

Приложение к отчету о выполнении проекта № 16-19-00044
«Принципы распределения задач между сервисными роботами и средствами киберфизического интеллектуального пространства при многомодальном обслуживании пользователей» в 2018 году

Работы по проекту выполнялись согласно утвержденному плану на 2018 год, включая тестирование первой демонстрационной версии системы распределения задач по многомодальному обслуживанию пользователей и анализ полученных результатов, разработку второй демонстрационной версии распределения задач по многомодальному обслуживанию пользователей между мобильными роботами и другими обеспечивающими устройствами киберфизического пространства в выбранных сценариях с учетом выявленных недостатков первой версии, разработку алгоритмического и программного обеспечения управления движением антропоморфного робота Антареса. Кроме того, полученные результаты были апробированы на международных конференциях, подготовлены публикации по результатам исследований, а также оригинал-макет монографии, посвященной анализу проблем взаимодействия пользователя с киберфизическим пространством и решениям, предложенным в результате выполнения проекта, ведется сопровождение веб-страницы, посвященной проекту.

В результате выполнения проекта в 2018 году были рассмотрены вопросы обеспечения доступности сервисов корпоративного интеллектуального пространства. В качестве угроз доступности для такой системы рассматриваются намеренные и ненамеренные воздействия на интеллектуальное пространство. Были выделены следующие угрозы доступности: конфликтующие запросы, исчерпание ресурсов, сетевые атаки. Для противодействия этим угрозам были исследованы известные методы организации запросов, предложена методика оценки доступности сервисов, построена модель системы массового обслуживания и проведены эксперименты с различными стратегиями управления запросами и методами их фильтрации. Эксперименты показывают, что оценка эффективности такого управления имеет максимумы, зависящие от условий функционирования прикладной системы и параметров системы управления запросами. Таким образом, задача определения этих параметров может быть представлена как оптимизационная задача в многомерном пространстве.

Для реализации второй демонстрационной версии системы распределения задач по многомодальному обслуживанию пользователей была предложена модель оценки качества восприятия сервисов киберфизического интеллектуального пространства, которая позволяет на основе исторических данных об объективных параметрах качества обслуживания пользователей на этапе разработки киберфизического интеллектуального пространства анализировать ключевые показатели производительности системы и корректировать предъявляемые к ним требования. Рассмотрены модели сервисов видеоконференцсвязи, корпоративного телевидения, а также локализации и навигации, выделены общие и специфические показатели качества обслуживания сервисов.

При работе над проектом были учтены замечания экспертов, а именно: 1). В текущий отчет включены данные о сопровождении веб-страницы и разработке материалов, посвященных проблематике проекта и представляющих текущие научные результаты. Веб-страница проекта находится на сайте лаборатории автономных робототехнических систем http://robotics.nw.ru/servis/project_detail.php?ID=186 и освещает текущие научные результаты исполнителей.

2) Приведено описание топологии объектно-ориентированной функциональной модели, предлагаемой МИНОС, включая необходимые наборы структурных и процессных диаграмм.

3) Приведены данные об экспериментальной апробации программного обеспечения МИНОС, расширены и дополнены сведения о методике тестирования и критериях оценки выполнения сценариев многомодального взаимодействия пользователей. Кроме того,

расширены и дополнены сведения об исследовании конечно-элементной модели робота Антарес.

В приложении к отчету приведено расширенное описание результатов, включая дополнения в соответствии с замечаниями экспертов, графические материалы и ссылки на литературу. Далее кратко описаны результаты проведенных работ по проекту. Развернутое описание, включая графические материалы, формулы представлены в приложении к отчету.

1. Результаты тестирования первой демонстрационной версии системы распределения задач по многомодальному обслуживанию пользователей. Спецификации требований ко второй версии с учетом ошибок, выявленных в ходе тестирования

В 2018 году в рамках работы по проекту был проведен анализ результатов тестирования первой демонстрационной версии системы распределения задач по многомодальному обслуживанию пользователей, разработана спецификация требований ко второй версии с учетом ошибок, выявленных в ходе тестирования. Была разработана вторая демонстрационная версия системы распределения задач по многомодальному обслуживанию пользователей между мобильными роботами и другими обеспечивающими устройствами киберфизического пространства в выбранных сценариях.

Рассмотрим следующие ключевые задачи, решаемые киберфизическим интеллектуальным пространством (КФП) МИНОС: организация многомодального проактивного обслуживания пользователей; предоставление группы пользовательских сервисов, в том числе: сервисов локализации и навигации пользователей по территории, сервиса корпоративного телевидения, сервиса видеоконференцсвязи; управление сбором, хранением, агрегацией, обработкой и анализом данных о пользовательской активности; контроль за выполнением, приоритезацией, отслеживанием поставленных пользователями задач. В рамках каждой из них были выделены ключевые функциональные возможности.

В рамках организации многомодального проактивного обслуживания пользователей в окружающем киберфизическом интеллектуальном пространстве были выделены следующие ключевые функциональные возможности КФП:

- генерация маршрутов для перемещения пользователей по территории;
- локализация местоположения пользователей посредством компонентов КФС (видеокамеры, стационарные терминалы, мобильные робототехнические платформы (МРП), турникет);
- поэтапное сопровождение пользователей с помощью робототехнических компонент КФП (Антарес; МРП);
- формирование пользователями запросов на сопровождение третьих лиц;
- персонализированное информирование пользователей посредством сервиса корпоративного телевидения (КТ);
- формирование персонализированного контента на основе анализа данных о пользовательских предпочтениях;
- персонализация материалов, предоставляемых сервисом КТ на основе результатов идентификации пользователей посредством системного сервиса распознавания лиц;
- предоставление сервиса корпоративной видеоконференцсвязи (ВКС);
- внутрикорпоративный обмен мультимедийными данными на основе системы ВКС;
- контроль доступа к системе ВКС на основе результатов идентификации пользователей, в том числе с использованием системного сервиса распознавания лиц.

В рамках управления сбором, хранением, агрегацией, обработкой и анализом данных о пользовательской активности выделены следующие ключевые функциональные возможности:

- обеспечение возможности бесперебойного сбора данных с компонентов КФП;

- предоставление возможностей по записи, чтению и хранению данных, собранных различными компонентами МИНОС;
- обеспечение возможности обработки и анализа пользовательских данных, а также данных о функционировании компонентов системы и системы в целом с целью проведения аналитики и персонализации сервисов.

В рамках контроля за выполнением, приоритизацией и отслеживанием поставленных пользователями задач выделены следующие ключевые функциональные возможности:

- формирование списка задач для компонентов системы на основе алгоритма распределения задач между компонентами интеллектуального пространства;
- приоритизация задач, которые ставят пользователи перед системой;
- контроль процесса выполнения и результатов выполнения пользовательских запросов к системе.

Приведем модель данных КФП в нотации The Oracle Big Data Platform (Рис. 1). Модель управления информацией, представленная на рис. 1, иллюстрирует ключевые компоненты и потоки данных, а также выделяет различные формы сбора, обработки и анализа различных групп данных. Данная модель определяет основные потоки входных данных, которые в дальнейшем будут использоваться в аналитике, а также потоки наиболее ценных оперативных данных. В модели указываются: основные методы обработки и анализа данных, использование которых предполагается в системе, а также ключевые результаты анализа первичных данных. В рамках модели опосредованно приводятся ключевые подсистемы, сервисы и приложения, которые могут быть разработаны на основе результирующих данных. В заключение в рамках модели определяются основные преимущества и улучшения, которые могут быть достигнуты с использованием приложений и дополнительных подсистем, возникших в результате описанного в модели процесса обработки и анализа данных.

В рамках МИНОС к ключевым исходным данным относятся данные о взаимодействии пользователей с сервисами системы, данные о функционировании отдельных сервисов (журналы записей, события и пр.), а также данные о взаимодействии с компонентами КФП. На основе как приведенных на рис. 1, так и других данных с использованием средств интеллектуального анализа данных формируются: журнал системной активности, история пользовательской активности, а также уточняется информация в хранилище профилей пользователей. Полученные результаты используются: для построения паттернов поведения пользователей, аналитических отчетов относительно функционирования системы и пользовательской активности, кроме того, результаты анализа первичных данных необходимы для корректного функционирования подсистемы распределения и приоритизации задач, а также подсистемы управления хранилищами и подсистемы, осуществляющей управление системными ресурсами. Все вышеперечисленное позволяет обеспечить более высокое качество работы пользовательских сервисов МИНОС, оптимизировать управление системными ресурсами, а также обеспечить персонализацию взаимодействия пользователей с КФП.

Для описания топологии функциональной модели рассмотрим расширенную UML-диаграмму компонентов КФП (Рис.2).

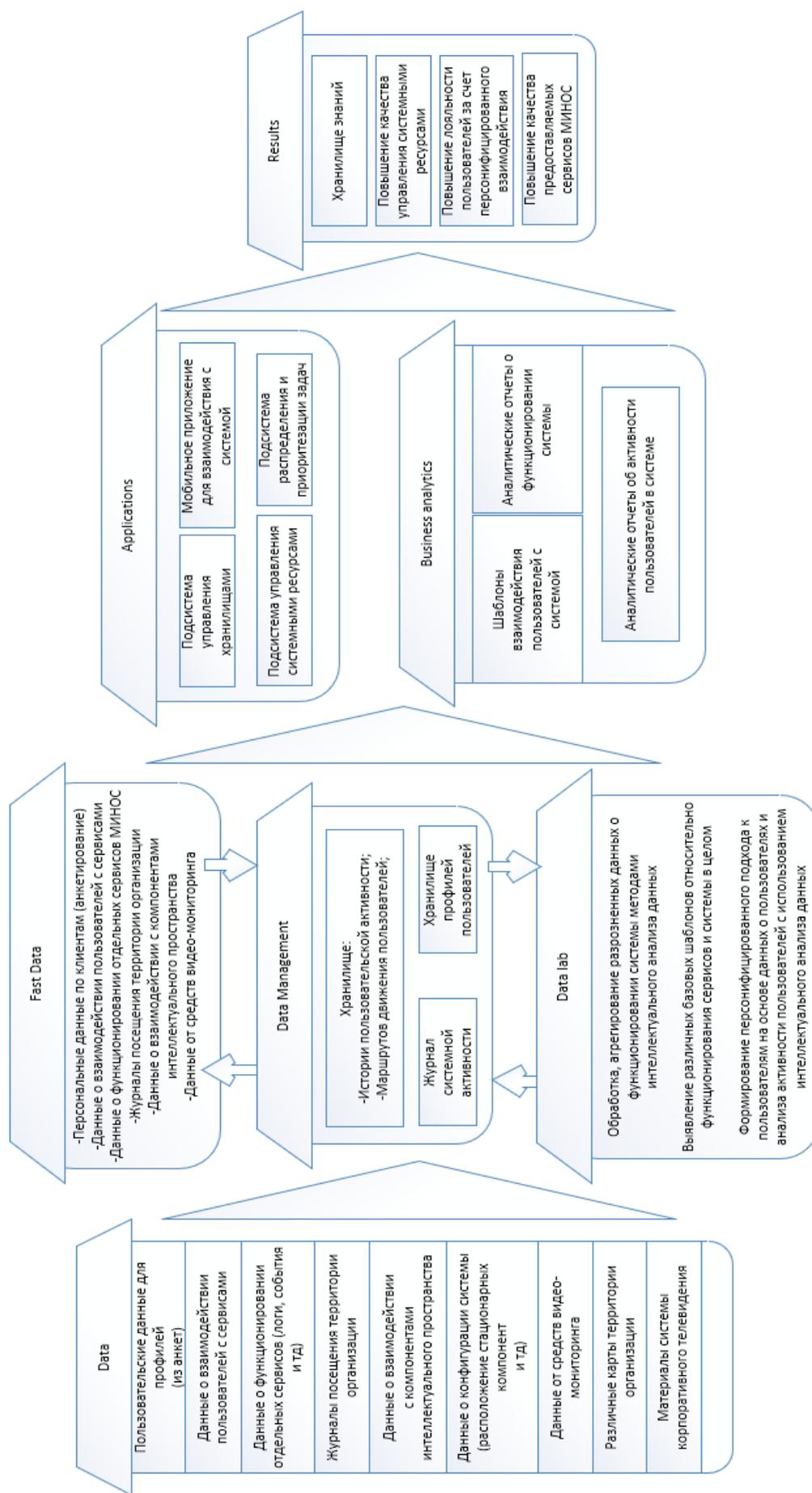


Рис. 1. Концептуальная модель унифицированного управления информацией системы на основе эталонной архитектуры управления информацией Oracle

Диаграмма компонентов описывает особенности физического представления системы и позволяет определить архитектуру разрабатываемой системы, установив зависимости между ее компонентами. В данном случае данная диаграмма была использована для построения логической архитектуры системы. Среди основных компонентов данной системы в рамках выбранной модели можно выделить следующие сервера и хранилища:

- центральный сервер управления – планировщик задач (отвечает за поступление задач, определяет порядок их приоритета, контролирует возможность доступа к разделяемым данным);
- хранилище профилей пользователей – статические редко изменяемые данные, требуется возможность доступа со стороны большинства компонентов (на чтение);
- хранилище результатов анализа пользовательской активности – содержит динамически дополняемые и переопределяемые данные, формируемые на основе анализа взаимодействия пользователей с различными компонентами МИНОС – пользовательские предпочтения, паттерны поведения пользователей;
- сервер по анализу пользовательской активности – проводит анализ и трансформирует данные об активности конкретных пользователей, формируемые компонентами системы, в пользовательские предпочтения и паттерны поведения;
- хранилища результатов работы по каждому из основных сервисов – содержат первичные данные о результатах функционирования сервисов (необработанные данные);
- аналитические хранилища результатов работы каждого из основных сервисов (аналитические результаты, агрегированные показатели, паттерны взаимодействия с сервисом и т.д.);
- сервер функционирования – обеспечивает непосредственное поддержание работы сервиса;
- сервер аналитики по сервису – анализирует результаты функционирования сервисов и помещает их в аналитические хранилища.

Отдельно следует выделить группу приложений для взаимодействия пользователей с МИНОС посредством различных устройств. Также необходимо отметить наличие административного сервиса для осуществления управления системой.

Рассмотрим событийно-процессную модель КФП (Рис.3). Данная диаграмма уровня «пользователь – приложение» отражает основные события, возникающие при взаимодействии пользователей с компонентами КФП, а также ключевые процессы, их инициирующие. Рассмотрим подробнее некоторые из них:

- Запрос на взаимодействие со стационарным терминалом МИНОС – Запрос на считывание QR-метки – процесс считывания метки мобильным приложением МИНОС с помощью пользовательского устройства – событие успешного считывания метки – процесс частичной передачи функций управления терминалом данному пользователю – (событие) о возможности взаимодействия с терминалом – процесс взаимодействия пользователя с функционалом терминала – запрос на управление показом слайдов сервиса КТ/запрос на подключение к сервису ВКС.
- Запрос на сопровождение посредством МРП – процесс автоматического определения местоположения пользователя – запрос на указание точки назначения – процесс определения пункта назначения пользователем – событие подтверждения корректности маршрута – процесс подачи МРП к местоположению пользователя – запрос на начало сопровождения – процесс сопровождения пользователя.
- Событие запуска пользователем мобильного приложения МИНОС – запрос на вход в мобильное приложение – процесс введения и подтверждения пользовательских данных для доступа к приложению – доступ к мобильному приложению разрешен – процесс взаимодействия пользователя с приложением – запрос на использование сервисов МИНОС – процесс взаимодействия пользователя с сервисами МИНОС.

