^{i-Q_1-1} \tilde{\pi}_1(l) \cdot \sum_{k=1}^{i-Q_1-l} \pi_k(i - Q_1 - l - k),$$

В результате получаем, что вероятность обслуживания поступившей в систему заявки $p^{(serv)}$ (3.17) и вероятность потери поступившей в систему заявки $p^{(loss)}$ (3.17) (с учетом формул (3.3), (3.4), (3.7), (3.8), (3.18), (3.19) и (3.20)) представимы в виде (3.15) и (3.16):

$$p^{(serv)} = \sum_{i=0}^{\infty} p_i^{(serv)} \pi_i = 1 - \pi_{Q_1+1} \cdot \frac{1 - \hat{\pi}(g)}{(1-g)(1-g\hat{\pi}(g))}.$$

$$p^{(loss)} = \sum_{i=0}^{\infty} p_i^{(loss)} \pi_i = \pi_{Q_1+1} \cdot \frac{1 - \hat{\pi}(g)}{(1-g)(1-g\hat{\pi}(g))}.$$

Доказательство завершено.

3.4. Временные характеристики системы

Пусть как и ранее, $W^{(serv)}(x)$ — стационарная функция распределения времени ожидания начала обслуживания несброшенной заявки, а $W^{(loss)}(x)$ — стационарная функция распределения времени пребывания в накопителе сброшенной заявки, а $W_i^{(serv)}(x)$ и $W_i^{(loss)}(x)$ — вспомогательные условные вероятности того, что за время x с момента прихода в систему, заявка, заставшая в системе ровно i ($i \geq 0$) других заявок, будет обслужена или, соответственно, сброшена.

$W_{i,j}^{(serv)}(x)$ и $W_{i,j}^{(loss)}(x)$ — вспомогательные вероятности того, что за время x заявка, перед которой i заявок и за которой в накопите j других заявок, будет либо обслужена, либо сброшена.

Выше определенные функции распределения имеют вид (3.21) и (3.22) соответственно.

$$W^{(serv)}(x) = \frac{1}{p^{(serv)}} \cdot \sum_{i=0}^{\infty} W_i^{(serv)}(x) \cdot \pi_i = \frac{1}{p^{(serv)}} \cdot \sum_{i=0}^{\infty} W_{i,0}^{(serv)}(x) \cdot \pi_i, \quad (3.21)$$

$$W^{(loss)}(x) = \frac{1}{p^{(loss)}} \cdot \sum_{i=0}^{\infty} W_i^{(loss)}(x) \cdot \pi_i = \frac{1}{p^{(loss)}} \cdot \sum_{i=0}^{\infty} W_{i,0}^{(loss)}(x) \cdot \pi_i. \quad (3.22)$$

Далее будем использовать плотности $w_{i,j}^{(serv)}(x)$, $w_{i,j}^{(loss)}(x)$ и преобразования Лапласа-Стилтьеса $\omega_{i,j}^{(serv)}(s)$ и $\omega_{i,j}^{(loss)}(s)$ введенных вспомогательных функций $W_{i,j}^{(serv)}(x)$ и $W_{i,j}^{(loss)}(x)$ ($i \geq 0, j \geq 0$).

Рассмотрим несколько вариантов обслуживания и вероятностного сброса (обобщенного обновления) поступивших в систему заявок.

3.4.1. Обслуживание и сброс заявок в порядке поступления

Утверждение 3.3. Преобразование Лапласа-Стилтьеса функции распределения времени ожидания начала обслуживания несброшенной заявки имеет вид (3.23):

$$\omega^{(serv)}(s) = \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \left(\frac{\mu}{\mu + s} \right)^i \pi_i + \pi_{Q_1+1} \left(\frac{\mu}{\mu + s} \right)^{Q_1+1} \frac{\hat{\pi}(g)(\mu + s)}{\mu + s - \mu g \hat{\pi}(g)} \right), \quad (3.23)$$

а ПЛС функции распределения времени пребывания в накопителе сброшенной заявки имеет вид (3.24):

$$\omega^{(loss)}(s) = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1 - \hat{\pi}(g)}{1 - g} \cdot \frac{\mu}{\mu + s - \mu g \hat{\pi}(g)}. \quad (3.24)$$

Доказательство: Если в момент поступления рассматриваемой заявки система пуста ($i = 0$), то:

$$W_0^{(serv)}(x) = 1, x \geq 0, \quad W_0^{(loss)}(x) = 0, x \geq 0.$$

Если в момент поступления рассматриваемой заявки система не пуста, но есть хотя бы одно свободное место в безопасной зоне ($0 < i \leq Q_1$), то механизм обновления не включен и время ожидания обслуживания для несброшенной заявки подчиняется распределению Эрланга с параметрами

μ и i ($0 < i \leq Q_1$):

$$W_i^{(serv)}(x) = H_i(x) = \int_0^x \frac{\mu^i x^{i-1}}{(i-1)!} e^{-\mu x} dx,$$

а для сброшенной заявки верно, что $W_i^{(loss)}(x) = 0$ ($0 < i \leq Q_1$).

Если же в момент поступления рассматриваемой заявки безопасная зона заполнена полностью ($i \geq Q_1 + 1$), то механизм обновления уже включен.

И тогда

$$W_{Q_1+i}^{(serv)}(x) = \sum_{j=0}^i H_{Q_1+j}(x) \cdot \pi_j(i-j),$$

$$W_{Q_1+i}^{(loss)}(x) = \tilde{\pi}_1(i) \cdot H_1(x) \sum_{k=2}^i H_k(x) \sum_{j=0}^{i-k} \pi_{k-1}(j) \tilde{\pi}_1(i+1-k-j),$$

Переходя к преобразованиям Лапласа-Стилтьеса

$$\omega^{(serv)}(s) = \int_0^{\infty} e^{-sx} dW^{(serv)}(x), \quad \omega^{(loss)}(s) = \int_0^{\infty} e^{-sx} dW^{(loss)}(x),$$

$$\omega_i^{(serv)}(s) = \int_0^{\infty} e^{-sx} dW_i^{(serv)}(x), \quad \omega_i^{(loss)}(s) = \int_0^{\infty} e^{-sx} dW_i^{(loss)}(x),$$

$$\omega_i^{(serv)}(s) = \sum_{j=1}^{i-Q_1} \left(\frac{\mu}{\mu+s} \right)^{Q_1+1} \cdot \pi_j(i-Q_j-j), \quad i \geq Q_1 + 1, \quad (3.25)$$

и используя выражения (3.12), (3.18), (3.19) и (3.25), получим, что

$$\begin{aligned}
\omega^{(serv)}(s) &= \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} \omega_i^{(serv)}(s) \cdot \pi_i = \\
&= \frac{1}{p^{(serv)}} \left(\pi_0 + \sum_{i=1}^{Q_1} \omega_i^{(serv)}(s) \cdot \pi_i + \sum_{Q_1+1}^{\infty} \omega_i^{(serv)}(s) \cdot \pi_i \right) = \\
&= \frac{1}{p^{(serv)}} \left(\pi_0 + \sum_{i=1}^{Q_1} \left(\frac{\mu}{\mu+s} \right)^i \cdot \pi_i + \right. \\
&\quad \left. + \sum_{i=Q_1+1}^{\infty} \sum_{j=1}^{i-Q_1} \left(\frac{\mu}{\mu+s} \right)^{Q_1+j} \cdot \pi_j(i-Q_1-j) \cdot \pi_i \right) = \\
&= \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \left(\frac{\mu}{\mu+s} \right)^i \pi_i + \pi_{Q_1+1} \left(\frac{\mu}{\mu+s} \right)^{Q_1+1} \frac{\widehat{\pi}(g)(\mu+s)}{\mu+s-\mu g \widehat{\pi}(g)} \right).
\end{aligned}$$

Для сброшенной заявки при $i \geq Q_1 + 1$ в терминах ПЛС верно выражение (3.26)

$$\begin{aligned}
\omega_i^{(loss)}(s) &= \widetilde{\pi}_1(i-Q_1) \cdot \frac{\mu}{\mu+s} + \\
&\sum_{k=2}^{i-Q_1} \left(\frac{\mu}{\mu+s} \right)^k \sum_{j=0}^{i-Q_1-k} \pi_{k-1}(j) \cdot \widetilde{\pi}_1(i-Q_1+1-k-j), \quad i > Q_1 + 1, \quad (3.26)
\end{aligned}$$

И тогда, учитывая выражения (3.3), (3.4), (3.7), (3.8), (3.12), (3.18), (3.19) и (3.26), запишем итоговое выражение для $\omega^{(loss)}(s)$

$$\begin{aligned}
\omega^{(loss)}(s) &= \frac{1}{p^{(loss)}} \cdot \sum_{i=0}^{\infty} \omega_i^{(loss)}(s) \cdot \pi_i = \frac{1}{p^{(loss)}} \sum_{i=Q_1+1}^{\infty} \omega_i^{(loss)}(s) \cdot \pi_{Q_1+1} \cdot g^{i-Q_1-1} = \\
&= \frac{\pi_{Q_1+1}}{p^{(loss)}} \sum_{i=Q_1+1}^{\infty} g^{i-Q_1-1} \cdot \omega_i^{(loss)}(s) = \\
&= \frac{\pi_{Q_1+1}}{p^{(loss)}} \left(\widetilde{\pi}_1(1) \cdot \frac{\mu}{\mu+s} + \sum_{i=Q_1+2}^{\infty} g^{i-Q_1-1} \left(\widetilde{\pi}_1(i-Q_1) \cdot \frac{\mu}{\mu+s} + \right. \right.
\end{aligned}$$

$$\begin{aligned}
& + \sum_{k=2}^{i-Q_1} \left(\frac{\mu}{\mu+s} \right)^k \sum_{j=0}^{i-Q_1-k} \pi_{k-1}(j) \cdot \tilde{\pi}_1(i-Q_1+1-k-j) \Bigg) = \\
& = \frac{\pi_{Q_1+1}}{p^{(loss)}} \left(\frac{\mu}{\mu+s} \cdot \frac{1-\hat{\pi}(g)}{1-g} + \frac{\mu}{\mu+s} \cdot \frac{g(1-\hat{\pi}(g))}{1-g} \cdot \frac{\mu\hat{\pi}(g)}{\mu+s-\mu g\hat{\pi}(g)} \right).
\end{aligned}$$

Итоговое выражение для $\omega^{(loss)}(s)$ представимо в виде (3.24):

$$\omega^{(loss)}(s) = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1-\hat{\pi}(g)}{1-g} \cdot \frac{\mu}{\mu+s-\mu g\hat{\pi}(g)}$$

Доказательство завершено.

Среднее время ожидания начала обслуживания и среднее время до сброса

Перейдем к средним числовым характеристикам.