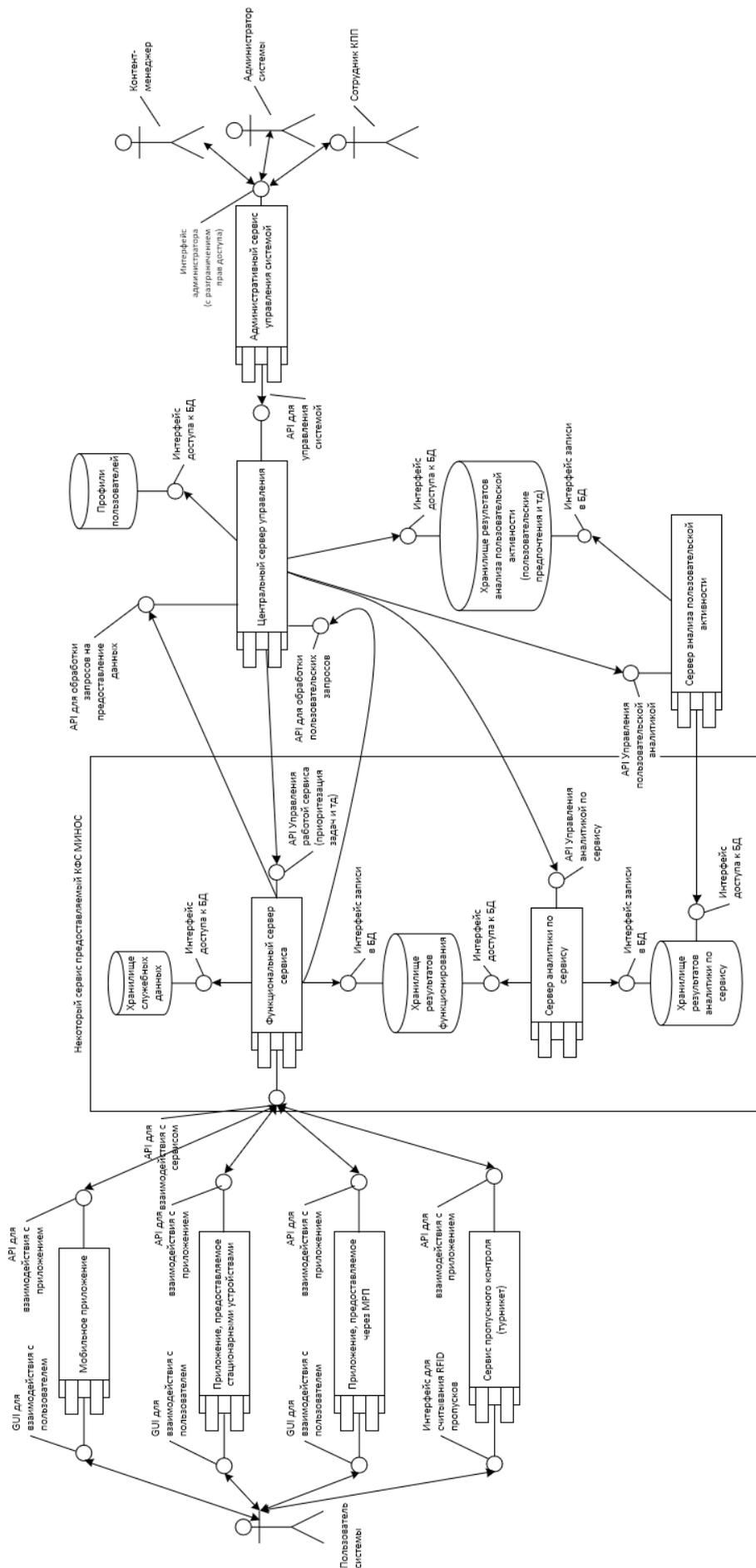


Рис. 2. Расширенная UML-диаграмма компонентов КФП

Для построения логической архитектуры данных была использована нотация Питера Чена (Рис. 4). С помощью данной модели типа «сущность – связь» были определены ключевые сущности и взаимосвязи между ними:

- пользователь (включает в себя такие группы пользователей, как сотрудник и гость);
- активность пользователя (отражает действия пользователя внутри КФП, данная сущность обладает наибольшим числом взаимосвязей);
- пользовательский запрос (данная сущность носит более строгий формальный характер с точки зрения описания атрибутов и более тесно связана с системными процессами внутри МИНОС);
- предпочтения пользователя (сущность вытекает из пользовательской активности, а потому имеет с ней тесную взаимосвязь);
- пользовательские сервисы МИНОС (данная сущность является абстракцией над реальными сервисами и нужна главным образом для обобщенного описания взаимосвязей пользовательских сервисов с прочими элементами);
- данные о функционировании сервисов (данная сущность является обобщенным отображением результатов работы сервисов, а потому тесно связана с предыдущей);
- системные события (данная сущность является частью предыдущей, но была отображена отдельно, поскольку является более формализованной, имеет фиксированный состав атрибутов, а также в значительно большей степени связана со внутрисистемными процессами);
- аналитика по результатам функционирования сервисов (сущность вытекает из сервисов МИНОС, а потому имеет с ней тесную взаимосвязь, отражает ключевые аналитические результаты функционирования системы в контексте сервисов);
- способ взаимодействия (данная сущность является абстракцией над конкретными вариантами пользовательского взаимодействия с КФП и нужна главным образом для обобщенного описания их взаимосвязей с прочими элементами).

Данное архитектурное описание системы содержит представление ее ключевых компонент в различных контекстах, что позволило сформировать унифицированное представление системы, и, таким образом, обеспечить взаимную интегрируемость ее компонентов и согласованность разделяемых в рамках системы данных. Представленная архитектура позволяет обеспечить соответствие функциональных возможностей системы выдвигаемым к ней требованиям, в частности, позволяет организовать многомодальное проактивное обслуживание пользователей в рамках КФП, персонализировать взаимодействие пользователей с предоставляемыми системой сервисами, а также достичь требуемого уровня эффективности управления системными ресурсами.

В рамках работ над второй демонстрационной версией распределения задач по многомодальному обслуживанию пользователей был рассмотрен более широкий круг вопросов, в частности, вопросы обеспечения качества предоставляемых сервисов и вопросы обеспечения доступности информации КФП. Были выявлены следующие типовые примеры угроз доступности информации в сервисах интеллектуального пространства:

1. Целенаправленные деструктивные воздействия на сервисы с помощью недостоверной или некорректной информации. В этом случае источником угрозы является пользователь, который может быть, как легитимным, так и нелегитимным. Типовым примером реализации такой угрозы является «отказ в обслуживании» (DoS).

2. Нецеленаправленные воздействия на сервис в условиях поступления от множества пользователей заявок, которые не могут быть выполнены сервисами в сроки, при которых актуальность заявок сохраняется. В этом случае источником угрозы является легитимный пользователь. Пример реализации угрозы – исчерпание пропускной способности канала передачи данных.

3. Ошибочное восприятие сервисами поступающих заявок. Эта угроза вероятна при использовании многомодальных средств человеко-машинного взаимодействия – при взаимодействии с сервисами с помощью речи, жестов, при распознавании образов на видео. Угроза может также осуществляться из-за ошибок в программном обеспечении. Источник угрозы – программное обеспечение. Обработка ошибочно воспринятых данных негативно сказывается на доступности сервисов для легитимных пользователей.

При распределении задач многомодального обслуживания пользователей доступность сервисов в различных условиях может оцениваться с помощью известных методов оценки качества обслуживания. Например, среднее время выполнения заявки может рассматриваться в качестве параметра качества обслуживания (QoS) как математическое ожидание интервала времени между отправкой запроса о выделении ресурса и его фактическим выделением. Эту величину можно оценить, как среднее арифметическое значение измерений, составляющих представительную выборку.

Пользователи могут отдавать команды сервисам киберфизического пространства используя различные модальности, например, речь и жесты, они также могут управлять сервисом с мобильного устройства или с персональных компьютеров (для администраторов). При разработке системы распределения задач по многомодальному обслуживанию пользователей необходимо найти оптимальную стратегию управления поступающими заявками, которая обеспечивала бы максимальное качество обслуживания. При этом были выявлены следующие ключевые требования, от оптимизации которых зависит качество обслуживания пользователей: интенсивность поступления заявок; их ресурсоёмкость; количество пользователей. Кроме того, все заявки характеризуется параметрами, определяющими пользователя, сессию, модальность и запрашиваемое действие, а также могут быть классифицированы следующим образом:

1. по распределённости – локальные (требуют обработки только на одном канале обслуживания), распределённые (требуют обработки на нескольких каналах), глобальные (требуют обработки на всех каналах);

2. по приоритету – критические (с немедленным вытеснением всех других заявок), регулярные (имеющие относительный приоритет в дисциплине обслуживания), фоновые (выполняются только при отсутствии других заявок, немедленно вытесняются любыми нефоновыми заявками);

3. по процедуре выполнения – непрерывные и вытесняемые;

4. по длительности – с фиксированным и изменяемым временем их выполнения;

5. по связи с другими заявками – одиночные, сессионные.

Более детально параметры заявки можно описать в формате JSON (Листинг 1).

```

request: {
  user: {
    id,
    priority
  },
  session: {
    id,
    start_timestamp
  },
  modality,
  command: {
    id,
    options,
    priority,
    resumable,
    resources: [ {
      id,
      requiredAmount,
      requiredTime
    } ]
  }
}

```

От этих параметров зависит приоритет заявки. Таким образом, для вычисления QoS предоставляемых пользователю услуг нужно определить: исходные данные, методику определения приоритетов, стратегию управления заявками. Условия возникновения заявок строго не определены и могут меняться в зависимости от контекста. Однако, при использовании центральной предельной теоремы теории вероятностей для потоков событий процесс поступления одиночных заявок можно описать с применением распределения Пуассона и использовать параметр распределения λ для характеристики интенсивности поступления заявок.

При построении методики определения приоритетов задач многомодального обслуживания пользователей необходимо использовать следующие принципы. Во-первых, необходимо принимать во внимание вероятность точного восприятия и распознавания команд в зависимости от модальности. Во-вторых, необходимо учитывать достоверность и легитимность воспринимаемых сигналов. В-третьих, при расстановке приоритетов заявок следует учитывать возможность пользователя воспринимать ответ сервиса. В-четвёртых, необходимо учитывать свойства самой задачи, в частности, срочность, которая определяется типом задачи автоматически (на основе приоритета пользователя или процесса) или администратором вручную, способность задачи к вытеснению с последующим возобновлением. При поступлении заявки возможны следующие способы её обработки: немедленное исполнение (с возможным вытеснением активной заявки); отклонение заявки; постановка заявки в очередь. В последнем случае могут быть использованы дисциплины обслуживания FIFO или LIFO.

Была проведена симуляция процесса обслуживания заявок в модели второй демонстрационной версии системы распределения задач по многомодальному обслуживанию пользователей. Экспериментальные исследования предложенной модели состояли в симуляции потока входящих заявок и процесса их обработки. Результат симуляции при таком подходе отражает эффективность управляющей стратегии обработки заявок. В рамках рассматриваемой модели не учитываются фоновые задачи, так как их выполнение никак не влияет на задачи более высокого приоритета. Критические задачи также не рассматриваются в силу их нерегулярности и малой вероятности появления. В общем виде процесс обработки новой заявки сводится к следующим шагам:

1. если в настоящий момент времени для обработки заявки имеется достаточно свободного ресурса, то заявка обрабатывается немедленно;

2. если в настоящий момент времени ресурс занят заявкой того же пользователя, то её обработка отменяется, а новая заявка обрабатывается немедленно;

3. в иных случаях используется одна из стратегий, рассмотренных далее.

К стратегиям без приоритета относятся:

- система с отказами, согласно которой обработка заявки отменяется, если ресурс занят;
- система с ограниченным временем ожидания, когда новая заявка обрабатывается после предыдущих, если не прошёл некоторый, заранее заданный период времени.

К стратегиям с приоритетом относятся:

- сдвиг задач с более низким приоритетом;
- вытеснение задач с более низким приоритетом;
- отмена задач с более низким приоритетом.

Для оценки качества процесса обработки заявок можно измерять следующие величины:

- относительная доля выполненных заявок (относительная пропускная способность);
- доля выполненных заявок, взвешенных по их приоритету;
- среднее время ожидания при выполнении заявок;
- время загрузки канала.

Для экспериментов использовалась модель системы, написанная на языке Python (рис. 5)

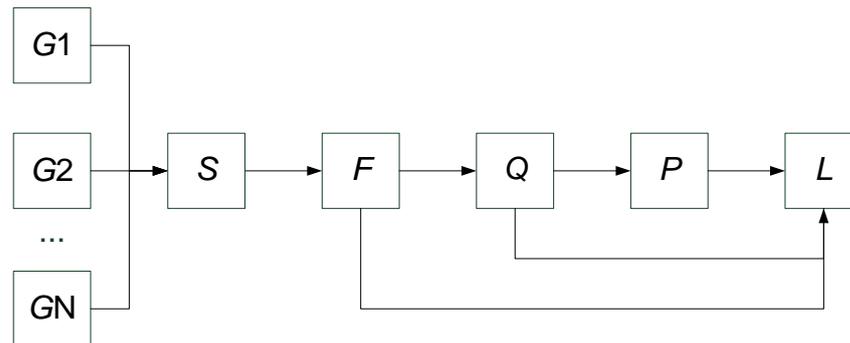


Рис. 5. Модель системы массового обслуживания

Представленная на рис. 5 модель состоит из следующих блоков: генераторы потоков ($G1 \dots GN$); диспетчер сессий (S); фильтр (F); контроллер очереди (Q); обработчик запросов (P); логгер (L). Генераторы потоков создают объекты класса «Поток», которые моделируют пользовательские сессии. Они характеризуются паттернами трафика и его распределениями. Кроме того, для каждого потока задаётся его длительность. Диспетчер сессий сохраняет объекты потоков и генерирует запросы для каждого активного потока в соответствии с его параметрами. Фильтр реализует меры защиты, которые зависят от конфигурации системы защиты. Контроллер очереди передаёт запросы обработчику, используя данные об очередности, приоритете запросов и о стратегиях управления. Логгер принимает сведения обо всех запросах и их статусе по окончании их обработки и сохраняет эти сведения в базу данных.

Рассмотрим для начала модель, в которой не используется фильтрация (т.е. блок F на рис. 5 передаёт все заявки в блок Q). В модели использованы следующие исходные данные: N типов одиночных задач от разных пользователей, появляющихся с интенсивностями λ_i и собственными приоритетами P_i , определёнными исходя из используемых модальностей. В качестве задач рассматриваются запросы пользователей на выдачу информации. Выполнено моделирование для стратегий, описанных выше. При моделировании учитывались три приоритета, связанные с используемыми модальностями. При

поступлении новой заявки в зависимости от приоритетов конфликтующей заявки выполняется одно из действий A_H (приоритет новой заявки выше), A_E (приоритеты равны), A_L (приоритет ниже). Если $A_H=A_L=A_E=A$, применяемая стратегия не учитывает приоритет.

При моделировании результат выполнения каждой задачи характеризуется степенью успешности её решения и задержкой (временем ожидания). Стратегии, при которых преобладает ожидание, порождают более высокую задержку и одновременно большую степень успешности её реализации. Для стратегий с преобладанием отказов результаты противоположные. Для комплексной оценки стратегий можно составить показатель эффективности выполнения задачи i , зависящий от обеих величин. Так, в случае отказа можно принять величину задержки $D_i = \infty$, а для перехода к конечным величинам использовать оценку (1):

$$E_i = \frac{D_0}{D_0 + D_i}, (1)$$

где D_0 – величина задержки, при которой показатель эффективности уменьшается вдвое. При рассмотрении стратегий, при которых задача может быть прервана, необходимо также учитывать фактическое время её выполнения. В этом случае показатель E_i можно обобщить в виде (2):

$$E_i = \frac{T_{Fi}D_0}{T_{Pi}(D_0 + D_i)}, (2)$$

где T_{Fi} – фактическое время выполнения, T_{Pi} – планируемое. Для получения интегрированной оценки той или иной стратегии с учётом приоритетов задач можно использовать показатель (3)

$$E = \frac{\sum_{i=1}^N P_i E_i}{\sum_{i=1}^N P_i}, (3)$$

где P_i – приоритет i -й задачи, N – количество задач.

Чтобы учесть легитимность запросов, можно ввести логическую оценку $C_i \in \{0;1\}$, где 0 означает отсутствие легитимности, а 1 характеризует доверенный запрос. Тогда представленная выше формула может быть обобщена следующим образом (4):

$$E = \frac{\sum_{i=1}^N P_i C_i E_i}{\sum_{i=1}^N P_i}. (4)$$

В более общем случае может применяться нечёткая оценка легитимности $C_i \in [0;1]$.

При низкой интенсивности поступления заявки не конфликтуют, значение эффекта E близко к 1 и почти не зависит от стратегии обработки заявок. При увеличении интенсивности стратегии показывают разный эффект. На рис. 6 показаны зависимости эффекта от интенсивности при стратегиях без приоритета.

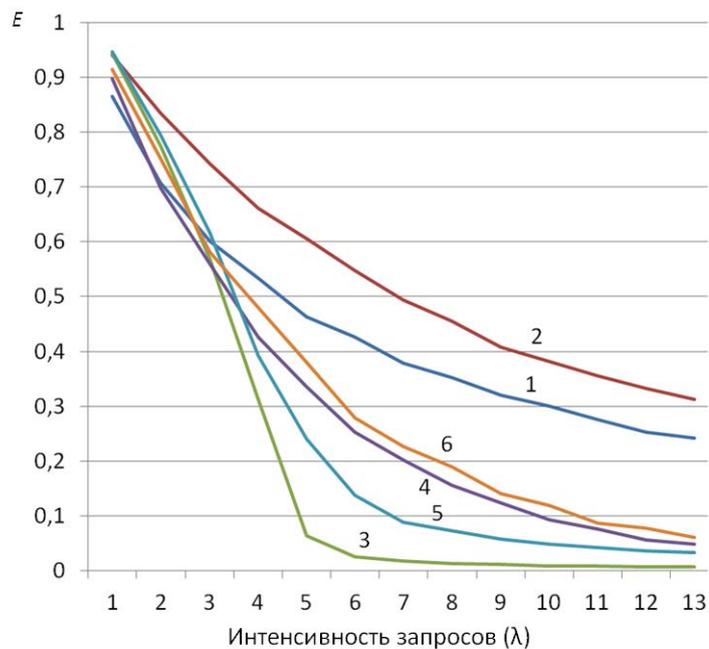


Рис. 6. Зависимость эффекта от интенсивности при различных стратегиях обработки заявок: 1 – отклонение новой заявки, 2 – отклонение конкурирующих заявок, 3 – FIFO, 4 – LIFO, 5 – FIFO (максимальная задержка 1), 6 – LIFO (максимальная задержка 1)

График показывает, что без учёта приоритетов стратегии с отказами более эффективны, чем стратегии с очередями в случае, если приоритет заявок не учитывается. Это, в частности, объясняется чувствительностью системы к задержкам. При экспериментах использовалось значение $D_0 = \bar{T}_p$, что с точки зрения человеко-машинного взаимодействия означает довольно низкую чувствительность. Причина в том, что время реакции интерфейса, при котором система воспринимается пользователем как управляемая, не превышает нескольких секунд. Очевидно, что более жёсткие условия при задании величины D_0 только уменьшают значение эффекта для этих стратегий.

Заметно также, что в стратегиях с очередями ограничение длины очереди положительно влияет на величину эффекта. Подобное ограничение приводит к отмене задач с большой величиной задержки, которые занимают ресурс, но не вносят существенного вклада в величину эффекта. Тем не менее, такие стратегии всё равно уступают стратегиям с отказами, которые являются их частным случаем при нулевой длине очереди.

Теперь рассмотрим стратегии с учётом приоритета и сравним их с беспriorитетными стратегиями с отказами. Рассматривалось множество стратегий, у которых среди действий A_H, A_L, A_E присутствуют хотя бы два различных, и при этом заявки с более высоким приоритетом не могут получать менее приоритетное обслуживание.

Графики для выявленных наиболее эффективных стратегий приведены на рис. 7.

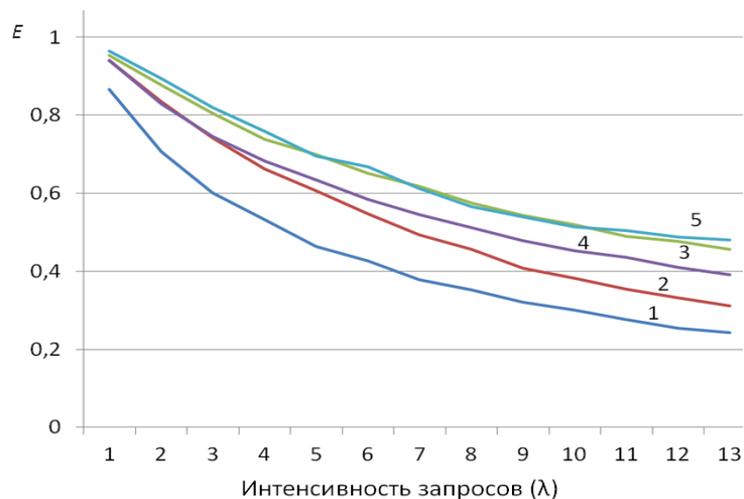


Рис. 7. Сравнение беспriorитетных (1, 2) и приоритетных (3, 4, 5) стратегий

На рис. 7 обозначены эффекты для стратегий:

1: отклонение новой заявки (A – отказ);

2: отклонение конкурирующих заявок (A – вытеснение);

3: отклонение конкурирующих заявок при более высоком или равном приоритете новой заявки (A_L – отказ, A_E, A_H – вытеснение);

4: отклонение конкурирующих заявок только при более высоком приоритете новой заявки (A_L, A_E – отказ, A_H – вытеснение);

5: очередь FIFO только при более низком приоритете новой заявки, иначе отклонение конкурирующих заявок (A_L – FIFO, A_E, A_H – вытеснение).

Для большинства стратегий моделирование показывает, что постановка заявки в очередь приводит к меньшему значению эффекта. Однако использование очереди FIFO при обработке задач с низким приоритетом приводит к небольшому повышению эффекта. Этот факт можно объяснить тем, что использование стратегий с отказами в этом случае приведёт к отмене низкоприоритетных заданий и их эффекту $E_i = 0$, а при использовании очереди эти задания имеют шанс на выполнение с задержкой при условии отсутствия заданий с более высоким приоритетом ($E_i \geq 0$).

Эксперименты с другими приоритетными стратегиями показывают более низкую эффективность, что показано на рис. 8.

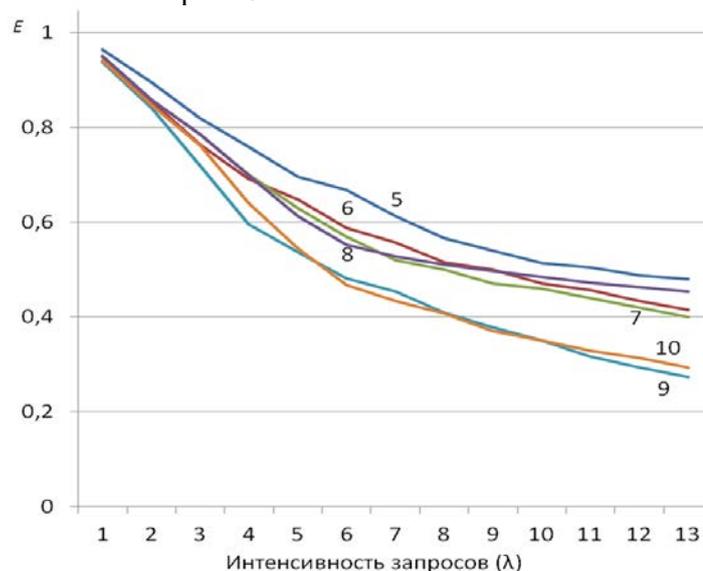


Рис. 8. Эффективность приоритетных стратегий

На рис. 8 обозначены стратегии:

б: A_L – отказ, A_E – вытеснение, A_H – LIFO;

- 7: A_L, A_E – вытеснение, A_H – LIFO;
- 8: A_L – FIFO, A_E – вытеснение, A_H – LIFO;
- 9: A_L – FIFO, A_E, A_H – LIFO;
- 10: A_L – FIFO, A_E – LIFO, A_H – вытеснение.

Таким образом, в рассматриваемых условиях можно считать оптимальной стратегию 5, предусматривающую использование дисциплины FIFO, если новая заявка имеет более низкий приоритет, и отклонение конкурирующих заявок в остальных случаях.

Теперь рассмотрим ситуацию, когда в системе обработки заявок в совокупности с определённой ранее стратегией 5 используется модуль фильтрации. При возрастании интенсивности запросов при наличии угрозы и отсутствии фильтрации часть запросов будет отклоняться при реализации управляющей стратегии. В частности, при наличии угроз доступности типа DDoS-атак легитимные запросы могут быть отклонены, а нелегитимные – выполнены, причём из-за большого количества нелегитимных запросов при DDoS-атаке последние будут чаще передаваться на исполнение. Следовательно, оценка эффекта будет уменьшаться. На рис. 9 представлена зависимость значения эффекта от средней интенсивности возникновения запросов λ_R и потоков λ_S при условии, что потоки легитимны:

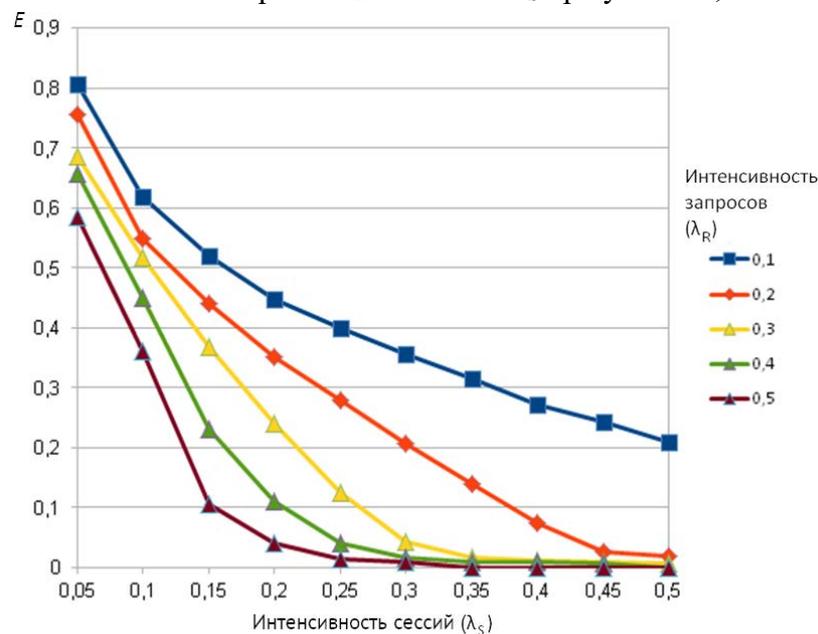


Рис. 9. Зависимость эффекта от интенсивностей λ_R и λ_S

Значение эффекта также значительно снижается при появлении паттернов нелегитимных запросов с интенсивностями λ_M (рис. 10):

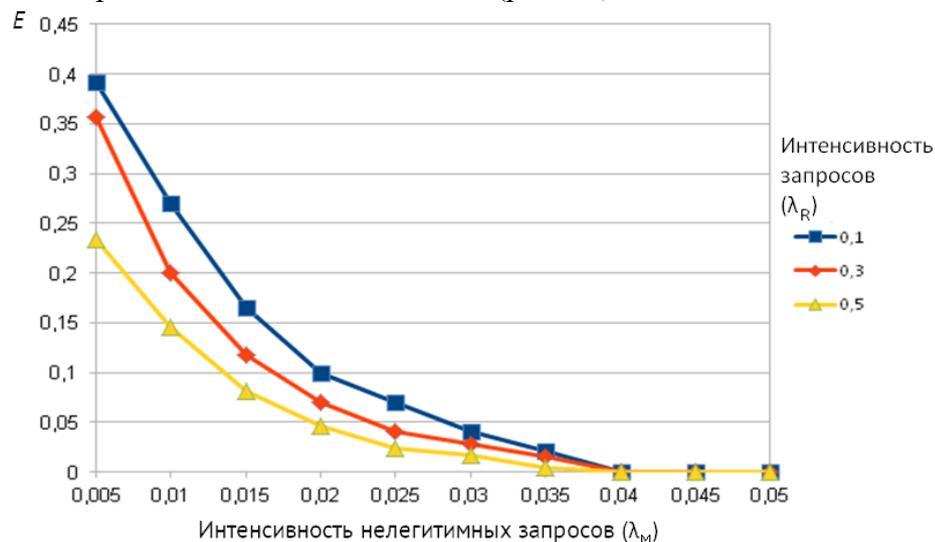


Рис. 10. Зависимость эффекта от λ_R и λ_M ($\lambda_S = 0,1$)

Фильтрация позволяет частично устранить нелегитимные запросы и увеличить значения эффекта. Было проведено моделирование для оценки значений эффекта с использованием фильтрации по разным пороговым значениям статистических моментов и энтропии. Рис. 11 иллюстрирует изменение эффекта при изменении порога оценки математического ожидания для частоты появления запросов. Рис. 12 характеризует ту же величину в зависимости от порога среднеквадратического отклонения. Рис. 13 отражает зависимость эффекта от допустимого нижнего порога энтропии.