Среднее время ожидания начала обслуживания определяется по формуле

$$W^{(serv)} = - \left(\omega^{(serv)}(s) \right)'_{s=0}.$$

Производная преобразования Лапласа-Стилтьеса (3.23):

$$\begin{aligned}
\left(\omega^{(serv)}(s) \right)'_{s=0} & = \frac{1}{p^{(serv)}} \left(\sum_{i=1}^{Q_1} \pi_i \cdot \frac{i\mu^i}{(\mu+s)^{i+1}} + \right. \\
& + \pi_{Q_1+1} \cdot \hat{\pi}(g) \cdot \left(\frac{-(Q_1+1) \cdot \mu^{Q_1+1}}{(\mu+s)^{Q_1+2}} \cdot \frac{\mu+s}{\mu+s-\mu g\hat{\pi}(g)} + \right. \\
& \left. \left. + \left(\frac{\mu}{\mu+s} \right)^{Q_1+1} \cdot \frac{(\mu+s) - \mu g\hat{\pi}(g) - (\mu+s)}{(\mu+s-\mu g\hat{\pi}(g))^2} \right) \right)_{s=0} =
\end{aligned}$$

$$\begin{aligned}
&= \frac{1}{p^{(serv)}} \left(- \sum_{i=1}^{Q_1} \frac{i\pi_i}{\mu} + \pi_{Q_1+1} \cdot \hat{\pi}(g) \left(- \frac{Q_1+1}{\mu - \mu g \hat{\pi}(g)} - \frac{\mu g \hat{\pi}(g)}{(\mu - \mu g \hat{\pi}(g))^2} \right) \right) = \\
&\quad - \frac{1}{p^{(serv)}} \left(\frac{1}{\mu} \sum_{i=1}^{Q_1} i\pi_i + \pi_{Q_1+1} \cdot \hat{\pi}(g) \cdot \frac{\mu(Q_1+1 - Q_1 g \hat{\pi}(g))}{\mu^2(1 - g \hat{\pi}(g))^2} \right).
\end{aligned}$$

Следовательно, среднее время ожидания начала обслуживания имеет вид (3.27):

$$w^{(serv)} = \frac{1}{p^{(serv)}} \left(\frac{1}{\mu} \sum_{i=1}^{Q_1} i\pi_i + \frac{\pi_{Q_1+1} \cdot \hat{\pi}(g) \cdot (Q_1+1 - Q_1 g \hat{\pi}(g))}{\mu(1 - g \hat{\pi}(g))^2} \right) \quad (3.27)$$

Для среднего времени пребывания в накопителе сброшенной заявки справедлива формула:

$$w^{(loss)} = - \left(\omega^{(loss)}(s) \right)'_{s=0}.$$

Производная преобразования Лапласа-Стилтьеса (3.24):

$$\begin{aligned}
\left(\omega^{(loss)}(s) \right)'_{s=0} &= \left(\frac{1}{p^{(loss)}} \cdot \pi_{Q_1+1} \cdot \frac{1 - \hat{\pi}(g)}{1 - g} \cdot \frac{\mu}{\mu + s - \mu g \hat{\pi}(g)} \right)'_{s=0} = \\
&= \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1 - \hat{\pi}(g)}{1 - g} \left(\frac{\mu}{\mu + s - \mu g \hat{\pi}(g)} \right)'_{s=0} = - \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1 - \hat{\pi}(g)}{\mu(1 - g)(1 - g \hat{\pi}(g))^2}.
\end{aligned}$$

Следовательно, среднее время пребывания в накопителе до сброса определяется по формуле (3.28)

$$w^{(loss)} = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1 - \hat{\pi}(g)}{\mu(1 - g)(1 - g \hat{\pi}(g))^2}. \quad (3.28)$$

Временные характеристики для произвольной заявки

Время пребывания в накопителе произвольной заявки в терминах ПЛС определяется по формуле:

$$\begin{aligned}
 \omega(s) &= p^{(serv)} \cdot \omega^{(serv)}(s) + p^{(loss)} \cdot \omega^{(loss)}(s) = \\
 &= p^{(serv)} \cdot \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \pi_i \left(\frac{\mu}{\mu+s} \right)^i + \pi_{Q_1+1} \left(\frac{\mu}{\mu+s} \right)^{Q_1+1} \cdot \frac{\hat{\pi}(g)(\mu+s)}{\mu+s-\mu g \hat{\pi}(g)} \right) + \\
 &\quad + p^{(loss)} \frac{1}{p^{(loss)}} \cdot \pi_{Q_1+1} \cdot \frac{1-\hat{\pi}(g)}{1-g} \cdot \frac{\mu}{\mu+s-\mu g \hat{\pi}(g)} = \\
 &= \sum_{i=0}^{Q_1} \pi_i \left(\frac{\mu}{\mu+s} \right)^i + \pi_{Q_1+1} \cdot \frac{\mu+s}{\mu+s-\mu g \hat{\pi}(g)} \\
 &\quad \left(\left(\frac{\mu}{\mu+s} \right)^{Q_1+1} \hat{\pi}(g) + \frac{\mu}{\mu+s} \cdot \frac{1-\hat{\pi}(g)}{1-g} \right).
 \end{aligned}$$

Пусть w — среднее время пребывания в накопителе произвольной заявки.

$$\begin{aligned}
 w &= p^{(serv)} \cdot w^{(serv)} + p^{(loss)} \cdot w^{(loss)} = \\
 &= p^{(serv)} \cdot \frac{1}{p^{(serv)}} \left(\frac{1}{\mu} \sum_{i=1}^{Q_1} i \pi_i + \pi_{Q_1+1} \frac{\hat{\pi}(g)(Q_1+1-Q_1 g \hat{\pi}(g))}{\mu(1-g \hat{\pi}(g))^2} \right) + \\
 &\quad + p^{(loss)} \cdot \frac{1}{p^{(loss)}} \cdot \pi_{Q_1+1} \cdot \frac{1-\hat{\pi}(g)}{1-g} \cdot \frac{1}{\mu(1-g \hat{\pi}(g))^2} = \\
 &= \frac{1}{\mu} \sum_{i=1}^{Q_1} i \pi_i + \frac{1}{\mu} \pi_{Q_1+1} \cdot \frac{1+Q_1 \hat{\pi}(g)(1-g)}{(1-g)(1-g \hat{\pi}(g))}.
 \end{aligned}$$

3.4.2. Обслуживание и обновление в инверсионном порядке

Рассмотрим вариант, когда поступающие в систему заявки обслуживаются и сбрасываются в порядке, обратном порядку поступления.

Воспользуемся принятыми ранее обозначениями.

Утверждение 3.4. Преобразование Лапласа-Стилтьеса функции распределения ожидания начала обслуживания имеет вид (3.29):

$$\begin{aligned} \omega^{(serv)}(s) &= \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} \omega_{i,0}^{(serv)}(s) \pi_i = \\ &= \frac{1}{p^{(serv)}} \left(\pi_0 + \sum_{i=1}^{Q_1} \omega_{i,0}^{(serv)}(s) \pi_i + \pi_{Q_1+1} \cdot \sum_{i=Q_1+1}^{\infty} \omega_{i,0}^{(serv)}(s) g^{i-Q_1-1} \right). \end{aligned} \quad (3.29)$$

Доказательство: Если в момент поступления заявка застаёт систему пустой, то она сразу переходит на обслуживание.

Если в момент поступления заявки пороговое значение в накопителе не был достигнуто (в системе было i других заявок ($0 < i < Q_1 + 1$)), то механизм обновления(сброса) не был включен и поступившая в систему заявка будет находиться в безопасной зоне. следовательно, не может быть сброшена:

$$w_{i,0}^{(serv)}(x) = \mu e^{-\mu x} \bar{A}(x) + \int_0^x e^{-\mu y} dA(y) \cdot w_{i,1}^{(serv)}(x - y),$$

но, так как обслуживание в инверсионном порядке, следует учитывать заявки, пришедшие в систему после рассматриваемой. Возможны два варианта.

Первый вариант — даже с учетом j ($j \geq 1$) пришедших после рассматриваемой заявки пороговое значение в накопителе не преодолено и механизм обновления не включен ($i + j \leq Q_1$):

$$w_{i,j}^{(serv)}(x) = \frac{\mu^{j+1} x^j}{j!} e^{-\mu x} \cdot \bar{A}(x) + \int_0^x \sum_{k=0}^j \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) \cdot w_{i,j-k+1}^{(serv)}(x-y).$$

И второй вариант, когда с учетом j ($j \geq 1$) пришедших после рассматриваемой заявки пороговое значение в накопителе преодолено и механизм обновления включен ($i + j > Q_1$):

$$\begin{aligned} w_{i,j}^{(serv)}(x) &= \sum_{k=1}^{j+i-Q_1} \frac{\mu^{k+Q_1-i} \cdot x^{k+Q_1-i}}{(k+Q_1-i)!} e^{-\mu x} \tilde{\pi}_k(i+j-Q_1-k) \bar{A}(x) + \\ &+ \int_0^x \sum_{k=0}^{i+j-Q_1} \frac{(\mu y)^k}{k!} e^{-\mu y} \sum_{l=0}^{i+j-Q_1-k} \pi_k(l) dA(y) \cdot w_{i,j-k-l+1}^{(serv)}(x-y) + \\ &+ \int_0^x \sum_{k=1}^{i+j-Q_1} \tilde{\pi}_k(i+j-Q_1-k) \sum_{l=0}^{Q_1-i} \frac{(\mu y)^{k+l}}{(k+l)!} e^{-\mu y} dA(y) \cdot w_{i,Q_1-i-l+1}^{(serv)}(x-y). \end{aligned}$$

Наконец, случай когда в момент своего поступления в систему заявка застаёт безопасную зону полностью занятой и механизм обновления (сброса) включен ($i > Q_1$) — рассматриваемая заявка может перейти за время x на обслуживание:

$$w_{i,0}^{(serv)}(x) = \mu e^{-\mu x} p \cdot \bar{A}(x) + \int_0^x e^{-\mu y} dA(y) \cdot w_{i,1}^{(serv)}(x-y),$$

$$\begin{aligned}
w_{i,j}^{(serv)}(x) &= \sum_{k=1}^{j+1} \frac{\mu^k x^{k-1}}{(k-1)!} \pi_k(j-k+1) \bar{A}(x) + \\
&\quad + \int_0^x \sum_{k=0}^j \frac{(\mu y)^k}{k!} e^{-\mu y} \sum_{l=0}^j \pi_k(l) dA(y) \cdot w_{i,j-k-l+1}^{(serv)}(x).
\end{aligned}$$

В терминах преобразований Лапласа-Стилтьеса выражения для плотностей, представленные выше, будут иметь следующий вид. Если $i+k \leq Q_1$, то:

$$\omega_{i,j}^{(serv)}(s) = \frac{(-1)^j \mu^{j+1}}{j!} \bar{A}^{(j)}(\mu+s) + \sum_{k=0}^j \frac{(-1)^k \mu^k}{k!} \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k+1}^{(serv)}(s)$$

Если $i+j > Q_1$, то:

$$\begin{aligned}
\omega_{i,j}^{(serv)}(s) &= \sum_{k=1}^{i+j-Q_1} \frac{(-1)^{k+Q_1-i} \cdot \mu^{k+Q_1-i+1}}{(k+Q_1-i)!} \tilde{\pi}_k(i+j-Q_1-k) \cdot \bar{\alpha}^{(k+Q_1-i)}(\mu+s) + \\
&\quad + \sum_{k=0}^{i+j-Q_1} \frac{(-1)^k \cdot \mu^k}{k!} \sum_{l=0}^{i+j-Q_1-k} \pi_k(l) \cdot \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k-l+1}^{(serv)}(s) + \\
&\quad + \sum_{k=1}^{i+i-Q_1} \tilde{\pi}_k(i+j-Q_1-k) \cdot \sum_{l=0}^{Q_1-i} \frac{(-1)^{k+l} \cdot \mu^{k+l}}{(k+l)!} \alpha^{(k+l)}(\mu+s) \cdot \omega_{i,Q_1-i-l+1}^{(serv)}(s).
\end{aligned}$$

Если $i > Q_1$, то

$$\begin{aligned}
\omega_{i,j}^{(serv)}(s) &= \sum_{k=1}^{j+1} \frac{(-1)^{k-1} \mu^k}{(k-1)!} \pi_k(j-k+1) \cdot \bar{\alpha}^{(k-1)}(\mu+s) + \\
&\quad + \sum_{k=0}^j \frac{(-1)^k \mu^k}{k!} \sum_{l=0}^{j-k} \pi_k(l) \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k-l+1}^{(serv)}(s).
\end{aligned}$$

Тогда ПЛС $\omega^{(serv)}(s)$ для $W^{(serv)}(s)$ имеет вид:

$$\begin{aligned}\omega^{(serv)}(s) &= \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} \omega_{i,0}^{(serv)}(s) \pi_i = \\ &= \frac{1}{p^{(serv)}} \left(\pi_0 + \sum_{i=1}^{Q_1} \omega_{i,0}^{(serv)}(s) \pi_i + \pi_{Q_1+1} \cdot \sum_{i=Q_1+1}^{\infty} \omega_{i,0}^{(serv)}(s) g^{i-Q_1-1} \right).\end{aligned}$$

Доказательство завершено.