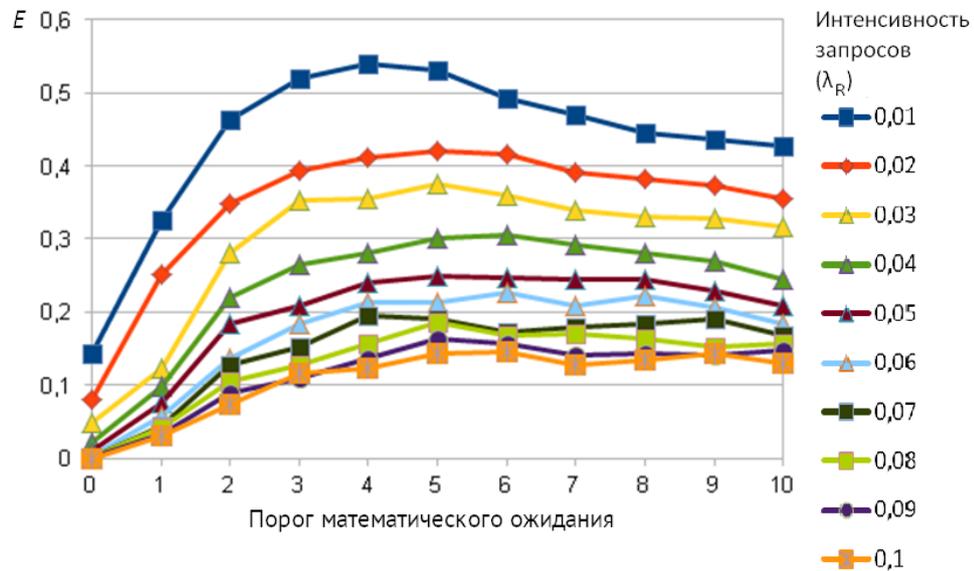


Рис. 11. Зависимость эффекта от интенсивности запросов и допустимого порога среднего значения

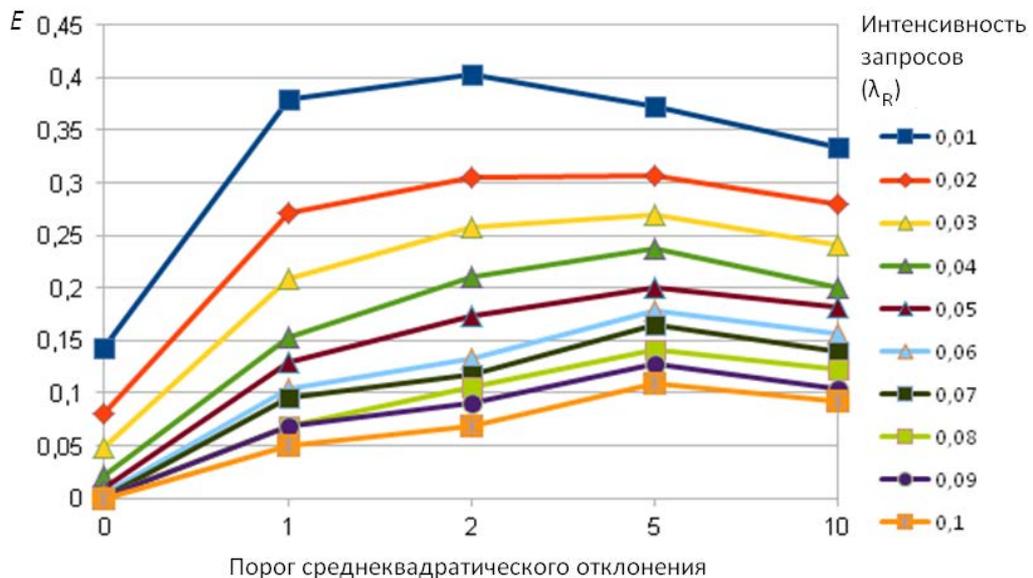


Рис. 12. Зависимость эффекта от интенсивности запросов и допустимого порога среднеквадратического отклонения

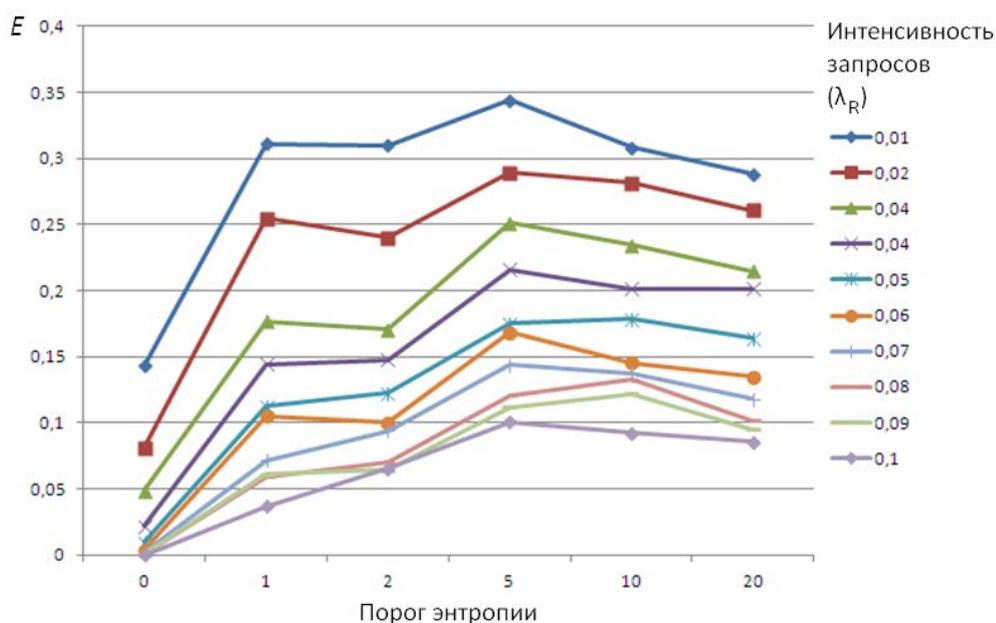


Рис. 13. Зависимость эффекта от интенсивности запросов и допустимого порога энтропии

Было выявлено, что значение эффекта имеет максимумы, которые зависят как от выбранных пороговых значений, так и от условий функционирования (интенсивностей запросов). Слишком низкие значения порогов приводят к множеству ошибок первого рода, высокие – к ложным срабатываниям. Таким образом, задача определения параметров для методов фильтрации может быть представлена как оптимизационная задача в многомерном пространстве, где измерения соответствуют параметрам применяемых методов фильтрации. Изменение условий функционирования системы может быть учтено с помощью адаптивного подхода к управлению методами фильтрации.

2. Вторая демонстрационная версия распределения задач по многомодальному обслуживанию пользователей между мобильными роботами и другими обеспечивающими устройствами киберфизического пространства в выбранных сценариях

Для второй демонстрационной версии распределения задач по многомодальному обслуживанию пользователей была создана модель оценки качества восприятия пользователем сервисов КФП. На примере МИНОС были выделены основные параметры QoS сервисов КФП и смоделирована комплексная оценка качества обслуживания пользователей (QoE) (рис. 14). Рассматривались следующие сервисы: видеоконференцсвязь (ВКС), корпоративное телевидение (КТ), локализация и навигация (ЛН).

При разработке методики оценивания многомодального взаимодействия пользователей с обеспечивающими устройствами окружающего КФП можно выделить показатели качества обслуживания (QoS), уникальные для каждого сервиса, а также ряд показателей, общих для всех сервисов. В частности, каждая пользовательская заявка требует обслуживания (выполнения), которое предполагает загрузку определенных аппаратных и программных ресурсов (дисплей, звук, пользовательская сессия) в течение некоторого времени.

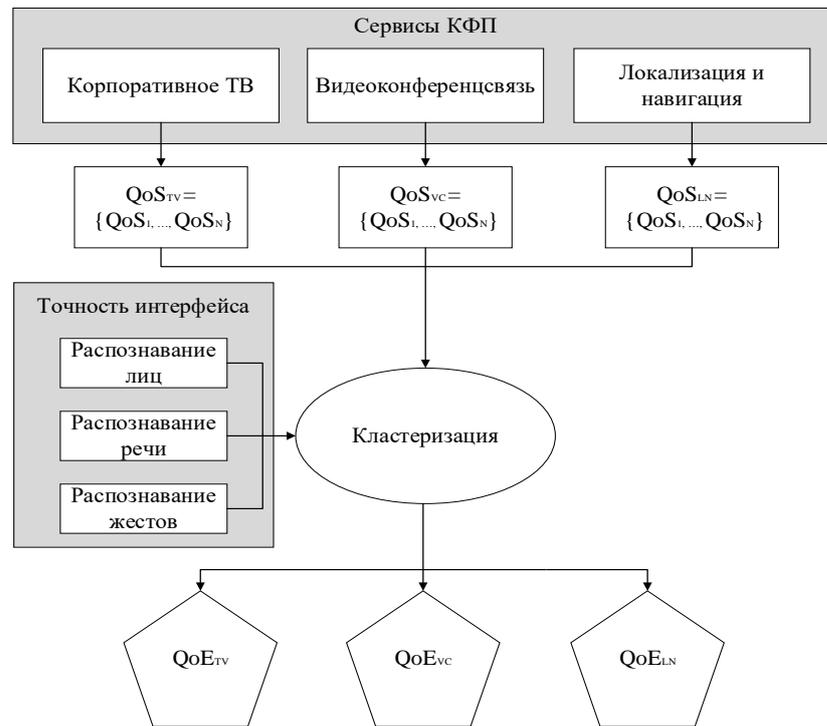


Рис. 14. Моделирование показателей качества обслуживания пользователей КФП

2.1. Общие параметры QoS

Основные параметры QoS, общие для всех модулей КФП, связаны со временем загрузки и инициализации приложений, временем обработки заявок при различном уровне нагрузки на систему (табл. 1). Время обработки запроса является ключевым фактором, определяющим качество обслуживания в информационных сервисах.

Таблица 1. Общие параметры QoS сервисов КФП

| Показатель | Обозначение | Возможные значения | Допустимые значения |
|-------------------------------------------------------------------------------------------------------|------------------------|--------------------|---------------------|
| Время загрузки приложения, с | t_l^{App} | $[0; \infty)$ | $[0; t_{UI}]$ |
| Время инициализации приложения, с | t_i^{App} | $[0; \infty)$ | $[0; t_{UI}]$ |
| Время актуализации данных в приложении, с | t_a^{App} | $[0; \infty)$ | $[0; t_{UI}]$ |
| Относительная доля выполненных заявок | $f_{np}^{Success}$ | $[0; 1]$ | $[1-p_D; 1]$ |
| Относительная доля выполненных заявок, взвешенная по приоритету | $f_p^{Success}$ | $[0; 1]$ | $[1-p_D; 1]$ |
| Относительная доля отказов в выполнении пользовательских заданий при отсутствии конкурирующих заданий | f_{normal}^{Denial} | $[0; 1]$ | $[0; p_D]$ |
| Относительная доля отказов в выполнении пользовательских заданий при наличии конкурирующих заданий | f_{stress}^{Denial} | $[0; 1]$ | $[0; p_D]$ |
| Время задержки в выполнении пользовательских заданий при отсутствии конкурирующих заданий, с | $t_{normal}^{Request}$ | $[0; \infty)$ | $[0; t_{UI}]$ |
| Время задержки в выполнении пользовательских заданий при наличии конкурирующих заданий, с | $t_{stress}^{Request}$ | $[0; \infty)$ | $[0; t_{UI}]$ |

В таблице используются следующие обозначения:

t_{UI} – максимальное время реакции пользовательского интерфейса, которое пользователь считает комфортным;

p_D – максимально допустимая вероятность отказа в выполнении пользовательской задачи.

Помимо вышеперечисленных параметров, на качество восприятия сервисов КФП существенно влияет модальность интерфейса, посредством которого пользователь взаимодействует с системой. По этой причине при моделировании системы оценивания качества обслуживания пользователей необходимо учитывать: точность используемых алгоритмов распознавания речи, точность используемых алгоритмов распознавания лиц, удобство графического интерфейса. Рассмотрим подробнее параметры QoS, специфические для каждого сервиса.

2.2. Сервис видеоконференцсвязи

Сервис ВКС является наиболее чувствительным к воздействию множества различных факторов и, вместе с тем, к нему предъявляются наиболее высокие требования, поскольку он должен обеспечить передачу аудио- и видеопотока приемлемого качества в реальном времени между разнородными пользовательскими устройствами. Помимо рассмотренных выше общих показателей выделяют следующие параметры QoS для сервиса видеоконференцсвязи: коэффициент потери пакетов (Packet Loss Rate, PLR), определяется как отношение количества потерянных пакетов к общему числу пакетов; джиттер (Jitter), искажение периодического характера потока пакетов между источником и пунктом назначения; пропускная способность (Bandwidth); время отклика пирингового соединения; потребление ресурсов центрального процессора и оперативной памяти клиентской и серверной частей приложения. Для моделирования качества обслуживания сервиса ВКС ограничимся рассмотрением следующих параметров (табл. 2), имеющих наибольшее влияние на качество обслуживания.

Таблица 2. Параметры QoS сервиса ВКС

| Показатель | Обозначение | Возможные значения | Допустимые значения |
|-------------------------------------------------------------------------|--------------------|--------------------|---------------------|
| Время отклика пирингового соединения, мс | t_{resp} | $[0; \infty)$ | $[0; t_{acc}]$ |
| Потребление ресурсов процессора серверной части приложения, % | r_{CPU}^{Server} | $[0; 100]$ | $[0; l_{CPU}]$ |
| Потребление ресурсов оперативной памяти серверной части приложения, МБ | r_{RAM}^{Server} | $[0; C_{RAM}]$ | $[0; l_{RAM}]$ |
| Потребление ресурсов процессора клиентской части приложения, % | r_{CPU}^{Client} | $[0; 100]$ | $[0; l_{CPU}]$ |
| Потребление ресурсов оперативной памяти клиентской части приложения, МБ | r_{RAM}^{Client} | $[0; C_{RAM}]$ | $[0; l_{RAM}]$ |
| Коэффициент потери пакетов (PLR), % | f_{PLR} | $[0; 100]$ | $[0; k_{PLR}]$ |

В таблице используются следующие обозначения:

t_{acc} – максимальное допустимое время отклика пирингового соединения, которое пользователь считает комфортным;

l_{CPU}, l_{RAM} – максимально допустимые значения нагрузки на центральный процессор и оперативную память соответственно; определяются исходя из специфики серверного оборудования и клиентского устройства;

k_{PLR} – максимально допустимый коэффициент потери пакетов.

Кроссплатформенные веб-технологии позволяют адаптировать работу приложения ВКС под различные программно-аппаратные платформы и обеспечивать контроль над обычными и управляемыми аккаунтами в единообразной адаптируемой среде в режиме реального времени. При этом, различия в конечном продукте относительно каждой из платформ носят условный характер – логика работы веб-приложения отделена от деталей

реализации аппаратной платформы, на которой оно запущено. Тем не менее, приложение ВКС обладает зависимостью от веб-браузеров. Кроме того, значение имеют и аппаратные возможности платформы, на которой оно запущено.

При разработке модуля ВКС были учтены эти особенности функционирования веб-приложений, чтобы оптимизировать обработку мультимедийных данных соответствующим образом. Как правило, средства ВКС ориентированы на захват клиентским устройством аудиовидеоданных только одного пользователя, и чаще всего, сеанс связи организован между двумя устройствами. Однако при проведении сеансов связи между несколькими пользователями такой подход зачастую недостаточен, и требуется многоканальная обработка аудиовизуальных сигналов на стороне каждого клиента.

В основе разработанного веб-приложения ВКС использованы решения, направленные на сокращение объема передаваемых внутри сети мультимедийных данных. Для этого, в архитектуру приложения был внедрён механизм соединения участников ВКС по схеме звезда (рис. 15). Это позволило добиться сокращения нагрузки на сервер приложения и уменьшения трафика.