Утверждение 3.5. Преобразование Лапласа-Стилтьеса функции распределения времени пребывания в накопителе сброшенной заявки имеет вид (3.30):

$$\omega^{(loss)}(s) = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1}{1-g} \cdot \frac{\mu(1-\widehat{\pi}(z(s)))}{\mu + s\mu \cdot z(s) \cdot \widehat{\pi}(z(s))}, \quad (3.30)$$

а среднее время пребывания сброшенной заявки в накопителе $w^{(loss)}$ определяется по формуле (3.31):

$$\begin{aligned}w^{(loss)} &= \frac{\pi_{Q_1+1}}{(1-g)p^{(loss)}} \left(\frac{\widehat{\pi}'(g) \cdot \alpha'(\mu - \mu g \widehat{\pi}(g)) \cdot (1-g)}{(1-g\widehat{\pi}(g))^2} + \right. \\ &\quad \left. + \frac{(1-\widehat{\pi}(g))(1-\mu\widehat{\pi}(g)\alpha'(\mu - \mu g \widehat{\pi}(g)))}{\mu(1-g\widehat{\pi}(g))^2} \right). \quad (3.31)\end{aligned}$$

Доказательство: Поступающая в систему заявка может быть сброшена только тогда, когда в момент своего поступления в систему она застает безопасную зону полностью занятой и механизм обновления (сброса) включен

($i > Q_1$):

$$w_{i,0}^{(loss)}(x) = \mu e^{-\mu x} \tilde{\pi}_1(1) \cdot \bar{A}(x) + \int_0^x e^{-\mu y} dA(y) \cdot w_{i,1}^{(loss)}(x-y),$$

$$w_{i,j}^{(loss)}(x) = \sum_{k=1}^{j+1} \frac{\mu^k x^{k-1}}{(k-1)!} \tilde{\pi}_k(j-k+2) \bar{A}(x) + \\ + \int_0^x \sum_{k=0}^j \frac{(\mu y)^k}{k!} e^{-\mu y} \sum_{l=0}^{j-k} \pi_k(l) dA(y) \cdot w_{i,j-k-l+1}^{(loss)}(x).$$

В терминах ПЛС:

$$\omega_{i,j}^{(loss)}(s) = \sum_{k=1}^{j+1} \frac{(-1)^{k-1} \mu^k}{(k-1)!} \tilde{\pi}_k(j-k+2) \cdot \bar{\alpha}^{(k-1)}(\mu+s) + \\ + \sum_{k=0}^j \sum_{l=0}^{j-k} \frac{(-1)^k \mu^k}{k!} \pi_k(l) \cdot \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k-l+1}^{(loss)}(s).$$

Тогда ПЛС $\omega^{(loss)}(s)$ для $W^{(loss)}(s)$ имеет вид:

$$\omega^{(loss)}(s) = \frac{1}{p^{(loss)}} \sum_{i=0}^{\infty} \omega_{i,0}^{(loss)}(s) \pi_i = \frac{\pi_{Q_1+1}}{p^{(loss)}} \sum_{i=Q_1+1}^{\infty} \omega_{i,0}^{(loss)}(s) \cdot g^{i-Q_1-1} = \\ = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \omega_0^{(loss)}(s) \cdot \sum_{i=Q_1+1}^{\infty} g^{i-Q_1-1} = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{\omega_0^{(loss)}(s)}{1-g}.$$

Найдем $\omega_0^{(loss)}(s)$. Введем производящую функцию:

$$\begin{aligned}
\psi(z, s) &= \sum_{j=0}^{\infty} z^j \omega_{i,j}^{(loss)}(s) = \\
&= \sum_{j=0}^{\infty} z^j \left(\sum_{k=1}^{j+1} \frac{(-1)^{(k-1)} \cdot \mu^k}{(k-1)!} \tilde{\pi}_k(j-k+2) \cdot \bar{\alpha}^{(k-1)}(\mu+s) + \right. \\
&\quad \left. + \sum_{k=0}^j \frac{(-\mu)^k}{k!} \sum_{l=0}^{j-k} \pi_k(l) \cdot \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k-l+1}^{(loss)}(s) \right) = \\
&= \sum_{j=0}^{\infty} z^j \sum_{k=1}^{j+1} \frac{(-1)^{(k-1)} \cdot \mu^k}{(k-1)!} \tilde{\pi}_k(j-k+2) \cdot \bar{\alpha}^{(k-1)}(\mu+s) + \\
&\quad + \sum_{j=0}^{\infty} z^j \sum_{k=0}^j \frac{(-\mu)^k}{k!} \sum_{l=0}^{j-k} \pi_k(l) \cdot \alpha^{(k)}(\mu+s) \cdot \omega_{i,j-k-l+1}^{(loss)}(s),
\end{aligned}$$

которая после ряда преобразований будет представима в виде:

$$\psi(z, s) = \frac{\mu(Q(z) - 1) \cdot (1 - \alpha(\mu + s - \mu z \hat{\pi}(z)))}{z - \alpha(\mu + s - \mu z \hat{\pi}(z))} - \frac{\omega_{i,0}^{(loss)}(s) \cdot \alpha(\mu + s - \mu z \hat{\pi}(z))}{z - \alpha(\mu + s - \mu z \hat{\pi}(z))}$$

Учитывая, что время пребывания в накопителе сброшенной заявки зависит только от того, сколько заявок пришло в систему после нее, то есть $\omega_{i,0}^{(loss)}(s) = \omega_0^{(loss)}(s)$, представленное выше выражение можно записать как:

$$\psi(z, s) = \frac{\mu(Q(z) - 1) \cdot (1 - \alpha(\mu + s - \mu z \hat{\pi}(z)))}{z - \alpha(\mu + s - \mu z \hat{\pi}(z))} - \frac{\omega_0^{(loss)}(s) \cdot \alpha(\mu + s - \mu z \hat{\pi}(z))}{z - \alpha(\mu + s - \mu z \hat{\pi}(z))}, \quad 0 < z < 1, s > 0.$$

При $z = \alpha(\mu + s - \mu z \widehat{\pi}(z))$ знаменатель выражения $\psi(z, s)$ обращается в 0. Следовательно при этом же значении z (в силу непрерывности $\psi(z, s)$) и числитель должен обращаться в ноль, и тогда ($z = z(s)$):

$$\omega_0^{(loss)}(s) = \frac{\mu \cdot (Q(z(s)) - 1)}{\alpha(\mu + s - \mu z(s) \cdot \widehat{\pi}(z(s)))} \cdot \frac{1 - \alpha(\mu + s - \mu z(s) \cdot \widehat{\pi}(z(s)))}{\mu + s - \mu z(s) \cdot \widehat{\pi}(z(s))}$$

Упростим вид $\omega_0^{(loss)}(s)$ используя то, что $Q(z) = \frac{1 - z \widehat{\pi}(z)}{1 - z}$, и в результате, преобразование Лапласа-Стилтьеса функции распределения времени пребывания в накопителе сброшенной заявки будет иметь вид:

$$\omega^{(loss)}(s) = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{\omega_0^{(loss)}(s)}{1 - g} = \frac{\pi_{Q_1+1}}{p^{(loss)}} \cdot \frac{1}{1 - g} \cdot \frac{\mu(1 - \widehat{\pi}(z(s)))}{\mu + s\mu \cdot z(s) \cdot \widehat{\pi}(z(s))}.$$

Отметим, что при $s = 0$ $z(0) = \alpha(\mu + s - \mu z(0) \cdot \widehat{\pi}(z(0)))$ и совпадает с выражением для параметра g ($0 < g < 1$: $g = \alpha(\mu + s - \mu g \widehat{\pi}(g))$), то есть $z(0) = g$.

Среднее время пребывания сброшенной заявки в накопителе $w^{(loss)}$ определяется по формуле:

$$\begin{aligned} w^{(loss)} &= - \left(\omega^{(loss)}(s) \right)'_{s=0} = \frac{-\pi_{Q_1+1} \cdot \mu}{(1 - g)p^{(loss)}} \left(\frac{1 - \widehat{\pi}(z(s))}{\mu + s - \mu z(s) \cdot \widehat{\pi}(z(s))} \right)'_{s=0} = \\ &= \frac{\pi_{Q_1+1}}{(1 - g)p^{(loss)}} \left(\frac{\widehat{\pi}'(g) \cdot \alpha'(\mu - \mu g \widehat{\pi}(g)) \cdot (1 - g)}{(1 - g \widehat{\pi}(g))^2} = \right. \\ &\quad \left. + \frac{(1 - \widehat{\pi}(g))(1 - \mu \widehat{\pi}(g) \alpha'(\mu - \mu g \widehat{\pi}(g)))}{\mu(1 - g \widehat{\pi}(g))^2} \right). \end{aligned}$$

Доказательство завершено.

Заключение

В заключении диссертационной работы представим основные полученные результаты:

1. Сделан обзор ряда алгоритмов активного управления очередями семейства RED (с описанием их основных характеристик) и представлены результаты, полученные для систем с полным и обобщенным обновлением.
2. Рассмотрен механизм порогового управления трафиком на примере системы СМО $G|M|1|\infty$ с вероятностным механизмом сброса (полного обновления), момент включения (выключения) которого зависит от преодоления текущей длиной очереди порогового значения Q_1 в накопителе. Данный порог также определяет область в накопителе, из которой поступившие в систему заявки не могут быть сброшены. Доказано, что вероятностное распределение числа заявок в системе (по вложенной цепи Маркова) имеет геометрический вид (если преодолен порог). Представлены формулы для расчета вероятности того, что поступившая в систему заявка либо будет обслужена, либо будет сброшена. Также в терминах ПЛС получены выражения для стационарного распределения времени пребывания в накопителе обслуженной и сброшенной заявок, а также средние временные

характеристики при обслуживании заявок либо в порядке поступления, либо в инверсионном порядке. Также выведено стационарное распределение числа обслуженных либо сброшенных заявок.

3. На примере системы СМО $G|M|1|\infty$ с полным обновлением рассмотрен механизм порогового управления трафиком, для которого пороговое значение Q_1 задает только момент включения механизма сброса. Доказано, что вероятностное распределение числа заявок в системе (по вложенной цепи Маркова) имеет геометрический вид (если преодолен порог Q_1). Представлены уравнения для стационарного распределения времени пребывания в накопителе обслуженной или сброшенной заявок.
4. На примере системы СМО $G|M|1|\infty$ с обобщенным обновлением рассмотрен механизм порогового управления трафиком, для которого пороговое значение Q_1 не только определяет момент включения механизма сброса, но и задает границы безопасной области в накопителе, из которой находящиеся в ней заявки не могут быть сброшены. Доказано, что вероятностное распределение числа заявок в системе (по вложенной цепи Маркова) имеет геометрический вид (если преодолен порог). Представлены формулы для расчета вероятности того, что поступившая в систему заявка либо будет обслужена, либо будет сброшена. Также в терминах ПЛС получены выражения для стационарного распределения времени пребывания в накопителе обслуженной и сброшенной заявок, а также средние временные характеристики при

обслуживании и сбросе заявок либо в порядке поступления, либо в инверсионном порядке.

5. В среде GPSS построены имитационные модели для всех рассмотренных случаев, доказано, что средние значения основных характеристик не зависят от дисциплины обслуживания/обновления.

Полученные в рассмотренных моделях аналитические выражения можно применять, в частности, для нахождения оценки таких показателей качества как: общей задержки передачи сообщения (среднее время пребывания в системе обслуженной заявки), среднего числа переданных пакетов (среднее число обслуженных заявок), вероятность потери принятого в систему пакета, вероятность дальнейшей передачи пакета, а также для оценки некоторых алгоритмов управления трафиком, рассмотренных в первой главе.

Список литературы

1. A fairness study of the adaptive RIO active queue management algorithm / R. Cartas [и др.] // Proceedings of the Fifth Mexican International Conference in Computer Science, 2004. ENC 2004. — 2004. — С. 57—63. — DOI: 10.1109/ENC.2004.1342589.
2. A Novel Active Queue Management Algorithm Based on Average Queue Length Change Rate / C. Pan [и др.] // IEEE Access. — 2022. — Т. 10. — С. 75558—75570. — DOI: 10.1109/ACCESS.2022.3189183.
3. A Novel Congestion Control Algorithm Using Buffer Occupancy RED / S. Singha [и др.] // Computational Intelligence in Pattern Recognition / под ред. А. К. Das [и др.]. — Singapore : Springer Singapore, 2022. — С. 519—528. — DOI: 10.1007/978-981-16-2543-5_44.
4. A self-configuring RED gateway / W.-C. Feng [и др.] // IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320). Т. 3. — 1999. — 1320—1328 vol.3. — DOI: 10.1109/INFCOM.1999.752150.
5. A Tutorial on Mathematical Modeling of 5G/6G Millimeter Wave and Terahertz Cellular Systems / D. Moltchanov [и др.] // IEEE Communications Surveys & Tutorials. — 2022. — Т. 24, № 2. — С. 1072—1116. — DOI: 10.1109/COMST.2022.3156207.

6. *Abbas G., Halim Z., Abbas Z. H.* Fairness-Driven Queue Management: A Survey and Taxonomy // *IEEE Communications Surveys & Tutorials*. — 2016. — Т. 18, № 1. — С. 324–367. — DOI: 10.1109/COMST.2015.2463121.
7. *Adams R.* Active Queue Management: A Survey // *Communications Surveys & Tutorials, IEEE*. — 2013. — Т. 15. — С. 1425–1476. — DOI: 10.1109/SURV.2012.082212.00018.
8. *Adamu A., Surajo Y., Jafar M. T.* SARED: A Self-Adaptive Active Queue Management Scheme for Improving Quality of Service in Network Systems // *Computer Science*. — 2021. — Т. 22. — DOI: 10.7494/csci.2021.22.2.4020.
9. An AQM based congestion control for eNB RLC in 4G/LTE network / A. K. Paul [и др.] // 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). — 2016. — С. 1–5. — DOI: 10.1109/CCECE.2016.7726792.
10. Analysis of an M|G|1|R queue with batch arrivals and two hysteretic overload control policies / Y. Gaidamaka [и др.] // *International Journal of Applied Mathematics and Computer Science*. — 2014. — Т. 24. — С. 519–534. — DOI: 10.2478/amcs-2014-0038.
11. Analysis of Multi-Server Queue with Self-Sustained Servers / A. Dudin [и др.] // *Mathematics*. — 2021. — Т. 9, № 17. — DOI: 10.3390/math9172134. — URL: <https://www.mdpi.com/2227-7390/9/17/2134>.

12. Analysis of Single-Server Multi-Class Queue with Unreliable Service, Batch Correlated Arrivals, Customers Impatience, and Dynamical Change of Priorities / A. Dudin [и др.] // Mathematics. — 2021. — Т. 9, № 11. — DOI: 10.3390/math9111257. — URL: <https://www.mdpi.com/2227-7390/9/11/1257>.
13. *Andrzej Chydzinski L. C.* Analysis of AQM queues with queue size based packet dropping // International Journal of Applied Mathematics and Computer Science. — 2011. — Т. 21, № 3. — С. 567—577. — URL: <http://eudml.org/doc/208071>.
14. *Anjum F. M., Tassiulas L.* Balanced RED: an Algorithm to Achieve Fairness in the Internet : tex. отч. — 1999. — URL: <http://www.dtic.mil/dtic/tr/fulltext/u2/a439654.pdf>.
15. *Atencia I., Bocharov P. P.* On the $M/G/1/0$ queueing system under LCFS PR discipline with repeated and negative customers with batch arrivals // Proc. 3-rd Europ. Cong. Math. — Barcelona, 2000. — С. 133.
16. *Aweya J., Ouellette M., Montuno D. Y.* A control theoretic approach to active queue management // Computer Networks. — 2001. — Т. 36, № 2. — С. 203—235. — DOI: 10.1016/S1389-1286(00)00206-1. — Theme issue: Overlay Networks.
17. *Baker F., Fairhurst G.* IETF Recommendations Regarding Active Queue Management : tex. отч. / Internet Engineering Task Force. — 2015. — URL: <https://tools.ietf.org/html/rfc7567>.

18. *Bayer N., Boxma O. J.* Wiener-Hopf analysis of an $M/G/1$ queue with negative customers and of related class of random walk // Queueing Systems. — 1996. — № 23. — С. 301–316.
19. *Bocharov P. P., D'Apice C., Pechinkin A. V.* Queueing Theory. — Berlin, Boston : De Gruyter, 2003. — ISBN 9783110936025. — DOI: doi:10.1515/9783110936025.
20. *Bocharov P. P., Naoumov V. A.* Matrix-geometric stationary distribution for the $PH/PH/1/r$ queue // Elektronische Informationsverarbeitung und Kybernetik. — 1986. — Т. 22, № 4. — С. 179–186.
21. *Bogdanova E. V., Milovanova T. A., Zaryadov I. S.* The Analysis of Queueing System with General Service Distribution and Renovation // Вестник РУДН. Серия «Математика, информатика, физика». — 2017. — Т. 25, № 1. — С. 3–8.
22. *Bogdanova E. V., Zaryadov I. S., Milovanova T. A.* Probability-Time Characteristics of $M|G|1|1$ Queueing System with Renovation // CEUR Workshop Proceedings. Т. 1995. — CEUR, 2017. — С. 125–131. — URL: <https://ceur-ws.org/Vol-1995/paper-17-981.pdf>.
23. *Chakka R., Harrison P. G.* A Markov modulated multi-server queue with negative customers — The $MMCPP/GE/c/L$ G-queue // Acta Informatika. — 2001. — Т. 37. — С. 881–919.
24. Characteristics of Lost and Served Packets for Retrial Queueing System with General Renovation and Recurrent Input Flow / E. V.

- Bogdanova [и др.] // Distributed Computer and Communication Networks. Communications in Computer and Information Science. T. 919 / под ред. V. M. Vishnevskiy, D. V. Kozyrev. — Cham : Springer International Publishing, 2018. — С. 327—340. — DOI: doi:10.1007/978-3-030-36625-4_39.
25. *Chrost L., Chydzinski A.* On the Evaluation of the Active Queue Management Mechanisms // 2009 First International Conference on Evolving Internet. — 2009. — С. 113—118. — DOI: 10.1109/INTERNET.2009.25.
26. *Chung J., Claypool M.* Rate-based active queue management with priority classes for better video transmission // Proceedings ISCC 2002 Seventh International Symposium on Computers and Communications. — 2002. — С. 99—105. — DOI: 10.1109/ISCC.2002.1021664.
27. *Chydzinski A., Adamczyk B.* Queues with the dropping function and general service time // PLoS ONE. — 2019. — Т. 14, № 7. — e0219444. — DOI: 10.1371/journal.pone.0219444.
28. *Chydzinski A.* Non-Stationary Characteristics of AQM Based on the Queue Length // Sensors. — 2023. — Т. 23, № 8. — С. 485. — DOI: 10.3390/s23010485. — URL: <https://www.mdpi.com/1099-4300/22/8/825>.
29. *Chydzinski A.* Queues With the Dropping Function and Non-Poisson Arrivals // IEEE Access. — 2020. — Т. 8. — С. 39819—39829. — DOI: 10.1109/ACCESS.2020.2976147.

30. *Chydziński A., Brachman A.* Performance of AQM Routers in the Presence of New TCP Variants // 2010 Second International Conference on Advances in Future Internet. — 2010. — С. 88–93. — DOI: 10.1109/AFIN.2010.22.
31. Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2. Тех. отч. / Cisco. — URL: http://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c.html.
32. *Clark D., Fang W.* Explicit allocation of best-effort packet delivery service // IEEE/ACM Transactions on Networking. — 1998. — Т. 6, № 4. — С. 362–373. — DOI: 10.1109/90.720870.
33. Class-Based Weighted Fair Queueing and Weighted Random Early Detection : тех. отч. / Cisco. — URL: https://www.cisco.com/en/US/docs/ios/12_0t/12_0t5/feature/guide/cbwfq.html.
34. Congestion control in differentiated services networks using Fuzzy-RED / C. Chrysostomou [и др.] // Control Engineering Practice. — 2003. — Т. 11, № 10. — С. 1153–1170. — DOI: 10.1016/S0967-0661(03)00052-2. — Special Section on Control Methods for Telecommunication.
35. Congestion Control Scheme Performance Analysis Based on Nonlinear RED / C.-W. Feng [и др.] // IEEE Systems Journal. — 2017. — Т. 11, № 4. — С. 2247–2254. — DOI: 10.1109/JSYST.2014.2375314.

36. Controlled Delay Active Queue Management : тех. отч. / K. Nichols [и др.] ; Internet Engineering Task Force. — 2018. — URL: <https://tools.ietf.org/html/rfc8289>.
37. *Danladi S. B., Ambursa F. U.* DyRED: An Enhanced Random Early Detection Based on a new Adaptive Congestion Control // 2019 15th International Conference on Electronics, Computer and Computation (ICECCO). — 2019. — С. 1–5. — DOI: [10.1109/ICECCO48375.2019.9043276](https://doi.org/10.1109/ICECCO48375.2019.9043276).
38. *De Cnodder S., Pauwels K., Elloumi O.* A rate based RED mechanism // Proceedings of the 10th ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV). — Chapel Hill, NC, US, 2000. — URL: <http://www.nossdav.org/2000/papers/19.pdf>.
39. *Do T. V.* Bibliography on G-networks, negative customers and applications // Mathematical and Computer Modelling. — 2011. — Т. 53, № 1. — С. 205–212. — ISSN 0895-7177. — DOI: <https://doi.org/10.1016/j.mcm.2010.08.006>.
40. *Dudin A. N., Karolik A. V.* BMAP/SM/1 queue with Markovian input of disasters and non-instantaneous recovery // Performance Evaluation. — 2001. — Т. 45, № 1. — С. 19–32.
41. *Dudin A. N., Klimenok V. I., Vishnevsky V. M.* The Theory of Queuing Systems with Correlated Flows. — Springer Nature Switzerland AG : Springer Cham, 2020.

42. *Erlang A. K.* Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges // The Post Office Electrical Engineers Journal. — 1917. — Т. 10. — С. 189—197.
43. Experimental study of assured services in a diffserv IP QoS network / N. Seddigh [и др.] // Internet Routing and Quality of Service. Т. 3529 / под ред. R. O. Onvural [и др.]. — International Society for Optics, Photonics. SPIE, 1998. — С. 217—230. — DOI: 10.1117/12.333712.
44. *Feng W.-c., Kandlur D., Saha D.* The BLUE active queue management algorithms // Networking, IEEE/ACM Transactions on. — 2002. — Т. 10. — С. 513—528. — DOI: 10.1109/TNET.2002.801399.
45. *Feng W.-C.* Improving Internet Congestion Control and Queue Management Algorithms. Doctor of Philosophy dissertation : тех. отч. / The University of Michigan. — 1999. — URL: http://thefengs.com/wuchang/umich_diss.html.
46. *Feng W., Kandlur D. D. and Saha D., Shin K. G.* BLUE: A New Class of Active Queue Management Algorithms : тех. отч. / The University of Michigan. — 1999. — URL: <https://www.cse.umich.edu/techreports/cse/99/CSE-TR-387-99.pdf>.
47. Flexible Random Early Detection Algorithm for Queue Management in Routers / A. Adamu [и др.] // Distributed Computer and Communication Networks / под ред. V. M. Vishnevskiy, K. E. Samouylov, D. V. Kozyrev. — Cham : Springer International Publishing, 2020. — С. 196—208. — DOI: 10.1007/978-3-030-66471-8_16.