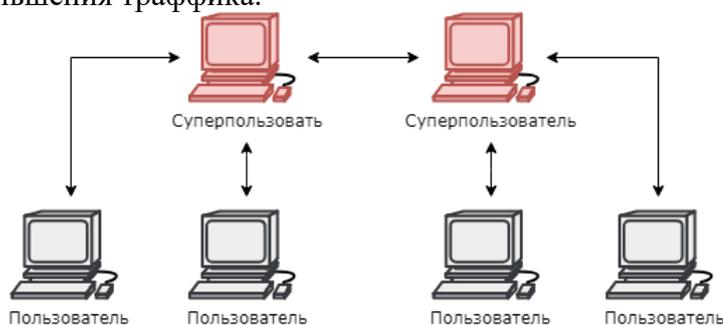


Рис. 15. Организация сети для обмена мультимедийными данными между участниками видеоконференцсвязи

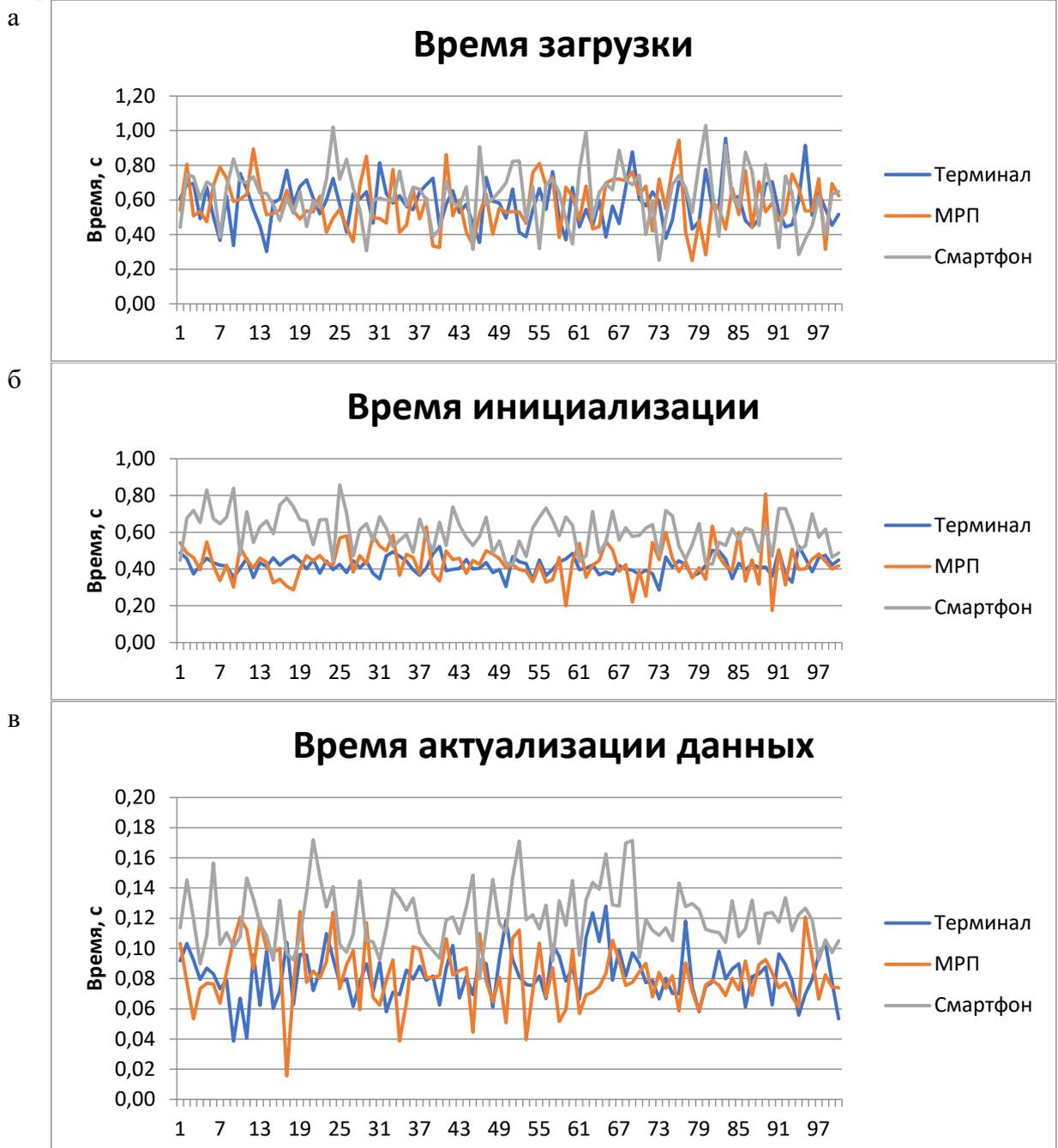
С учетом замечаний рецензента во второй версии системы была проведена более детальная апробация разработанных сервисов с учетом различных критериев. В табл. 3 приведены экспериментальные значения QoS сервиса ВКС, как общие, так и специфические.

Таблица 3. Общие параметры QoS сервисов КФП

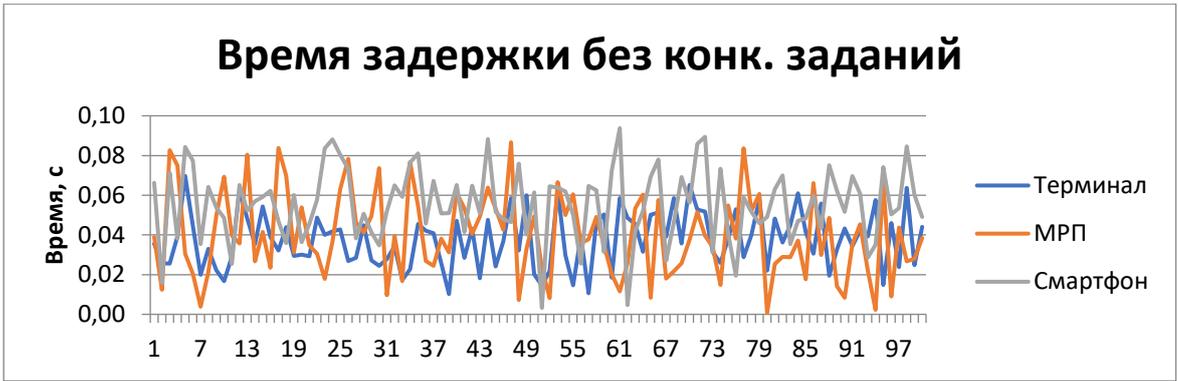
| Показатель | Среднее значение | Возможные значения | Допустимые значения |
|-------------------------------------------------------------------------------------------------------|------------------|--------------------|---------------------|
| Время загрузки приложения, с | 0.59 | [0.22; 0.92) | [0; 1] |
| Время инициализации приложения, с | 0.41 | [0.17; 0.41) | [0; 1] |
| Время актуализации данных в приложении, с | 0.08 | [0.01; 0.96) | [0; 0.5] |
| Относительная доля выполненных заявок | 0.96 | [0,93; 0,99] | [0,9; 1] |
| Относительная доля выполненных заявок, взвешенная по приоритету | 0.95 | [0,92; 0,96] | [0,92; 0,96] |
| Относительная доля отказов в выполнении пользовательских заданий при отсутствии конкурирующих заданий | 0,04 | [0,03; 0,05] | [0; 0,7] |
| Относительная доля отказов в выполнении пользовательских заданий при наличии конкурирующих заданий | 0,05 | [0,02; 0,05] | [0,02; 0,05] |
| Время задержки в выполнении пользовательских заданий при отсутствии конкурирующих заданий, с | 0,04 | [0; 0,9) | [0; 1] |
| Время задержки в выполнении пользовательских заданий при наличии конкурирующих заданий, с | 0.58 | [0.03; 1.3) | [0; 1,5] |
| Время отклика пирингового соединения, мс | 5.81 | [0; 10) | [0; 10] |

| Показатель | Среднее значение | Возможные значения | Допустимые значения |
|-------------------------------------------------------------------------|------------------|--------------------|---------------------|
| Потребление ресурсов процессора серверной части приложения, % | 0.09 | [0; 0.22) | [0; 20] |
| Потребление ресурсов оперативной памяти серверной части приложения, МБ | 16.79 | [9.52; 23.22) | [0; 1000] |
| Потребление ресурсов процессора клиентской части приложения, % | 2.97 | [0.6; 5.54) | [0; 10] |
| Потребление ресурсов оперативной памяти клиентской части приложения, МБ | 14.24 | [12.2; 16.5) | [0; 100] |
| Коэффициент потери пакетов (PLR), % | 0,99 | [0.8; 1.13) | [0; 2] |

На рис. 16 приведены результаты измерений параметров QoS работы сервиса на различных аппаратных платформах: стационарных терминалах, мобильных платформах и смартфонах.



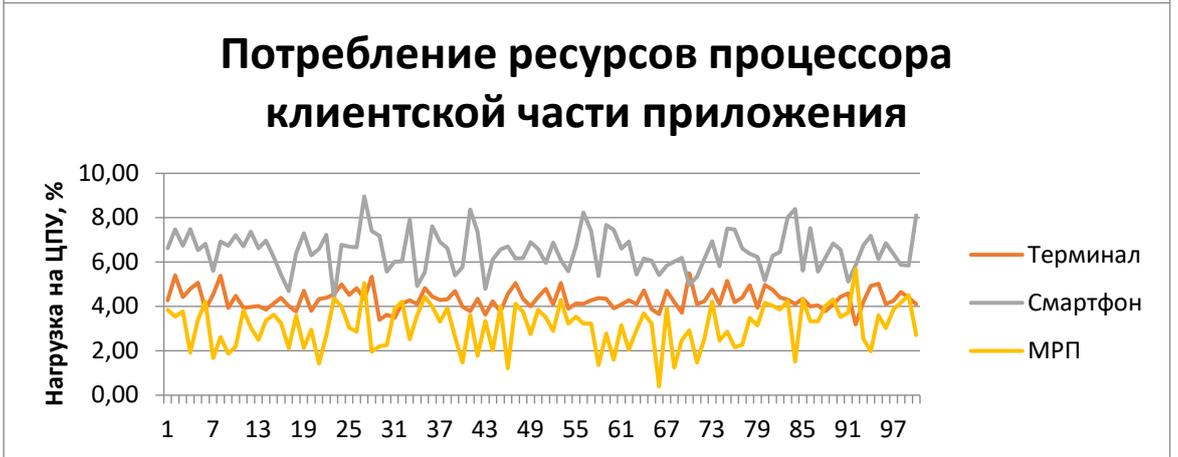
Г



Д



е



Ж



Рис 16. Результаты тестирования сервиса ВКС

Данные замеров параметров QoS, представленные на рис. 16, свидетельствуют об отсутствии существенных различий в работе приложения ВКС на различных платформах с точки зрения конечного пользователя. При этом значения замеров остаются в пределах допустимого диапазона, что позволяет сделать вывод о соответствии качества работы приложения пользовательским ожиданиям в стандартных сценариях работы приложений. Результаты, представленные на рис. 16 е,ж, позволяют сделать вывод, что потребление аппаратных ресурсов приложением ВКС на различных платформах остаётся в пределах допустимого диапазона. Лучше всего приложение показало себя на мобильной платформе, хуже – на смартфоне. При этом различие в расходе ресурсов на устройствах с разной производительностью остаётся несущественным, обеспечивая схожий опыт использования ВКС.

Разработанная архитектура соединения пользователей в приложении ВКС обладает рядом преимуществ, способствующих конкурентоспособности разработанного приложения на фоне аналогов. Предложенная организация модулей ПО сетевого соединения приложения ВКС предоставляет пространство для внедрения дополнительных функций в приложение. В рамках КФС управляемые аккаунты могут использоваться в модулях обеспечения безопасности и управления доступом, к примеру, для обеспечения контроля за рабочими местами, производством и другими объектами. Организация хранения и обработки данных позволяет внедрить модуль ВКС в КФС, а также сделать удобным его использование как на терминалах и мобильных робототехнических платформах, так и на клиентских мобильных устройствах.

2.3. Сервис корпоративного телевидения

Сервис КТ взаимодействует с пользователями с помощью стационарных камер и экранов, расположенных в разных местах организации. Кроме того, пользователи могут управлять сервисом с помощью мобильных устройств. В функции сервиса входит трансляция на стационарные экраны информации для сотрудников и посетителей (сведения об институте и его деятельности, демонстрация разработок, объявления, приветствия, поздравления) по их запросу и/или в соответствии с расписанием. Параметры QoS этого сервиса связаны с задержками, отказами и потерями, возникающими при трансляции медиаконтента. Для моделирования сервиса КТ используем параметры, представленные в табл. 4.

Таблица 4. Параметры QoS сервиса КТ

| Показатель | Обозначение | Возможные значения | Допустимые значения |
|-------------------------------------------------------------------------|-----------------------|--------------------|------------------------|
| Задержка между планируемым и фактическим временем трансляции медиа, с | t_{Delay}^{CT} | $[0; \infty)$ | $[0; t_{distraction}]$ |
| Относительное время простоя | f_{down}^{CT} | $[0; 1]$ | $[0; t_{distraction}]$ |
| Соотношение времени трансляции и объёма транслируемой информации, бит/с | $r_{perception}^{CT}$ | $[0; \infty)$ | $[R_{min}; R_{max}]$ |
| Время смены медиафайлов, с | t_{load}^{CT} | $[0; \infty)$ | $[0; t_{UI}]$ |
| Относительная доля неуспешных загрузок контента | f_d^{CT} | $[0; 1]$ | $[0; p_D]$ |
| Относительная доля потерь по времени | f_i^{CT} | $[0; 1]$ | $[0; 1-I/I_{max}]$ |
| Относительная доля потерь по области отображения | f_i^{CT} | $[0; 1]$ | $[0; 1-I/I_{max}]$ |
| Коэффициент искажения | k_d^{CT} | $[0; \infty)$ | $[0; 1-I/I_{max}]$ |

В таблице используются следующие обозначения:

$t_{distraction}$ – время, по истечении которого пользователь с большой вероятностью отвлечётся на другие задачи;

I – количество информации в медиафайле, бит;

I_{max} – норма максимально допустимого количества информации в медиафайле, бит;

R_{min}, R_{max} – оценки минимальной и максимальной скорости восприятия информации пользователем, бит/с.

В табл. 5 представлены экспериментальные значения QoS сервиса КТ, как общие, так и специфические.

Таблица 5. Экспериментальные параметры QoS сервиса КТ

| Показатель | Обозначение | Среднее значение | Допустимые значения |
|-------------------------------------------------------------------------------------------------------|------------------------|------------------|---------------------|
| Время загрузки приложения, с | t_f^{App} | 2,73 с | [0; 10] |
| Время инициализации приложения, с | t_i^{App} | 0,04 с | [0; 5] |
| Время актуализации данных в приложении, с | t_a^{App} | 1,15 с | [0; 5] |
| Относительная доля выполненных заявок | $f_{np}^{Success}$ | 0,98 | [0,9; 1] |
| Относительная доля отказов в выполнении пользовательских заданий при отсутствии конкурирующих заданий | f_{normal}^{Denial} | 0,02 | [0; 0,1] |
| Время задержки в выполнении пользовательских заданий при отсутствии конкурирующих заданий, с | $t_{normal}^{Request}$ | 0,53 с | [0; 1] |
| Задержка между планируемым и фактическим временем трансляции медиа, с | t_{Delay}^{CT} | 0,53 с | [0; 5] |
| Относительное время простоя | f_{down}^{CT} | 0,02 | [0; 0,1] |
| Соотношение времени трансляции и объёма транслируемой информации, бит/с | $r_{perception}^{CT}$ | 50 609 бит/с | |
| Относительная доля неуспешных загрузок контента | f_d^{CT} | 0,02 | [0; 0,01] |
| Относительная доля потерь по времени | f_t^{CT} | 0,02 | [0; 0,1] |
| Относительная доля потерь по области отображения | f_i^{CT} | 0 | [0; 0,01] |
| Коэффициент искажения | k_d^{CT} | 0 | [0; 0,01] |

Показатели относительной доли потерь по области изображения и коэффициента искажения – общие показатели качества обслуживания, которые существенны для телевизионных трансляций и стриминговых сервисов. Поскольку сервис КТ реализован в приложении на стационарных терминалах и МРП, а также клиентских устройствах, данные показатели не рассматриваются ввиду отсутствия искажения транслируемого изображения.

На рис. 17 приведены результаты измерений параметров QoS работы сервиса КТ на стационарных терминалах.