48. *Floyd S.* Recommendation on using the «gentle variant of RED» : тех. отч. — 2000. — URL: <http://www.icir.org/floyd/red/gentle.html>.
49. *Floyd S., Jacobson V.* Random Early Detection Gateways for Congestion Avoidance // IEEE/ACM Transactions on Networking. — 1993. — 1(4). — C. 397—413.
50. *Floyd S., Gummadi R., Shenker S.* Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. — 2001. — URL: <http://www.icir.org/floyd/papers/adaptiveRed.pdf>.
51. *Fourneau J. M., Gelenbe E., Suros R.* G-networks with multiple classes of positive and negative customers // Computer Science. — 1996. — T. 155. — C. 141—156.
52. *Gelenbe E.* G-Networks with triggered customer movement // Journal of Applied Probability. — 1993. — T. 30, № 3. — C. 742—748.
53. *Gelenbe E.* G-networks: an unifying model for neural and queueing networks // Annals of Operations Research. — 1994. — T. 48, № 1—4. — C. 433—461.
54. *Gelenbe E.* Product-Form Queueing Networks with Negative and Positive Customers // Journal of Applied Probability. — 1991. — T. 28. — C. 656—663.
55. *Gelenbe E.* Random neural networks with negative and positive signals and product form solution // Neural Computation. — 1989. — T. 1, № 4. — C. 502—510.

56. *Gelenbe E., Glynn P., Sigman K.* Queues with negative arrivals // Journal of Applied Probability. — 1991. — T. 28. — C. 245—250.
57. *Gelenbe E., Nakp M.* G-Networks Can Detect Different Types of Cyberattacks // 2022 30th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS). — 2022. — C. 9—16. — DOI: 10.1109/MASCOTS56607.2022.00010.
58. *George J., Santhosh R.* Congestion Control Mechanism for Unresponsive Flows in Internet Through Active Queue Management System (AQM) // Mobile Computing and Sustainable Informatics / под ред. S. Shakya [и др.]. — Singapor : Springer Singapore, 2022. — C. 765—777. — DOI: 10.1007/978-981-16-1866-6_58.
59. *Grieco L. A., Mascolo S.* TCP Westwood and Easy RED to Improve Fairness in High-Speed Networks // Protocols for High-Speed Networks. — Berlin, Heidelberg : Springer, 2002. — C. 130—146.
60. *Gymez-Corral A., Martos M. E.* Marked Markovian Arrivals in a Tandem G-Network with Blocking // Methodol. Comput. Appl. Probab. — 2009. — № 11. — C. 621—649.
61. *Harrison P. G., Pitel E.* Sojourn times in single server queues with negative customers // Queueing Systems. — 2002. — № 41. — C. 943—963.

62. *Hilquias V. C. C., Zaryadov I. S.* Comparison of two single-server queueing systems with exponential service times and threshold-based renovation // CEUR Workshop Proceedings. Т. 2946. — CEUR, 2021. — С. 54—63. — URL: <https://ceur-ws.org/Vol-2946/paper-05.pdf>.
63. *Hilquias V. C. C., Zaryadov I. S., Milovanova T. A.* Queuing system with threshold-based general renovation mechanism // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2022). Материалы XXV Международной научной конференции / под ред. В. М. Вишнеvский, К. Е. Самуилов. — Москва, 2022. — С. 158—167.
64. *Hilquias V. C. C., Zaryadov I. S., Milovanova T. A.* Single-server queueing systems with exponential service times and threshold-based renovation // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2021). Материалы XXIV Международной научной конференции / под ред. В. М. Вишнеvский, К. Е. Самуилов. — Москва, 2021. — С. 229—236.
65. *Hilquias V. C. C., Zaryadov I. S., Milovanova T. A.* Two Types of Single-Server Queueing Systems with Threshold-Based Renovation Mechanism // Distributed Computer and Communication Networks: Control, Computation, Communications. Lecture Notes in Computer Science. Т. 13144 / под ред. V. M. Vishnevskiy, K. E. Samouylov, D. V. Kozyrev. — Cham : Springer International Publishing, 2021. — С. 196—210. — DOI: [doi:10.1007/978-3-030-92507-9_17](https://doi.org/10.1007/978-3-030-92507-9_17).

66. *Hilquias V. C. C., Zaryadov I. S.* Single-server Queuing Systems With Exponential Service Times and Threshold-based Renovation // 13th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). — 2021. — С. 91—97. — DOI: 10.1109/ICUMT54235.2021.9631585.
67. *Hilquias V. C., Zaryadov I. S., Matyushenko S. I.* GPSS simulation model of a system with a general renovation threshold based mechanism // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. Материалы Всероссийской конференции с международным участием. — Москва, 2023. — С. 20—25.
68. *Hu L., Kshemkalyani A.* HRED: a simple and efficient active queue management algorithm // Proceedings. 13th International Conference on Computer Communications and Networks (IEEE Cat. No.04EX969). — 2004. — С. 387—393. — DOI: 10.1109/ICCCN.2004.1401681.
69. Hybrid Simulation Of Active Traffic Management / A. V. Korolkova [и др.] // Proceedings 30th European Conference on Modelling and Simulation. — egsburg, Germany : ECMS, 2016. — С. 685—691. — URL: https://www.scs-europe.net/dlib/2016/ecms2016acceptedpapers/0685-probstat_ECMS_0076.pdf.
70. IGRED: An Improved Gentle Random Early Detection Method for Management of Congested Networks / H. Abdel-Jaber [и др.] // Journal

- of Interconnection Networks. — 2019. — T. 19. — C. 1950004. — DOI: 10.1142/S021926591950004X.
71. Improving fairness of RED aided by lightweight flow information / S. Hanlin [и др.] // 2009 2nd IEEE International Conference on Broadband Network & Multimedia Technology. — 2009. — C. 335—339.
72. *Irazabal M., Lopez-Aguilera E., Demirkol I.* Active Queue Management as Quality of Service Enabler for 5G Networks // 2019 European Conference on Networks and Communications (EuCNC). — 2019. — C. 421—426. — DOI: 10.1109/EuCNC.2019.8802027.
73. *Jacobson V., Nichols K. M., Poduri K.* RED in a Different Light //. — 1999. — URL: <https://citeseerx.ist.psu.edu/doc/10.1.1.22.9406>.
74. *Jafri S. T. A., Ahmed I., Ali S.* Queue-Buffer Optimization Based on Aggressive Random Early Detection in Massive NB-IoT MANET for 5G Applications // Electronics. — 2022. — T. 11, № 18. — ISSN 2079-9292. — DOI: 10.3390/electronics11182955. — URL: <https://www.mdpi.com/2079-9292/11/18/2955>.
75. *Jin-Ting W., Peng Z.* A Single-server Discrete-time Retrial G-queue with Server Breakdowns and Repairs // Acta Mathematicae Applicatae Sinica. — 2009. — T. 25, № 4. — C. 675—684.

76. *Kendall D. J.* Stochastic processes occurring in the theory of queues and their analysis by the method of the embedded Markov chains // *Ann. Math. Stat.* — 1953. — № 24. — С. 338—354.
77. *Kim C., Klimenok V. I., Dudin A. N.* Analysis of unreliable *BMAP|PH|N* type queue with Markovian flow of breakdowns // *Applied Mathematics and Computation.* — 2017. — № 314. — С. 154—172. — DOI: 10.1016/j.amc.2017.06.035.
78. *Kim C., Klimenok V. I., Orlovsky D. S.* Multi-Server Queueing System with a Batch Markovian Arrival Process and Negative Customers // *Automation and Remote Control.* — 2006. — № 12. — С. 106—122.
79. *Kim T.-h., Lee K.-h.* Refined Adaptive RED in TCP/IP Networks // 2006 SICE-ICASE International Joint Conference. — 2006. — С. 3722—3725. — DOI: 10.1109/SICE.2006.314633.
80. *Kleinrock L.* Queueing systems. — Brisbane, Toronto : John Wiley & Sons, 1975.
81. *Klimenok V., Dudin A., Vishnevsky V.* Queueing System with Two Unreliable Servers and Backup Server as a Model of Hybrid Communication System // *Distributed Computer and Communication Networks* / под ред. V. M. Vishnevskiy, K. E. Samouylov, D. V. Kozyrev. — Cham : Springer International Publishing, 2020. — С. 176—195.

82. *Konovalov M., Razumchik R.* Comparison of two active queue management schemes through the M/D/1/N queue // *Informatika i ee Primeneniya (Informatics and Applications)*. — 2017. — T. 12, № 4. — С. 9—15. — DOI: 10.14357/19922264180402.
83. *Konovalov M., Razumchik R.* Finite capacity single-server queue with Poisson input, general service and delayed renovation // *European Journal of Operational Research*. — 2023. — T. 304, № 3. — С. 1075—1083. — DOI: 10.1016/j.ejor.2022.05.047.
84. *Konovalov M., Razumchik R.* Numerical Analysis of Improved Access Restriction Algorithms in a GI/G/1/N System // *Journal of Communications Technology and Electronics*. — 2018. — T. 63. — С. 616—625. — DOI: 10.1134/S1064226918060141.
85. *Konovalov M., Razumchik R.* Queueing systems with renovation vs. queues with RED. Supplementary Material. — 2017. — arXiv: 1709.01477 [cs.PF]. — URL: <https://arxiv.org/abs/1709.01477>.
86. *Korolkova A. V., Zaryadov I. S.* The Mathematical Model of the Traffic Transfer Process with a Rate Adjustable by RED // *ICUMT 2010 — International Conference on Ultra Modern Telecommunications*. — Moscow, 2010. — С. 1—5.
87. *Kreinin A.* Inhomogeneous Random Walks: Applications in Queueing and Finance // *CanQueue*. — Fields Institute. Toronto, 2003.

88. *Kreinin A.* Queueing Systems with Renovation // Journal of Applied Math. Stochast. Analysis. — 1997. — T. 10, № 4. — С. 431—443.
89. *Li J., Zhang L.* $M^X|M|c$ Queue with Catastrophes and State-Dependent Control at Idle Time // Frontiers of Mathematics in China. — 2017. — T. 12, № 6. — С. 1427—1439. — DOI: 10.1007/s11464-017-0674-8.
90. *Lin D., Morris R.* Dynamics of Random Early Detection //. — Cannes, France : Association for Computing Machinery, 1997. — С. 127—137. — (SIGCOMM '97). — DOI: 10.1145/263105.263154. — URL: <http://nil.lcs.mit.edu/rtm/papers/fred.pdf>.
91. LRED: a robust active queue management scheme based on packet loss ratio / C. Wang [и др.] // IEEE INFOCOM 2004. T. 1. — 2004. — С. 12. — DOI: 10.1109/INFCOM.2004.1354476.
92. *Malinkovsky Y.* Stationary Distribution of Queueing Networks with Countable Set of Types of Batch Negative Customer Arrivals // Information Technologies and Mathematical Modelling - Queueing Theory and Applications / под ред. А. Dudin [и др.]. — Cham : Springer International Publishing, 2016. — С. 221—227. — DOI: 10.1007/978-3-319-44615-8_19.
93. *Mandili H. el, Nsiri B.* A new type of negative customers // International Journal of Computing Science and Mathematics. — 2017. — Т. 8. — С. 193. — DOI: 10.1504/IJCSM.2017.083751.

94. *Manzo R., Cascone N., Razumchik R. V.* Exponential Queuing System with Negative Customers and Bunker for Ousted Customers // Automation and Remote Control. — 2008. — № 9. — С. 103–113.
95. *Menth M., Veith S.* Active Queue Management Based on Congestion Policing (CP-AQM) // Measurement, Modelling and Evaluation of Computing Systems / под ред. R. German, K.-S. Hielscher, U. R. Krieger. — Cham : Springer International Publishing, 2018. — С. 173–187. — DOI: 10.1007/978-3-319-74947-1_12.
96. Modeling and Analyzing Licensed Shared Access Operation for 5G Network as an Inhomogeneous Queue with Catastrophes / I. Gudkova [и др.] // International Congress on Ultra Modern Telecommunications and Control Systems and Workshops. — IEEE. Lisbon, Portugal, 2016. — С. 282–287. — DOI: 10.1109/ICUMT.2016.7765372.
97. *Naoumov V. A.* Matrix-Multiplicative Approach to Quasi-Birth-and-Death Processes Analysis // Matrix-Analytic Methods in Stochastic Models / под ред. M. D. Inc. — New-York, 1997. — С. 87–107.
98. *Naumov V., Samouylov K.* Resource System with Losses in a Random Environment // Mathematics. — 2021. — Т. 9, № 21. — DOI: 10.3390/math9212685. — URL: <https://www.mdpi.com/2227-7390/9/21/2685>.
99. *Naumov V. A., Gaidamaka Y. V., Samouylov K. E.* On Two Interacting Markovian Queueing Systems // Mathematics. — 2019. — Т. 7, № 9. —

- DOI: 10.3390/math7090799. — URL: <https://www.mdpi.com/2227-7390/7/9/799>.
100. Network Slice Degradation Probability as a Metric for Defining Slice Performance Isolation / N. Polyakov [и др.] // Internet of Things, Smart Spaces, and Next Generation Networks and Systems / под ред. Y. Koucheryavy, S. Balandin, S. Andreev. — Cham : Springer International Publishing, 2022. — С. 481—492.
 101. *Neuts M. F.* Matrix-geometric solutions in stochastic models. An algorithmic approach. — Baltimore, London : The Johns Hopkins Univ. Press, 1981.
 102. *Neuts M. F.* Structured Stochastic Matrices of M/G/1 Type and Their Applications. — New York : Marcel Dekker Inc., 1989.
 103. New RED-Type TCP-AQM Algorithms Based on Beta Distribution Drop Functions / A. Giménez [и др.] // Applied Sciences. — 2022. — Т. 12, № 21. — DOI: 10.3390/app122111176. — URL: <https://www.mdpi.com/2076-3417/12/21/11176>.
 104. *Ng P. C. H., Boon-Hee P. S.* Queueing Modelling Fundamentals: With Applications in Communication Networks. — John Wiley & Sons, 2008.
 105. *Nichols K., Jacobson V.* Controlling Queue Delay: A Modern AQM is Just One Piece of the Solution to Bufferbloat // Queue. — New York, NY, USA, 2012. — Т. 10, № 5. — С. 20—34. — DOI: 10.1145/2208917.2209336.

106. On active queue management in cellular networks / J. Beshay [и др.] // . — 2017. — С. 384—389. — DOI: 10.1109/INFCOMW.2017.8116407.
107. *Orozco J., Ros D.* An Adaptive RIO (A-RIO) Queue Management Algorithm // Quality for All / под ред. G. Karlsson, M. I. Smirnov. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2003. — С. 11—20. — DOI: 10.1007/978-3-540-45188-4_2.
108. *Ott T., Lakshman T., Wong L.* SRED: stabilized RED // IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320). T. 3. — 1999. — С. 1346—1355. — DOI: 10.1109/INFCOM.1999.752153.
109. *Patel C. M.* URED: Upper threshold RED an efficient congestion control algorithm // 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT). — 2013. — С. 1—5. — DOI: 10.1109/ICCCNT.2013.6726469.
110. *Pechinkin A. V., Razumchik R. V.* The Stationary Distribution of the Waiting Time in a Queueing System with Negative Customers and a Bunker for Superseded Customers in the Case of the LAST-LIFO-LIFO Discipline // Journal of Communications Technology and Electronics. — 2012. — Т. 57, № 12. — С. 1331—1339.
111. *Pechinkin A., Razumchik R.* Sojourn times in different modes in the queueing system with batch Poisson arrivals, general service times, and hysteretic policy // Journal of Communications Technology and

- Electronics. — 2016. — Т. 61. — С. 679—687. — DOI: 10.1134/S1064226916060188.
112. Performance Analysis of Multi-Band Microwave and Millimeter-Wave Operation in 5G NR Systems / V. Begishev [и др.] // IEEE Transactions on Wireless Communications. — 2021. — Т. 20, № 6. — С. 3475—3490. — DOI: 10.1109/TWC.2021.3051027.
113. Performance evaluation of an M/G/n-type queue with bounded capacity and packet dropping / O. Tikhonenko [и др.] // International Journal of Applied Mathematics and Computer Science. — 2016. — Т. 26. — С. 841—854. — DOI: 10.1515/amcs-2016-0060.
114. *Phirke V., Claypool M., Kinicki R.* RED-Worcester - traffic sensitive active queue management // 10th IEEE International Conference on Network Protocols, 2002. Proceedings. — 2002. — С. 194—195. — DOI: 10.1109/ICNP.2002.1181403.
115. POWARED for Non-Linear Adaptive RED / B. Ng [и др.] // 2005 Asia-Pacific Conference on Communications. — 2005. — С. 832—836. — DOI: 10.1109/APCC.2005.1554179.
116. Preemptive Priority Queuing System with Randomized Push-Out Mechanism and Negative Customers / P. Shorenko [и др.] // Internet of Things, Smart Spaces, and Next Generation Networks and Systems / под ред. О. Galinina [и др.]. — Cham : Springer International Publishing, 2019. — С. 305—317. — DOI: 10.1007/978-3-030-30859-9_26.

117. *Quan-Lin L., Yiqiang Q. Z.* A MAP/G/1 Queue with Negative Customers // Queueing Systems. — 2004. — № 47. — С. 5—43.
118. *Que D., Chen Z., Chen B.* An improvement algorithm based on RED and its performance analysis // 2008 9th International Conference on Signal Processing. — 2008. — С. 2005—2008. — DOI: 10.1109/ICOSP.2008.4697538.
119. Queueing Theory / P. P. Bocharov [и др.]. — Utrecht, Boston : VSP, 2004.
120. *Al-Raddady F., Woodward M.* A New Adaptive Congestion Control Mechanism for the Internet Based on RED // 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07). Т. 2. — 2007. — С. 934—939. — DOI: 10.1109/AINAW.2007.31.
121. *Ramakrishnan K., Floyd S., Black D.* The Addition of Explicit Congestion Notification (ECN) to IP. RFC 3168 : тех. отч. / Internet Engineering Task Force. — 2020. — URL: <https://datatracker.ietf.org/doc/rfc3168/>.
122. Random early detection-quadratic linear: an enhanced active queue management algorithm / S. Hassan [и др.] // Bulletin of Electrical Engineering and Informatics. — 2022. — Т. 11, № 4. — С. 2262—2272. — DOI: 10.11591/eei.v11i4.3875.

123. *Razumchik R. V.* Analysis of Finite Capacity Queue with Negative Customers and Bunker for Ousted Customers Using Chebyshev and Gegenbauer Polynomials // Asia-Pacific Journal of Operational Research. — 2014. — Т. 31, № 04. — С. 1450029.
124. Revisiting the Gentle Parameter of the Random Early Detection (RED) for TCP Congestion Control / N. Hamadneh [и др.] // Journal of Communications. — 2019. — Т. 14. — С. 229–235. — DOI: 10.12720/jcm.14.3.229-235.
125. RRED: robust RED algorithm to counter low-rate denial-of-service attacks / C. Zhang [и др.] // IEEE Communications Letters. — 2010. — Т. 14, № 5. — С. 489–491. — DOI: 10.1109/LCOMM.2010.05.091407.
126. *Ryoo I., Yang M.* A State Dependent RED: An Enhanced Active Queue Management Scheme for Real-Time Internet Services // IEICE Trans. Commun. — 2006. — Т. E89–B, № 2. — С. 614–617.
127. *Ryoo I.-t., Yang M.* A State Dependent RED: An Enhanced Active Queue Management Scheme for Real-Time Internet Services // IEICE Trans. Commun. — 2006. — Т. 89–B. — С. 614–617.
128. Sally Floyd. — URL: <http://www.icir.org/floyd/>.
129. *Semenova O. V., Dudin A. N.* $M/M/N$ queueing system with controlled service mode and disaster // Automat. Control Comput. Sci. — 2007. — Т. 41, № 6. — С. 350–357.

130. Stationary characteristics of the $GI|M|1$ queue with general renovation and feedback / I. Zaryadov [и др.] // 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). — IEEE, 2019. — DOI: 10.1109/ICUMT54235.2021.9631585.
131. *Tahiliani M. P., Shet K. C., Basavaraju T. G.* FARED: Fast Adapting RED Gateways for TCP/IP Networks // Advanced Computing, Networking and Security / под ред. P. S. Thilagam [и др.]. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2012. — С. 435—443. — DOI: 10.1007/978-3-642-29280-4_51.
132. *Tahiliani M. P., Shet K., Basavaraju T.* CARED: Cautious Adaptive RED gateways for TCP/IP networks // Journal of Network and Computer Applications. — 2012. — Т. 35, № 2. — С. 857—864. — DOI: 10.1016/j.jnca.2011.12.003. — Simulation and Testbeds.
133. *Takacs L.* Introduction to the theory of queues. — New York : Oxford University Press, 1962.
134. Techniques for Eliminating Packet Loss in Congested TCP/IP Networks : тех. отч. / W.-c. Feng [и др.]. — 1997. — URL: <https://www.eecs.umich.edu/techreports/cse/97/CSE-TR-349-97.pdf>.
135. The General Renovation as the Active Queue Management Mechanism. Some Aspects and Results / V. C. C. Hilquias [и др.] // Distributed Computer and Communication Networks. Communications in Computer and Information Science. Т. 1141 / под ред. V. M. Vishnevskiy, K. E.

- Samouylov, D. V. Kozyrev. — Cham : Springer International Publishing, 2019. — С. 488—502. — DOI: doi:10.1007/978-3-030-36625-4_39.
136. The Queueing System with Threshold-Based Direct and Inverse General Renovation Mechanism / V. C. C. Hilquias [и др.] // Distributed Computer and Communication Networks: Control, Computation, Communications. Lecture Notes in Computer Science. Т. 13766 / под ред. V. M. Vishnevskiy, K. E. Samouylov, D. V. Kozyrev. — Cham : Springer Nature Switzerland, 2022. — С. 309—323. — DOI: doi:10.1007/978-3-031-23207-7_24.
137. *Towsley D., Tripathi S. K.* A single server priority queue with server failure and queue flushing // Oper. Res. Lett. — 1991. — № 10. — С. 353—362.
138. Transient analysis of a single server queue with catastrophes, failures and repairs / B. Krishna Kumar [и др.] // Queueing Systems: Theory and Applications. — 2007. — Т. 56, № 3/4. — С. 133—141.
139. *Wu H., Wu C., Lin W.* SF-RED - A Novel Server-based AQM to Provide Inter-server Fairness Service // 12th International Conference on Parallel and Distributed Systems, ICPADS 2006, Minneapolis, Minnesota, USA, July 12-15, 2006. — IEEE Computer Society, 2006. — С. 501—506. — DOI: 10.1109/ICPADS.2006.99.
140. *Xue L.* The Implementation of An Improved ARED Congestion Control Algorithm // 2020 13th International Symposium on Computational

- Intelligence and Design (ISCID). — 2020. — С. 22–25. — DOI: 10.1109/ISCID51228.2020.00012.
141. *YangWoo S.* Multi-server retrial queue with negative customers and disasters // *Queueing Syst.* — 2007. — № 55. — С. 223–237.
142. *Yechiali U.* Queues with system disasters and impatient customers when system is down // *Queueing Syst. Theory Appl.* — 2007. — Т. 56, № 3/4. — С. 195–202.
143. *Zaryadov I. S.* Queueing Systems with General Renovation // ICUMT 2009 — International Conference on Ultra Modern Telecommunications. — St.-Petersburg, 2009. — С. 1–6.
144. *Zaryadov I. S., Gorbunova A. V.* The Analysis of Queueing System with Two Input Flows and Stochastic Drop Mechanism // *Вестник РУДН. Серия «Математика, информатика, физика».* — 2015. — № 2. — С. 33–37.
145. *Zaryadov I. S., Scherbanskaya A. A.* Time Characteristics of Queueing System with Renovation and Reservice // *Вестник РУДН. Серия «Математика, информатика, физика».* — 2014. — № 2. — С. 61–66.
146. *Zaryadov I., Razumchik R., Milovanova T.* Stationary Waiting Time Distribution in $G|M|n|r$ with Random Renovation Policy // *Distributed Computer and Communication Networks. Communications in Computer and Information Science.* Т. 678 / под ред. V. M. Vishnevskiy, K. E.