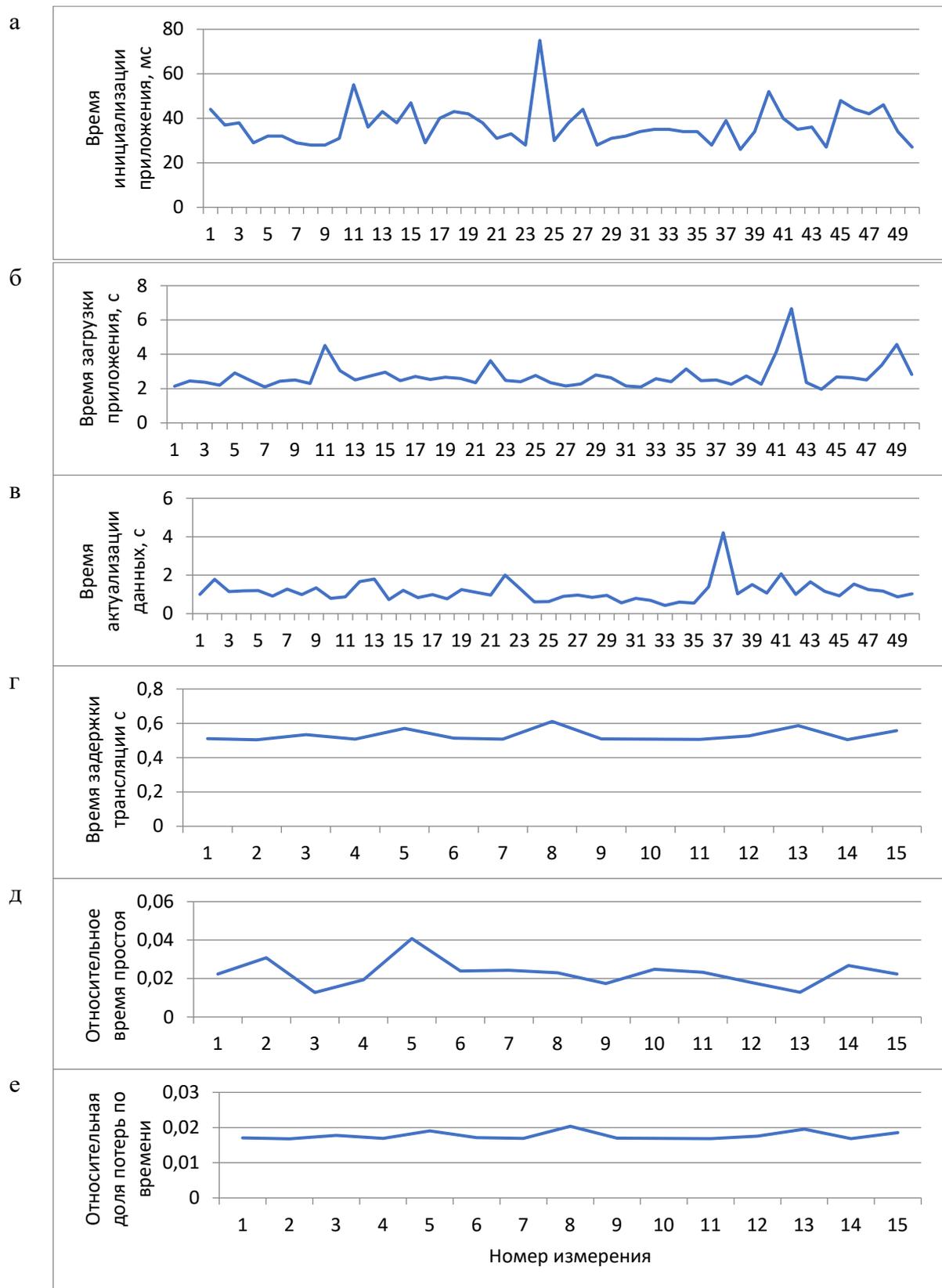


Рис. 17. Результаты тестирования сервиса КТ

Данные тестирования сервиса КТ, представленные на рис. 17, свидетельствуют о довольно высокой производительности в режиме реального времени, что позволяет сделать вывод о соответствии качества работы приложения пользовательским ожиданиям в стандартных сценариях работы приложений. В рамках работ над модулем КТ была сформирована архитектура сервиса, разработано клиентское и серверное программное обеспечение для управления системой, а также реализован веб-интерфейс администратора этой системы. Данная система корпоративного телевидения использует стандартные

сетевые технологии и не привязана к конкретным программно-аппаратным платформам, а также соответствует критериям расширяемости и переносимости. Модуль КТ предусматривает возможность интерактивности и персонализации путем интеграции с системными сервисами идентификации и хранилищем профилей пользователей.

2.4. Сервис локализации и навигации

Для моделирования сервиса локализации и навигации в киберфизическом пространстве была построена тестовая модель в среде Unity 3D, имитирующая этаж здания и людей, передвигающихся по нему. На QoS сервиса локализации и навигации внутри помещений влияет морфология среды, структура системы локализации, а также свойства используемой технологии (WiFi, UWB, ZigBee, RFID, Bluetooth, и тд.). Основные параметры, которые при этом учитываются – это точность локализации пользователя (Localization Accuracy), определяемая в нашем случае как ошибка предсказания обученного алгоритма, и частота отчета о местоположении (Position Report Frequency), определяемая на стадии разработки аппаратной части системы (табл. 6).

Таблица 6. Параметры QoS сервиса ЛН

| Показатель | Обозначение | Возможные значения | Допустимые значения |
|--------------------------------------------|-------------|--------------------|--------------------------|
| Точность локализации пользователя | Acc_{loc} | [0; 1] | $[Acc_{min}; 1]$ |
| Частота отчета о местоположении (PRF), 1/с | PRF_{loc} | $[0; PRF_{max}]$ | $[PRF_{min}; PRF_{max}]$ |

В таблице используются следующие обозначения:

Acc_{min} – минимально допустимая точность алгоритма локализации;

$PRF_{min}; PRF_{max}$ – минимально и максимально допустимые значения частоты отчета о местоположении.

В рамках выполнения проекта при реализации сервиса ЛН было произведено сравнение модулей, использующих беспроводные технологии передачи данных, с точки зрения обеспечения требуемой скорости передачи данных для проектирования КФП в зависимости от предполагаемого расстояния между модулями. Рассмотрены наиболее популярные и бюджетные решения, представленные сегодня на рынке: Wi-Fi - Bluetooth модули: ESP32, ESP8266, модуль LoRa - SX 1276 и Bluetooth модули HC-05 и HC-06. По результатам исследования (табл. 7), были выявлены следующие достоинства и недостатки рассмотренных технологий.

У Wi-Fi-модулей к положительным качествам относятся самая высокая пропускная способность, большой радиус действия, низкая стоимость, защищенность, к отрицательным – высокая загруженность частот диапазона Wi-Fi, из-за чего может происходить потеря пакетов данных.

Преимуществом модулей LoRa является большой территориальный охват низкочастотных радиоволн, что позволяет отправлять на данные на расстояние свыше 1 км, но отрицательным характеристикам относится низкая скорость передачи данных, что также объясняется передачей данных на низкой частоте.

Главным преимуществом модулей Bluetooth является низкая стоимость и, сравнительно с LoRa, высокая скорость передачи данных. При этом их отличает маленький радиус действия, что не позволяет их использовать на больших территориях.

При построении беспроводной вычислительной сети КФП модули LoRa целесообразно использовать для передачи небольшого объема данных, как правило, управляющей информации. Для организации передачи неслужебных данных средствами КФП условиях офисных помещений более подходящими являются модули Bluetooth и Wi-Fi, благодаря их высокой скорости передачи данных на относительно небольшие расстояния.

Таблица 7. Параметры модулей беспроводных технологий передачи данных

| модуль | F, МГц | SOM, дБ | P _t , dBm | P _{sen} , kbps | |
|-----------------------------|--------|---------|----------------------|-------------------------|-----------------------|
| | | | | Скорость, kbps | Чувствительность, dBm |
| ESP-32 и ESP-8266 WiFi | 2412 | 20 | 16,5 | 72 | -71 |
| | | | | 54 | -75 |
| | | | | 32 | -79.5 |
| | | | | 11 | -91 |
| | | | | 6 | -93 |
| | | | | 1 | -98 |
| SX 1276 Lora | 625 | 20 | 13 | 1.2 | -121 |
| | | | | 4.8 | -117 |
| | | | | 38.4 | -108 |
| | | | | 250 | -95 |
| ESP-32 и ESP-8266 Bluetooth | 2412 | 20 | 4 | 3 | -84 |
| | | | | 2 | -98 |
| HC-05 и HC-06 | 2412 | 20 | 4 | 2 | -80 |

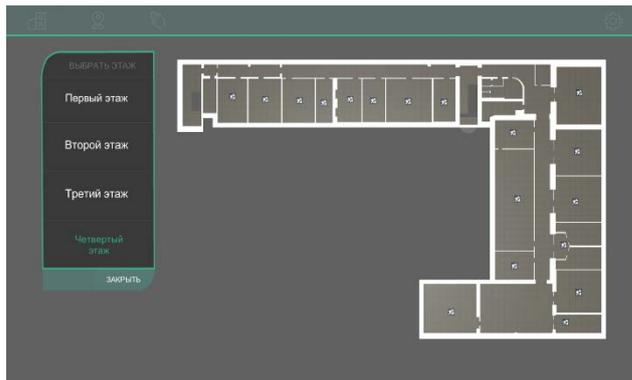
Разработанное приложение для локализации и навигации предоставляет пользователю возможность определения своего местоположения в системе, а также построения маршрута между заданными точками. Разработаны административная и пользовательская части приложения. Первая предназначена для администраторов системы и служит инструментом работы с геолокационными метками, которые можно создавать, удалять и редактировать. Пользовательская часть предназначена для людей, которые могут, отсканировав метку при помощи камеры, определить свое местоположение и использовать его как отправную точку для построения маршрута. Точки для построения маршрута можно выбирать как вручную на карте, так и уже из списка имеющихся в системе меток. Изначально пользователю нужно выбрать необходимый этаж здания и уже на нем выбрать нужную метку.

Маршруты можно строить как на одном этаже, так и с одного этажа на другой. На рисунке 18 б изображен маршрут, построенный проложенный из помещения 404 в помещение 416. На рисунке 18 в,г изображено построение маршрута из помещения 404 на четвертом этаже в помещение 317 на третьем этаже. При локализации пользователя при помощи сканирования маркера возникает ошибка определения местоположения. После тестирования были получены результаты ошибки. В основном ошибка лежит в интервале от 0 до 1 метра. Среднее значение ошибки составило 0.38 метра (рис. 18 е). В табл. 8 приведена значения общих параметров QoS для сервиса ЛН.

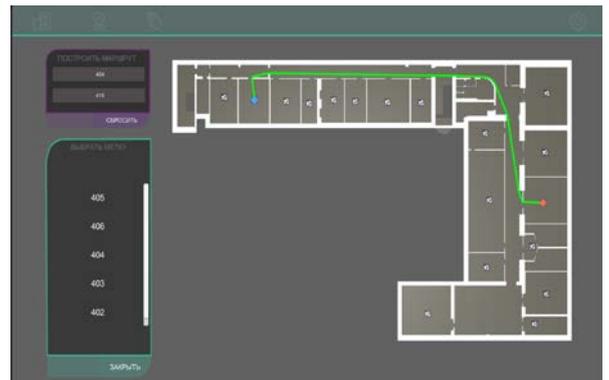
Таблица 8. Параметры QoS сервиса ЛН

| Показатель | Среднее значение | Возможные значения | Допустимые значения |
|-------------------------------------------------------------------------------------------------------|------------------|--------------------|---------------------|
| Время загрузки приложения, с | 4,5 | [2,1;7,2) | [0; 10] |
| Время инициализации приложения, с | 2,5 | [0,84; 3,84) | [0; 5] |
| Время актуализации данных в приложении, с | 2,1 | [0,97; 3,9) | [0; 5] |
| Относительная доля выполненных заявок | 0,96 | [0,93; 0,99] | [0,9; 1] |
| Относительная доля выполненных заявок, взвешенная по приоритету | 0,94 | [0,92; 0,96] | [0,9; 1] |
| Относительная доля отказов в выполнении пользовательских заданий при отсутствии конкурирующих заданий | 0,03 | [0,02; 0,04] | [0; 0,7] |

| Показатель | Среднее значение | Возможные значения | Допустимые значения |
|----------------------------------------------------------------------------------------------------|------------------|--------------------|---------------------|
| Относительная доля отказов в выполнении пользовательских заданий при наличии конкурирующих заданий | 0,03 | [0,02; 0,05] | [0; 0,1] |
| Время задержки в выполнении пользовательских заданий при отсутствии конкурирующих заданий, с | 0,1 | [0; 0,19) | [0; 1] |
| Время задержки в выполнении пользовательских заданий при наличии конкурирующих заданий, с | 0,33 | [0,11; 0,6) | [0; 1,5] |



а



б



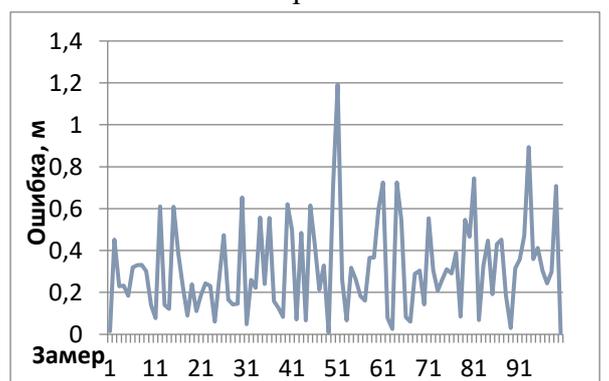
в



г



д



е

Рис. 18. Приложение ЛН: а) Модель этажа; б) построение маршрута на одном этаже; в, г) построение маршрута на разных этажах, д) маршрут, заканчивающийся в произвольно заданной точке, е) измерения ошибки локализации
 Результаты тестирования работы приложения локализации и навигации на различных аппаратных средствах представлены на рис. 19.

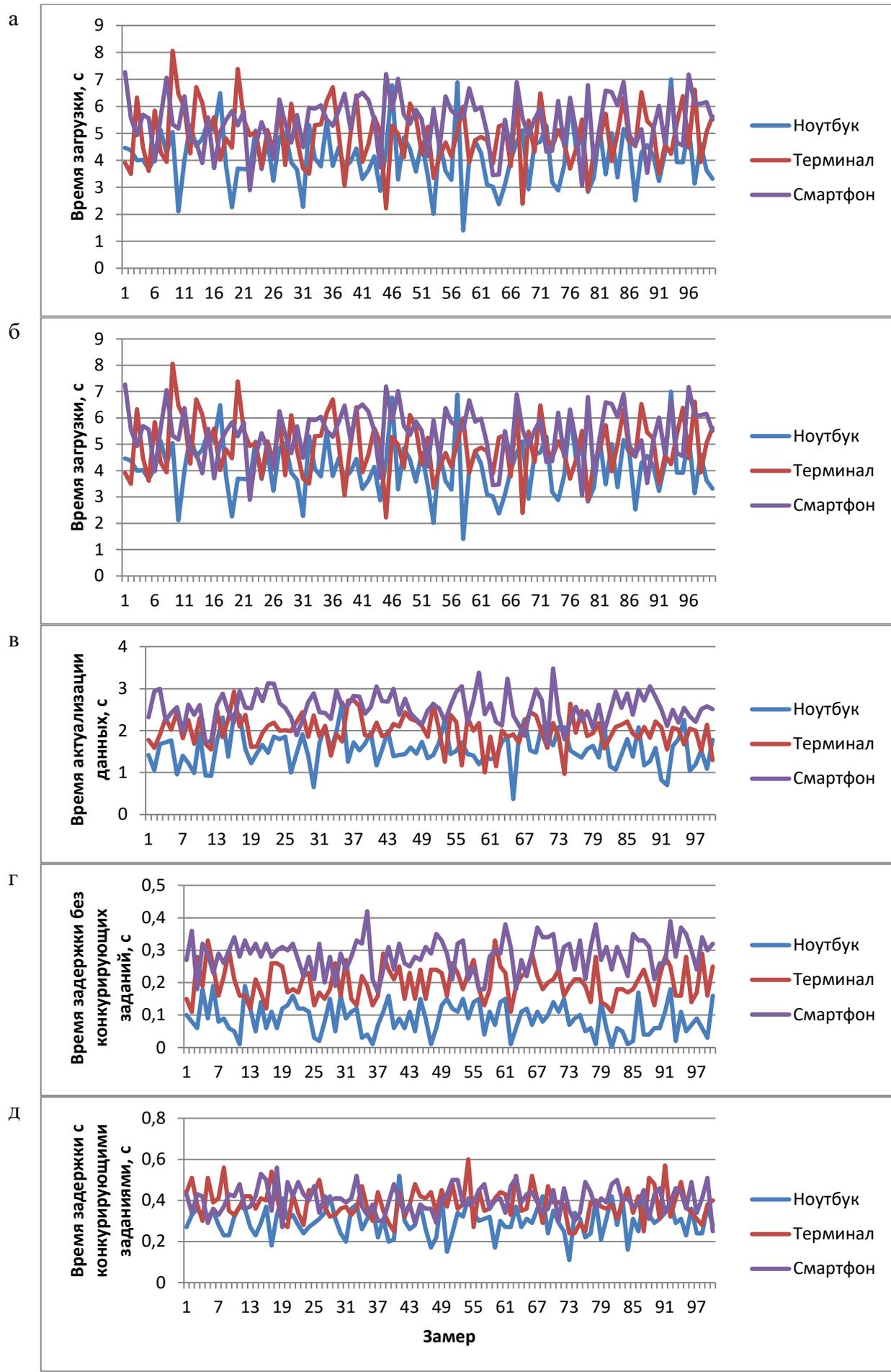


Рис.19. Результаты тестирования сервиса ЛН

Результаты тестирования сервиса ЛН, представленные на рис. 19, свидетельствуют о приемлемом с точки зрения пользователя качестве работы сервиса независимо от аппаратной платформы. Кроме того, информация о фактическом расположении пользователя в интеллектуальном пространстве дает возможность оптимизировать работу других систем и сервисов КФП, включая сервис персонализации и инженерные сети, что способствует более эффективному распределению ресурсов организации. Кроме того, сервис ЛН обеспечивает налаживание дополнительных взаимосвязей между элементами интеллектуального пространства, а исторические данные о перемещениях пользователя способствует более эффективному распределению задач между средствами КФП.