- Samouylov, D. V. Kozyrev. — Cham : Springer International Publishing, 2016. — С. 349—360. — DOI: doi:10.1007/978-3-319-51917-3_31.
147. *Zhang Y., Ma J., Wang Y. and Xu C.* MRED: an improved nonlinear RED algorithm // International Conference Proceedings on Computer and Automation Engineering (ICCAE 2011). Т. 44. — 2012. — С. 6—11.
148. *Zheng B., Atiquzzaman M.* DSRED: an active queue management scheme for next generation networks // Proceedings 25th Annual IEEE Conference on Local Computer Networks. LCN 2000. — 2000. — С. 242—251. — DOI: 10.1109/LCN.2000.891036.
149. *Zheng B., Atiquzzaman M.* DSRED: A New Queue Management Scheme for the Next Generation Internet // IEICE Transactions on Communications. — 2006. — Т. E89B. — DOI: 10.1093/ietcom/e89-b.3.764.
150. *Zhou K., Yeung K., Li V.* Nonlinear RED: A simple yet efficient active queue management scheme // Computer Networks. — 2006. — Т. 50. — С. 3784—3794. — DOI: 10.1016/j.comnet.2006.04.007.
151. Анализ многолинейной марковской системы массового обслуживания с неограниченным накопителем и отрицательными заявками / П. П. Бочаров [и др.] // Автоматика и телемеханика. — 2007. — № 1. — С. 93—104.
152. Анализ показателей функционирования RED-подобных алгоритмов с помощью систем массового обслуживания / В. К. К. Илкиаш [и др.]

- др.] // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. Материалы Всероссийской конференции с международным участием. — Москва, 2019. — С. 58—63.
153. *Башарин Г. П.* Лекции по математической теории телетрафика. — М. : РУДН, 2004.
154. *Башарин Г. П., Бочаров П. П., Коган А. Я.* Анализ очередей в вычислительных сетях. Теория и методы расчёта. — М. : Наука, 1989.
155. *Башарин Г. П., Самуёлов К. Е.* Современный этап в развитии теории телетрафика // Информационная математика. — 2001. — Т. 1, № 1. — С. 153—166.
156. *Башарин Г. П., Харкевич А. Д., А. Ш. М.* Массовое обслуживание в телефонии. — М. : Наука, 1968.
157. *Боровков А. А.* Асимптотические методы в теории массового обслуживания. — М. : Наука, 1980.
158. *Боровков А. А.* Вероятностные процессы в теории массового обслуживания. — М. : Наука, 1972.
159. *Бочаров П. П., Вишневецкий В. М.* G-сети: развитие теории мультипликативных сетей // Автоматика и телемеханика. — 2003. — № 5. — С. 46—74.
160. *Бочаров П. П., Зарядов И. С.* Стационарное распределение вероятностей в системах массового обслуживания с обновлением // Вестник

- РУДН. Серия «Математика. Информатика. Физика». — 2007. — № 1/2. — С. 15—25.
161. *Бочаров П. П., Печинкин А. В.* Теория массового обслуживания. — М. : РУДН, 1995.
162. *Виана К. К. И., Зарядов И. С.* Вероятностно-временные характеристики однопороговой системы с обновлением // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. Материалы Всероссийской конференции с международным участием. — Москва, 2021. — С. 31—37.
163. *Виана К. К. И., Зарядов И. С.* Сравнение результатов имитационного моделирования двух типов однопороговых систем с обновлением // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. Материалы Всероссийской конференции с международным участием. — Москва, 2021. — С. 38—44.
164. *Виана Карвалью Кравид И., Зарядов И. С., Милованова Т. А.* Системы массового обслуживания с различными видами обновления и порогом как математические модели алгоритмов активного управления очередями // Discrete and Continuous Models and Applied Computational Science. — 2020. — Т. 28, № 4. — С. 305—318. — URL: <https://journals.rudn.ru/miph/article/view/25178>.
165. *Вишневецкий В. М.* Теоретические основы проектирования компьютерных сетей. — М. : Техносфера, 2003.

166. *Гнеденко Б. В., Коваленко И. Н.* Введение в теорию массового обслуживания. — М. : ЛКИ, 2007.
167. *Горбунова А. В., Лебедев А. В.* Система массового обслуживания с двумя входящими потоками, абсолютным приоритетом и стохастическим сбросом // Автоматика и телемеханика. — 2020. — № 12. — С. 111—128.
168. *Зарядов И. С.* Зависимость среднего времени пребывания заявки в системе от дисциплин обслуживания и обобщенного обновления // International Workshop “Distributed Computer and Communication Networks” 14-я Международная конференция «Распределенные компьютерные и телекоммуникационные сети: теория и приложения» (DCCN-2010). — Москва, 2010. — С. 224—231.
169. *Зарядов И. С.* Ненадёжные системы с различными вариантами обновления // Международная конференция MMR2009 – математические методы в теории надёжности. Теория. Методы. Приложения. — Москва, 2009. — С. 573—575.
170. *Зарядов И. С.* Расчет показателей качества функционирования систем передачи и обработки данных с помощью обобщенного обновления : дис. . . . канд. / Зарядов И. С. — Москва : Российский университет дружбы народов (РУДН), 2010.
171. *Зарядов И. С.* Система массового обслуживания $GI/M/n/\infty$ с обобщённым обновлением // Автоматика и телемеханика. — 2010. — № 4. — С. 130—139.

172. *Зарядов И. С.* Стационарные характеристики обслуживания в системе $G/M/n/r$ с обобщённым обновлением // Вестник РУДН. Серия «Математика. Информатика. Физика». — 2008. — № 2. — С. 3—10.
173. *Зарядов И. С.* Стационарные характеристики обслуживания системы $G/M/n/r$ с некоторыми вариантами дисциплины обобщённого обновления // Информационные процессы. — 2008. — Т. 8, № 2. — С. 108—124.
174. *Зарядов И. С., Виана И. К. К., Милованова Т. А.* Анализ систем массового обслуживания с пороговым механизмом обновления и инверсионной дисциплиной обслуживания // Discrete and Continuous Models and Applied Computational Science. — 2022. — Т. 30, № 2. — С. 160—182. — URL: <https://journals.rudn.ru/miph/article/view/30954>.
175. *Зарядов И. С., Горбунова А. В.* Анализ характеристик системы массового обслуживания с двумя входящими потоками, относительным приоритетом и сбросом // Современные информационные технологии и ИТ-образование. — 2014. — № 10. — С. 388—394.
176. *Зарядов И. С., Королькова А. В.* Применение модели с обобщённым обновлением к анализу характеристик систем активного управления очередями типа Random Early Detection (RED) // T-Comm: Телекоммуникации и транспорт. — 2011. — № 7. — С. 84—88.
177. *Зарядов И. С., Печинкин А. В.* Стационарные временные характеристики системы $GI/M/n/\infty$ с некоторыми вариантами дисциплины

- обобщённого обновления // Автоматика и телемеханика. — 2009. — № 12. — С. 161—174.
178. *Кокс Д., Смит В.* Теория очередей. — М. : Мир, 1966.
179. *Королькова А. В.* Метод расчёта вероятности сброса пакетов в алгоритме RED // Вестник Российского университета дружбы народов. Серия «Математика. Информатика. Физика.» — 2007. — № 12. — С. 32—37.
180. *Королькова А. В., Кулябов Д. С., Черноиванов А. И.* К вопросу о классификации алгоритмов RED // Вестник РУДН. Серия «Математика. Информатика. Физика.» — 2009. — № 3. — С. 34—46.
181. Математическая модель расчета и анализа характеристик систем с обобщенным обновлением и повторным обслуживанием / И. С. Зарядов [и др.] // Т-Сотт: Телекоммуникации и транспорт. — 2014. — № 6. — С. 16—20.
182. Модели для анализа качества обслуживания в сетях связи следующего поколения / Г. П. Башарин [и др.]. — Москва : Учебное пособие, электронное издание. Регистрационное свидетельство №15157 от 10.02.2009, номер гос. регистрации 0320802930, 2009.
183. *Наумов В. А.* Численные методы анализа марковских сетей. — М. : Изд-во УДН, 1985.
184. *Наумов В. А., Самуйлов К. Е., Яркина Н. В.* Теория телетрафика мультисервисных сетей. — Москва : РУДН, 2007.

185. *Печинкин А. В.* Марковская система обслуживания с конечным накопителем и отрицательными заявками, действующими на конец очереди // Информационные процессы. — 2007. — Т. 7. — С. 138—152.
186. *Печинкин А. В., Разумчик Р. В.* Система массового обслуживания с отрицательными заявками и бункером для вытесненных заявок в дискретном времени // Автоматика и телемеханика. — 2009. — № 12. — С. 109—120.
187. *Печинкин А. В., Соколов И. А., Чаплыгин В. В.* Многолинейная система массового обслуживания с конечным накопителем и ненадёжными приборами // Информатика и ее применения. — 2007. — Т. 1, № 1. — С. 27—39.
188. *Севастьянов Б. А.* Эргодическая теорема для марковских процессов и ее приложение к телефонным линиям с отказами // Теория вероятностей и ее прим. — 1957. — Т. 2, № 1. — С. 106—116.
189. Сети и телекоммуникации. Учебник и практикум / К. Е. Самуйлов [и др.]. — Москва : Юрайт, 2020.
190. *Хинчин А. Я.* Работы по математической теории массового обслуживания. — М. : Физматгиз, 1963.

Приложение А. Результаты имитационного моделирования в среде GPSS для однопороговых моделей управления входящим трафиком на основе полного обновления

А.1. Имитационная модель системы массового обслуживания $G/M/1/\infty$ с одним порогом, безопасной зоной и полным обновлением

```

PROB FUNCTION RN1,D2
0.01,0/1,1
Q_1 VARIABLE 40

GENERATE(Exponential(1,0,1/14))
LINK LIST1,LIFO,metka1
metka1 SEIZE Pribor других заявок)

ADVANCE (Exponential(1,0,1/16))
TEST LE CH$LIST1,V$Q_1,metka2
RELEASE Pribor
TRANSFER ,metka_end

metka2 TEST E FN$PROB,0,metka3
RELEASE Pribor
UNLINK LIST1,metka1,1
UNLINK LIST1,metka4,(CH$LIST1-V$Q_1)
metka4 TERMINATE 0

metka3 RELEASE Pribor
metka_end UNLINK LIST1,metka1,1
TERMINATE 0

```

```

GENERATE 100000
TERMINATE 1
START 1

```

А.2. Имитационная модель системы массового обслуживания $G/M/1/\infty$ с одним порогом, без безопасной зоны и полным обновлением

```

PROB FUNCTION RN1,D2
0.01,0/1,1
Q_1 VARIABLE 100 ;пороговое значение

GENERATE(Exponential(1,0,1/14)) ;генерируем заявки
LINK LIST1,FIFO,metka1
metka1 SEIZE Pribor заявок)15
ADVANCE (Exponential(1,0,1/16)) ;обслуживание
TEST LE CH$LIST1,V$Q_1,metka2
RELEASE Pribor
TRANSFER ,metka_end

metka2 TEST E FN$PROB,0,metka3
RELEASE Pribor ;освобождаем прибор
UNLINK LIST1,metka1,1
UNLINK LIST1,metka4,ALL
metka4 TERMINATE 0

metka3 RELEASE Pribor ;освобождаем прибор
metka_end UNLINK LIST1,metka1,1
TERMINATE 0

GENERATE 100000 ; Working day (minutes)
TERMINATE 1 ; Minus one minute
START 1 ; Start from the first minute

```

А.3. Сравнение результатов имитационного моделирования для однопороговых моделей с полным обновлением

Ниже представлены три таблицы с результатами моделирования для системы с обновлением и без безопасной зоны (sys1) и для системы с безопасной зоной и обновлением (sys2).