2.5. Модель комплексной оценки параметров качества сервисов МИНОС

Чтобы оценить работу системы распределения задач между средствами КФП были смоделированы параметры качества восприятия сервисов QoS для всех сервисов системы и математическое представление отношений ключевых показателей производительности. При решении данной задачи был использован алгоритм кластеризации k-means++ (реализация библиотеки scikit-learn). Исходные данные были предварительно стандартизированы: среднее по всей выборке было приведено к 0, дисперсия приведена к 1. Для выделения ключевых параметров, вносящих наибольший вклад в оценку пользователем сервисов КФП, удобно воспользоваться методами контролируемой и полуконтролируемой кластеризации (supervised и semi-supervised clustering) или методами самоорганизующихся карт (self-organized maps). При выборе параметров алгоритма количество кластеров определяется выбранной шкалой оценивания сервисов (в нашем случае, от 1 до 5). Результат кластеризации после снижения размерности пространства с 46 до 2 с помощью метода главных компонент (реализация библиотеки scikit-learn) для каждого сервиса представлен ниже (рис. 20).

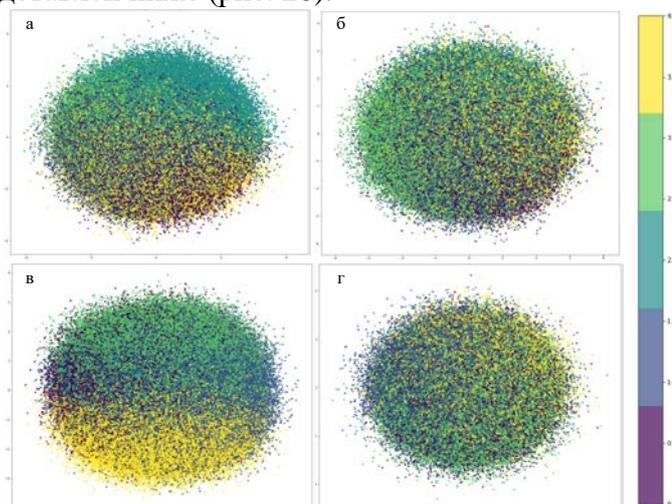


Рис. 20. Результат кластеризации данных QoS сервисов КФП: а) ВКС; б) КТ; в) ЛН; г) интегральная оценка сервисов

Предложенная модель оценки качества восприятия сервисов КФП позволяет на основе исторических данных об объективных параметрах качества обслуживания пользователей сделать вывод о том, насколько пользователь будет удовлетворен предоставленными ему сервисами. Дальнейший анализ полученных данных дает возможность оценить, какие параметры качества вносят наибольший вклад в субъективную оценку сервиса пользователем, а также, какие параметры сервисов КФП требуют доработки и улучшения. Представленная модель, безусловно, требует валидации на основе анкетирования пользователей, однако может использоваться уже на этапе разработки для мониторинга ключевых показателей производительности системы и проверки их на соответствие предъявляемым требованиям.

3. Архитектура программного обеспечения управления движением антропоморфного робота Антарес

В качестве платформы для разработки программного обеспечения управления движением антропоморфного робота Антарес выбрана операционная система – Linux Ubuntu 18.04, фреймворк ROS (Robot Operating System), а именно последняя версия фреймворка – ROS Melodic. Языки разработки: C++, Python; также используются языки разметки для конфигурационных и других описательных файлов: XML, YAML, URDF.

Рассмотрим архитектуру программного обеспечения робота Антарес на базе ROS. ROS мастер-сервер (ROS master) выступает как сервис имен для вычислительного графа ROS. Он содержит в себе информацию о топиках и сервисах ROS, которая необходима для работы узлов ROS. При запуске любого узла, данный узел связывается с мастер-сервером для сообщения информации о себе другим узлам.

Узлы связываются с другими узлами напрямую, мастер-сервер предоставляет только сервис имен. Узлы, которые подписываются на какой-либо топик, будут запрашивать соединения у узлов, которые публикуют сообщения в этот топик по заранее определенному протоколу связи. В работе используется протокол TCPROS, который использует стандартные TCP/IP сокеты. Сообщения передаются через интерфейс передачи и сериализации данных. Узел отправляет сообщение другим узлам путем публикации сообщения в выбранную тему (топик). Тема – имя, используемое для идентификации содержимого сообщения. Узлы подписываются на соответствующие темы (топики), для получения нужных им сообщений

Такая архитектура позволяет выполнить работу узлов с слабым сцеплением, т.е. с низкой взаимозависимостью между узлами, где имена являются основным средством для построения сложных систем. Каждая клиентская библиотека ROS поддерживает переназначение имен, т.е. скомпилированная программа может быть изменена во время работы для создания другой топологии вычислительного графа. Программное обеспечение робота Антарес включает следующие модули:

- Модуль конфигурирования и взаимодействия с сервоприводами. Реализован при помощи библиотеки для Dynamixel, поставляемой компанией Robotis (открытое бесплатное ПО);
- Модуль кинематики. В модуле реализованы алгоритмы обратной кинематики для нижних и верхних конечностей робота;
- Модуль походки. На данный момент модуль содержит нативную реализацию походки, представляющую собой адаптированную версию алгоритма походки Darwin OP.
- Модуль камеры. Получает изображения и отправляет в соответствующий ROS-топик;
- Модуль Point2World. По положению пикселя на картинке рассчитывает положение объекта в пространстве в локальных координатах робота;
- Модуль LookAt. Рассчитывает углы сервоприводов головы робота для направления камеры на нужный объект в пространстве.
- Модуль, получающий данные с IMU-сенсоров.
- Модуль фильтров для IMU-сенсоров.

График узлов и топиков ROS робота Антарес представлен на рис. 21.

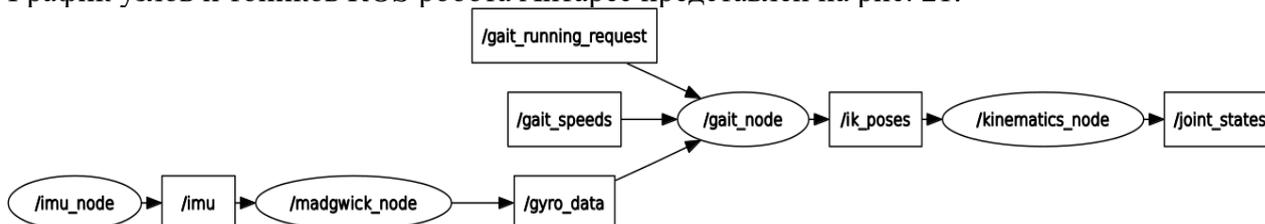


Рис. 21. График узлов и топиков ROS робота Антарес

Структура программных пакетов, использующихся для реализации программного обеспечения Антарес, представлена в табл. 9.

Таблица 9. Структура программных пакетов ROS

| Пакет | Назначение | Описание, классы, узлы |
|-----------------------------|----------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| antares | метапакет для ROS | используется для группировки других пакетов и удобного управления ими |
| antares_control | пакет для управления сервоприводами dynamixel | Класс joints_handler – генерирует траектории до целевой точки Принимает сообщения sensor_msgs::JointState Публикует сообщения sensor_msgs::JointTrajectory |
| | | Класс joints_control_module – модуль для контроллера robotis_controller для управления сервоприводами Принимает сообщения sensor_msgs::JointState |
| antares_description | пакет для описания структуры робота в urdf формате | Содержит .launch файлы для запуска визуализации в rViz |
| antares_gait | пакет для управления походкой | Узел imu_node – получает данные с IMU по I2C шине Публикуемые топики: /imu [antares_msgs/Imu_data] – данные с IMU Узел gait_node – генерирует походку |
| | | Публикуемые топики: /ik_poses [antares_msgs/IK_poses] – положения для обратной кинематики Подписанные топики: /gait_running_request [std_msgs/Bool] – включение/выключение походки /gait_speeds [geometry_msgs/Twist] – угловые и линейные скорости /gyro_data [geometry_msgs/Point] – интегрированные данные с гироскопа |
| antares_kinematics_dynamics | пакет для кинематики и динамики | Узел kinematics_node – рассчитывает обратную кинематику Публикуемые топики: /joint_states [sensor_msgs/JointState] – Состояния для сервоприводов Подписанные топики: /ik_poses [antares_msgs/IK_poses] – положения для обратной кинематики |
| antares_manager | пакет для запуска контроллера для сервоприводов | Узел antares_manager_node – запускает управление сервоприводами реальными и в симуляторе Публикуемые топики: /robotis/present_joint_states – текущие состояния сервоприводов Подписанные топики: /joint_states [antares_msgs/IK_poses] – целевые состояния для сервоприводов |

| | | |
|----------------|----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| antares_msgs | пакет для описания сообщений ROS | Описывает сообщения, использующиеся во всех пакетах проекта |
| antares_vision | пакет для компьютерного зрения | Узел camera_publisher_node – получает и публикует данные с камеры Публикуемые топики: /camera/image [sensor_msgs/Image] – изображение с камеры |
| | | Узел point2world_node – Получает точку в системе координат изображения, конвертирует в с.к. мира и публикует углы для поворота камеры в точку Публикуемые топики: /rotate_camera [antares_msgs/Camera_rotation] – целевая ориентация камеры Подписанные топики: /img_point [antares_msgs/Img_point] – точка на изображении |
| antares_vrep | пакет для динамической симуляции в программе V-Rep | Модуль для описания робота antares в формате URDF |

Программные пакеты (package) – основной способ организации ПО в ROS. Пакеты содержат в себе узлы, наборы данных, файлы конфигурации и т.п. Рассмотрим основные узлы, реализующие управление движением робота Антарес (рис. 22).

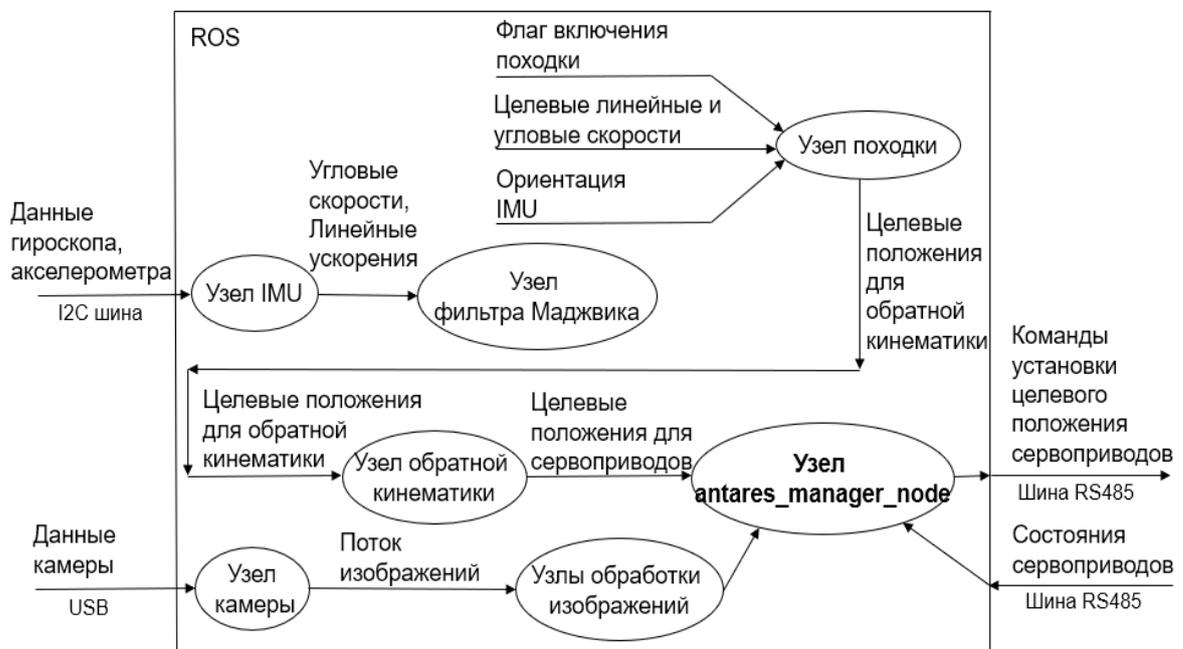


Рис. 22. Структура основных программных модулей Антарес

Узел antares_manager_node, при получении требуемых целевых состояний для сервоприводов от других узлов, отправляет данные сервоприводам через протокол Dynamixel Protocol 1.0 используя официальную программную реализацию управления по этому протоколу в библиотеке ROBOTIS Framework. Узел antares_manager_node публикует текущие и целевые состояния сервоприводов для других узлов ROS.

Для стабилизации работы походки используются данные IMU-сенсоров. Узел IMU получает данные от IMU по I2C шине и публикует угловые скорости и линейные ускорения IMU для узла фильтра Маджвика, который публикует ориентацию IMU в пространстве для использования в походке.

Узел походки принимает ориентацию IMU, целевые линейные и угловые скорости робота от пользователя и генерирует целевые точки для стоп в системе координат робота для расчета обратной кинематики. Затем, узел обратной кинематики рассчитывает целевые состояния для сервоприводов, и публикует их для узла `antares_manager_node`.

Использование ROS в архитектуре ПО Антареса позволяет выполнять программные элементы робототехнической системы, представленные узлами ROS, с низкой взаимозависимостью между друг другом. Это позволяет повторно использовать уже написанный код в разных робототехнических системах, работающих на ROS, а также позволяет проводить быстрое переконфигурирование архитектуры ПО робототехнической системы во время разработки. Разработанная архитектура робототехнической системы позволяет проводить быстрое прототипирование и тестирование робототехнического ПО, за счет использования инструмента визуализации `rViz`, симулятора `Gazebo` и различных встроенных библиотек. Использование ROS позволяет реализовать одинаковую среду разработки и тестирования робототехнического ПО для всех участников и как следствие, это ускоряет разработку ПО с большим количеством участников.