Таблица А.1.

Simulation results for different drop probabilities.

Drop probability		0.0025	0.005	0.01	0.025	0.05	0.1
Generated tasks	sys.1	999437	1000627	999286	1000116	999289	999928
	sys.2	1000148	1001042	1000211	1000028	999623	1000510
Serviced tasks	sys.1	995695	995719	992516	988573	986160	984624
	sys.2	999407	999823	998559	997262	996341	996638
Serviced tasks without calling the renv. mech.	sys.1	960378	966875	971329	974864	978438	979997
	sys.2	951429	956594	963167	967636	971884	977464
Dropped tasks	sys.1	3713	4901	6769	11542	13128	15303
	sys.2	738	1212	1650	2757	3279	3861
Probability of servicing tasks	sys.1	0.9963	0.9951	0.9932	0.9885	0.9869	0.9847
	sys.2	0.9993	0.9988	0.9983	0.9972	0.9967	0.9961
Probability of dropping tasks	sys.1	0.0037	0.0049	0.0068	0.0115	0.0131	0.0153
	sys.2	0.0007	0.0012	0.0016	0.0028	0.0033	0.0039
Average queue length	sys.1	8.127	7.934	7.487	7.088	6.755	6.506
	sys.2	8.802	8.697	8.351	8.113	8.067	7.982
Maximum queue length	sys.1	79	75	75	71	56	48
	sys.2	80	80	70	71	58	57
Average waiting time of a task in the queue	sys.1	0.898	0.876	0.83	0.789	0.753	0.726
	sys.2	0.967	0.955	0.92	0.895	0.89	0.879

В первой таблице представлены результаты моделирования для различных значений вероятности сброса q со следующими параметрами: пороговое

значение $Q_1 = 30$, интенсивность поступления заявок — 10 заявок в секунду, интенсивность обслуживания — 11 заявок в секунду. Моделирование длилось 100000 секунд.

Для системы с безопасной зоной значения вероятности обслуживания, числа обслуженных заявок, средней длины очереди и среднего времени ожидания начала обслуживания больше значений соответствующих характеристик для системы без безопасной зоны. При этом вероятность потери поступающей заявки для системы с безопасной зоной в разы меньше, чем для системы без безопасной зоны.

Таблица А.2.

Simulation results for different threshold values.

Threshold value		10	30	50	70	100
Generated tasks	sys.1	999875	999286	999775	999704	999350
	sys.2	997944	1000211	999710	999533	1000892
Serviced tasks	sys.1	965620	992516	998270	999465	999229
	sys.2	983879	998559	999370	999520	1000874
Serviced tasks without calling the renv. mech.	sys.1	738473	971329	994540	999359	999185
	sys.2	708912	963167	994122	999234	1000824
Dropped tasks	sys.1	34254	6769	1505	225	112
	sys.2	14058	1650	336	8	12
Probability of servicing tasks	sys.1	0.9657	0.9932	0.9985	0.9998	0.9999
	sys.2	0.9859	0.9983	0.9997	1.0000	1.0000
Probability of dropping tasks	sys.1	0.0343	0.0068	0.0015	0.0002	0.0001
	sys.2	0.0141	0.0016	0.0003	0.0000	0.0000
Average queue length	sys.1	5.623	7.487	8.668	8.768	9.166
	sys.2	6.418	8.351	8.914	9.11	9.467
Maximum queue length	sys.1	67	75	87	95	114
	sys.2	80	80	96	101	114
Average waiting time of a task in the queue	sys.1	0.64	0.83	0.954	0.965	1.008
	sys.2	0.72	1.002	0.92	0.981	1.036

Во второй таблице представлены результаты моделирования для различных пороговых значений Q_1 . Параметры моделирования: вероятность сброса $q = 0.01$, интенсивность поступление заявок — 10 заявок в секунду, интенсивность обслуживания — 11 заявок в секунду. Моделирование длилось 100000 секунд. Для системы с безопасной зоной значения вероятности обслуживания, числа обслуженных заявок, средней длины очереди и среднего времени ожидания начала обслуживания больше значений соответствующих характеристик для системы без безопасной зоны. При этом вероятность потери поступающей заявки для системы с безопасной зоной в разы меньше, чем для системы без безопасной зоны.

Таблица А.3.

Simulation results for different service intensities.

Service intensity		5 в 1с	8 в 1с	11 в 1с	15 в 1с
Genersted tasks	sys.1	999616	1001255	999286	999137
	sys.2	999540	997652	1000211	999137
Serviced tasks	sys.1	498806	787414	992516	999136
	sys.2	500950	801276	998559	999136
Serviced tasks without calling the renv. mech.	sys.1	113109	417466	971329	999129
	sys.2	4789	95898	963167	999129
Dropped tasks	sys.1	500645	213829	6769	0
	sys.2	498472	196344	1650	0
Probability of servicing tasks	sys.1	0.4990	0.7864	0.9932	1.0000
	sys.2	0.5012	0.8032	0.9983	1.0000
Probability of dropping tasks	sys.1	0.5008	0.2136	0.0068	0.0000
	sys.2	0.4987	0.1968	0.0016	0.0000
Average queue length	sys.1	106	35	7.487	1.34
	sys.2	131	54	8.351	1.34
Maximum queue length	sys.1	845	291	70	31
	sys.2	934	308	75	31
Average waiting time of a task in the queue	sys.1	10	3	0.83	0.201
	sys.2	13	5	0.92	0.201

Во третьей таблице представлены результаты моделирования для различных интенсивностей обслуживания заявок со следующими параметрами: пороговое значение $Q_1 = 30$, вероятность потери $q = 0.01$ интенсивность поступления заявок — 10 заявок в секунду. Моделирование длилось 100000 секунд. Для системы с безопасной зоной значения вероятности обслуживания, числа обслуженных заявок, средней длины очереди и среднего времени ожидания начала обслуживания больше значений соответствующих характеристик для системы без безопасной зоны. При этом вероятность потери поступающей заявки для системы с безопасной зоной в разы меньше, чем для системы без безопасной зоны.

По результатам моделирования можно сделать вывод, что при незначительном увеличении среднего времени ожидания начала обслуживания для систем с безопасной зоной резко снижается вероятность сброса принятой в систему заявки.

;

Приложение В. Результаты имитационного моделирования в среде GPSS

В.1. Имитационная модель в среде GPSS однопороговой системы массового обслуживания с обобщенным обновлением и безопасной зоной в накопителе

```

;Сегмент 1: Ввод данных
;
INITIAL X$ALAM,5
INITIAL X$AMU,6
INITIAL X$POROG,2
INITIAL X$ADEL,1
INITIAL X$KSERV,0
INITIAL X$KDEL,0
INITIAL X$PSERV,0
INITIAL X$PDEL,0
INITIAL X$WORK,0
TSERV TABLE MP5,0,0.02,50
TDEL TABLE MP5,0,0.02,50
;
; Сегмент 2: Генерация входящего потока и постановка заявок
; в очередь
;
GENERATE(exponential(1,0,(1/(X$ALAM))))
MARK 5
QUEUE SYST
QUEUE QUE
TEST G Q$SYST,1,NEXT1
LINK NAK,FIFO
;

```

```

;      Сегмент 3: Обслуживание заявок
;
NEXT1 SEIZE      PRIB
      DEPART     QUE
      TABULATE   TSERV
      ADVANCE(exponential(2,0,(1/(X$AMU))))
      RELEASE    PRIB
      DEPART     SYST
      TEST G     Q$SYST,0,NEXT3
      UNLINK     NAK,NEXT1,1
NEXT3 SAVEVALUE  KSERV+,1
      SAVEVALUE  PSERV,(X$KSERV/(X$KSERV+X$KDEL))

;      Сегмент 4: Моделирование сброса заявок
      TEST G     (Q$QUE-1),X$POROG,NEXT2
;
;      Сегмент 4: Моделирование сброса заявок
      SAVEVALUE  WORK,(poisson(3,X$ADEL))
      TEST G     X$WORK,(Q$QUE-X$POROG-1),NEXT4
      SAVEVALUE  WORK,(Q$QUE-1-X$POROG)
NEXT4 UNLINK     NAK,NEXT5,X$WORK
NEXT2 TERMINATE 1
NEXT5 DEPART     QUE
      DEPART     SYST
      TABULATE   TDEL
      SAVEVALUE  KDEL+,1

      SAVEVALUE  PDEL,(X$KDEL/(X$KSERV+X$KDEL))
;
      TERMINATE 1

```

В.2. Результаты имитационного моделирования в среде GPSS для модели с обобщенным обновлением и безопасной зоной

В первой таблице представлены результаты моделирования в среде GPSS для различного порогового значения Q_1 . При моделировании были приняты следующие параметры: $\mu = \frac{1}{6}$, $\lambda = \frac{1}{5}$ и через систему проходили 1000000 заявок. Вероятности обновления (сброса) $q(i)$ подчиняются пуассоновскому распределению. Из результатов моделирования видно, что с ростом по-

Таблица В.1.

GPSS simulation for differents threshold values

Threshold value	2	5	10	20	30	40
Probability of servicing tasks	0.89	0.948	0.982	0.997	0.99966	0.99998
Probability of dropping tasks	0.11	0.052	0.018	0.003	0.00034	0.00002
Serviced tasks	890208	948444	981840	997246	999660	999980
Dropped tasks	109792	51556	18160	2754	340	20
Average wait time to start service	0.238	0.386	0.579	0.756	0.795	0.768
Average time in the queue of dropped applications	0.507	0.903	1.631	3.172	4.649	5.788
Average queue length	1.335	2.059	2.988	3.807	3.975	3.834
Maximum queue length	23	24	31	35	42	47

рогового значения увеличивается вероятность обслуживания принятых в систему заявок и, соответственно, уменьшается вероятность потери. Среднее время ожидания начала обслуживания увеличивается, как и число

обслуженных заявок. Число сброшенных заявок с ростом порогового значения уменьшается, но при этом среднее время пребывания в накопителе сброшенной заявки увеличивается.

Во второй таблице представлены результаты моделирования в среде GPSS для различных интенсивностей обслуживания. При моделировании были приняты следующие параметры: $\lambda = \frac{1}{5}$, пороговое значение $Q_1 = 5$ и через систему проходило 1000000 заявок.

Таблица В.2.
GPSS simulation for differents service rates

Service rate	1/s	2/s	4/s	6/s	10/s	15/s
Probability of servicing tasks	0.499785	0.49979	0.769992	0.948444	0.996996	0.9998
Probability of dropping tasks	0.500215	0.50021	0.230008	0.051556	0.003004	0.0002
Serviced tasks	499785	499790	769992	948444	996996	999800
Dropped tasks	500215	500210	230008	51556	3004	200
Average wait time to start service	150001	24934	1.021	0.386	0.095	0.033
Average time in the queue of dropped applications	150001.8	24943	1.417	0.903	0.58	0.388
Average queue length	749677.6	124508	5.553	2.059	0.481	0.166
Maximum queue length	1501590	248965	59	24	14	11

Из результатов моделирования видно, что с ростом интенсивности обслуживания увеличивается вероятность обслуживания принятых в систему заявок и, соответственно, уменьшается вероятность потери. Среднее время ожидания начала обслуживания уменьшается, в то время как число обслуженных заявок увеличивается. Число сброшенных заявок с ростом интенсивности обслуживания уменьшается, как и среднее время пребывания в накопителе сброшенной заявки.

Список иллюстраций

2.1. Однопороговая система с безопасной зоной в накопителе . .	51
2.2. Однопороговая система без безопасной зоны в накопителе .	69
3.1. Система массового обслуживания $G/M/1/\infty$	81

Список таблиц

A.1. Simulation results for different drop probabilities.	142
A.2. Simulation results for different threshold values.	143
A.3. Simulation results for different service intensities.	144
B.1. GPSS simulation for different threshold values	148
B.2. GPSS simulation for different service rates	149