Для тестирования алгоритмов управления движением робота Антарес была построена симуляционная модель робота в формате URDF, которая позволяет без изменения системы работать с симуляторами и визуализаторами ROS. Модель включает в себя визуальные части, необходимые для отображения, и части коллизий, необходимые для симуляции динамики и физики. Части коллизий представлены примитивами для упрощения вычислений и ускорения работы симуляторов.

Была проработана модульная концепция робота, из конструкции исключены открытые для воздействия внешней среды механические передачи. Для снижения износа проводных соединений и осевых тазовых приводов робота разработан новый тазовый механизм робота, гасящий ударные нагрузки, возникающие при ходьбе робота, кулачковой упругой муфтой с пропускным каналом, отведенным для кабелей питания и управления педипуляторами робота (рис. 23). Упругая муфта гасит ударные нагрузки при передвижении робота благодаря демпферам из силикона и FLEX-пластика, используемого в 3D-печати. При сборке узла для снижения трения и увеличения срока службы деталей используются прокладки из фторопласта. Для проведения полноценного моделирования и анализа нагрузочных тестов полной компьютерной модели робота имеющиеся вычислительные мощности оказались недостаточны, вследствие чего натурные тесты робота проводились с подстраховкой со стороны оператора. Установлены датчики для ориентации робота в пространстве: модуль камеры и микрофона для получения аудиовизуальной информации и модуль гироскопа, акселерометра и магнетометра. Протестирована базовая походка прототипа робота при движении по ровной поверхности с копированием движений из симуляционной модели.

Вес готового к работе робота составляет 6,6 кг. Рост робота составляет 77 см. Базовый каркас робота обладает 20 степенями свободы. Возможно увеличение количества степеней свободы за счет установки захватных устройств и наклонно-осевых механизмов. Манипуляторы и педипуляторы робота оснащены специальными хабами, контролирующими питание и управляющими сервоприводами (рис. 24). Каждый основной узел робота имеет собственный субконтроллер и вспомогательный аккумулятор, позволяющий использовать его как отдельное устройство. Основной и вспомогательные аккумуляторы образуют распределенную систему питания, снижая потери энергии при питании конечных звеньев исполнительных механизмов робота.



Рис. 23. Тазовый трехосевой механизм

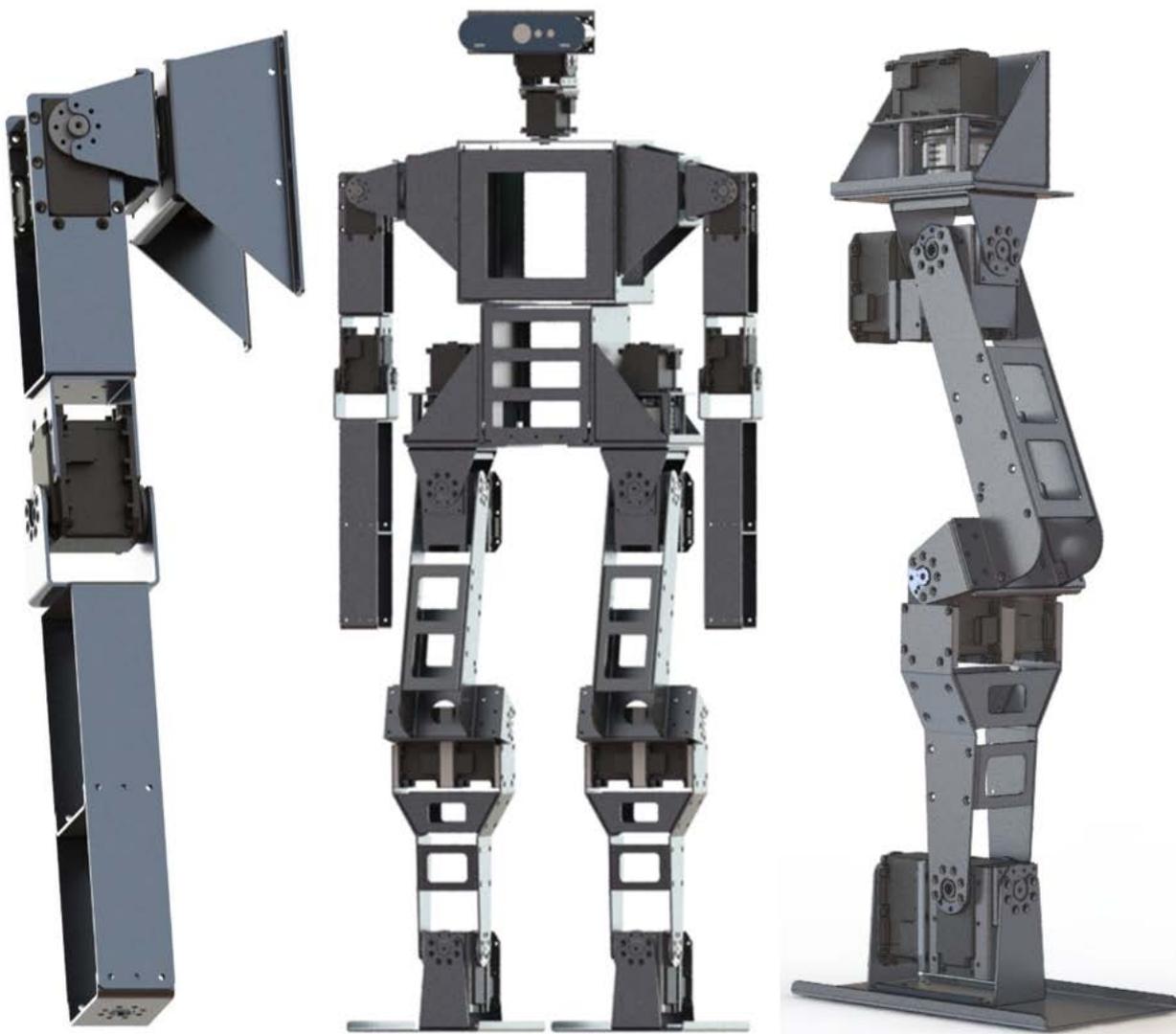


Рис. 24. Манипулятор. Общий вид каркаса робота. Педипулятор

Субконтроллеры выполняют роль хабов, объединяя в себе системы управления сервоприводами и питанием робота. Несущий каркас робота изготовлен из алюминиевых сплавов АМг6 и Д16Т. В конструкции робота используются демпфирующие элементы, изготовленные с применением 3D-печати, гасящие ударные нагрузки, а также фторопластовые элементы для снижения трения в узлах робота. В конструкции использовано 16 сервоприводов Dynamixel MX-64 в педипуляторах и плечевых суставах, а также 6 сервоприводов Dynamixel MX-28 в малонагруженных узлах. Бедренные узлы робота разработаны для размещения внутри дополнительных аккумуляторов, стабилизирующих питание в ногах робота. Для снижения нагрузки коленный сустав включает спаренные сервоприводы MX-64. Голеностопный и тазобедренный суставы по функциональности подобны суставам в человеческом организме. Для очувствления робота используются Камера Logitech Brio (разрешение до 4096x2160; угол обзора 90 градусов; максимальная частота 90 Гц; встроенный микрофон; автофокус; цифровой 5-кратный зум; инфракрасный датчик для распознавания лиц) и IMU-сенсор (3-осный акселерометр LIS331DLH; 3-осный гироскоп L3G4200D; 3-осный магнетометр LIS3MDL; протокол связи I²C). Рабочий прототип робота Антарес представлен на рис. 25.

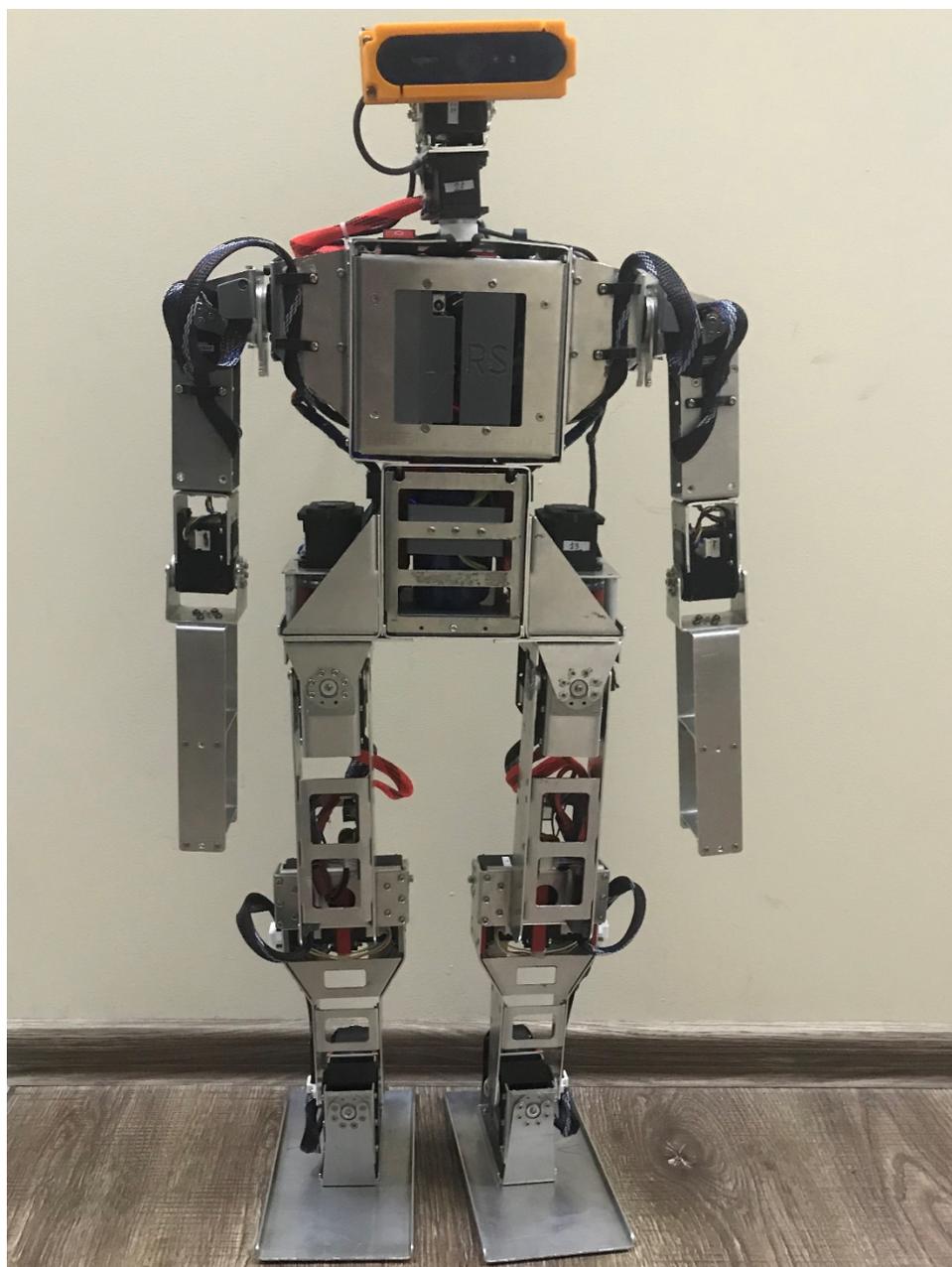


Рис. 25. Рабочий прототип Антарес

Архитектуры, методы и алгоритмы взаимодействия гетерогенных средств киберфизического интеллектуального пространства имеют существенное значение для развития цифровых технологий и роботизированных систем. Предложенные подходы и способы взаимодействия распределенных гетерогенных встроенных, клиентских, робототехнических средств в киберфизическом интеллектуальном пространстве обеспечивают персонализированное информационно-навигационное обслуживание пользователей разных категорий. Разработанные методы оценки качества восприятия сервисов информационно-навигационного обслуживания, предоставляемых киберфизическим интеллектуальным пространством, позволяют сделать вывод о том, насколько пользователь удовлетворен предоставленными ему сервисами, какие параметры качества вносят наибольший вклад в субъективную оценку сервиса пользователем, а также, какие параметры сервисов КФП требуют доработки и улучшения. Дальнейшие исследования предусматривают тестирование и улучшение предложенных архитектур, методов и алгоритмов.

Получен патент на изобретение № 2638003 от 08.12.2017 «Способ распределения задач между сервисными роботами и средствами киберфизического интеллектуального пространства при многомодальном обслуживании пользователей», авторы: Ронжин А.Л., Савельев А.И., патент зарегистрирован в ЕГИСУ НИОКТР: <https://rosrid.ru/rid/DKEKK5PGLPUO71HYB1HP1QNA>

По результатам 2018 года опубликованы следующие статьи, включая публикации, принятые к печати в 2017-м году, но не попавшие в отчет 2017-го года:

Публикации, поданные к печати, но не опубликованные в 2017 г.:

1. Карасёв Е. Ю., Ватаманюк И. В., Савельев А. И., Ронжин А. Л. Архитектурные решения интеграции модуля видео-конференц-связи в киберфизическое интеллектуальное пространство // Информационно-управляющие системы. 2018. № 1. С. 2–10. doi:10.15217/issn1684-8853.2018.1.2
2. Павлюк Н. А., Смирнов П. А. Моделирование и оптимизация каркасных элементов торса и тазового механизма антропоморфного робота «Антарес» // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. 2017. Т. 7, № 3(24). С. 58–65.
3. Павлюк Н.А. Моделирование опорной конструкции тазового механизма антропоморфного робота Антарес // Экстремальная робототехника. 2017. № 1. С. 155-160.
4. Ватаманюк И.В., Савельев А.И. Мобильная робототехническая платформа как компонент киберфизического интеллектуального пространства // Экстремальная робототехника. 2017. № 1. С. 37-42.
5. Vu, Q., Nguyen, V., Solenaya, O., Ronzhin, A., Mehmet, H. Algorithms for joint operation of service robotic platform and set of UAVs in agriculture tasks //Proceedings of the 5th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering, AIEEE 2017. 2018. pp. 1-6.

Статьи, опубликованные в 2018 г.:

1. Levonevskiy D., Vatamaniuk I., Saveliev A. Processing models for conflicting user requests in ubiquitous corporate smart spaces. MATEC Web Conf. 161 03006 (2018). DOI: 10.1051/mateconf/201816103006
2. Levonevskiy D., Vatamaniuk I., Saveliev A. Providing Availability of the Smart Space Services by Means of Incoming Data Control Methods //International Conference on Interactive Collaborative Robotics. – Springer, Cham, 2018. – pp. 170-180.
3. Denisov A., Saveliev A. Comparative analysis of wireless data exchange technologies for IoT-system realization. Information technologies and systems. 2018. pp. 422-429.
4. I.V. Vatamaniuk, D.A. Malov, D.K. Levonevskii. Modeling the QoE Estimation for Services of the Cyberphysical Intelligent Space //Proceedings of the 2018 IEEE Northwest

Russia Conference on Mathematical Methods in Engineering and Technology (MMET NW). 2018. pp. 436-439.

Статьи, принятые к печати в 2018 г.:

1. Левоневский Д.К., Ватаманюк И.В., Малов Д.А. Обеспечение доступности сервисов корпоративного интеллектуального пространства посредством управления потоком входных данных // Программная инженерия. – 2018. (принято к печати).
2. Левоневский Д.К. Использование современных информационных технологий в развитии корпоративных музеев // Материалы Четвертой международной межвузовской научно-практической конференции «Технологическая перспектива: новые рынки и точки экономического роста», 2018. (принято к печати).

Оригинал-макет монографии:

И.В. Ватаманюк, Д.К. Левоневский, Д.А. Малов, Р.Н. Яковлев, А.И. Савельев. Модели и способы взаимодействия пользователя с киберфизическим интеллектуальным пространством: моногр. СПб.: Лань, 2019. (принято к печати).