

М.Э. Нагорных, А.Н. Быков, С. А. Чернышев

МУЛЬТИАГЕНТНАЯ СИСТЕМА ФОРМИРОВАНИЯ РАСПИСАНИЯ В ВУЗЕ

Аннотация. Разработана мультиагентная система поддержки принятия решения для формирования расписания в вузе. Рассмотрены аналоги приложения, выявлены их сильные и слабые стороны. Применение мультиагентного подхода позволяет составлять расписание с учетом предпочтений относительно расписания каждой группы, что достигается за счет представления каждой группы как отдельного объекта системы, – так называемого агента. В статье представлены: стек технологий разработки, схемы взаимодействия модулей и акторов системы, а также результаты работы программы на реальном расписании университета.

Ключевые слова: мультиагентная система, составление расписания, акторы, Python.

M.E. Nagornykh, A.N. Bykov, S.A. Chernyshev

MULTI-AGENT SYSTEM FOR SCHEDULING AT A UNIVERSITY

Abstract. The multi-agent decision support system for the formation of a schedule in the university was developed. During the development were considered analogs of the application, identified their strengths and weaknesses. The application of the multi-agent approach allows to create a schedule taking into account the schedule preferences of each group, this is achieved by representing each group as a separate object of the system, a so-called agent. The article contains: a stack of development technologies, schemes of interaction between modules and actors of the system, and also demonstrates the results of the program on a real university schedule.

Keywords: multi-agent system, scheduling, actors, Python.

Введение

В планировании учебного процесса важную роль играет составление расписания, которое включает в себя анализ и упорядочение больших данных автоматизированных программ. Контроль данного процесса со стороны человека имеет ряд недостатков. Во-первых, влияние человеческого фактора; во-вторых, неудобство при составлении расписания машиной, так как используемые алгоритмы не позволяют учесть множество индивидуальных ограничений и пожеланий для каждой группы в университете, а также преподавателей, например, количество первых и вечерних пар или количество учебных дней. В результате итоговое расписание зачастую требует внесения множественных правок со стороны человека для устранения нестыковок в расписании разных групп.

Цель данной работы – создание программного обеспечения поддержки принятия решения, позволяющего упростить процесс автоматизированного составления расписания для учебного заведения и минимизировать влияние человека на систему благодаря использованию мультиагентного подхода.

Мультиагентное моделирование позволяет учитывать ограничения при формировании расписания как для каждой группы, так и для преподавателей, что, в свою очередь, позволяет получить расписание, наиболее соответствующее их пожеланиям и требованиям

Нагорных Максим Эдуардович

магистрант. Санкт-Петербургский государственный университет аэрокосмического приборостроения, Санкт-Петербург. Сфера научных интересов: имитационное моделирование; мультиагентные системы; системы поддержки принятия решений; разработка распределенных систем. Автор 7 опубликованных научных работ.

Электронный адрес: nagornykh_max@mail.ru

Быков Алексей Николаевич

магистрант. Санкт-Петербургский государственный университет аэрокосмического приборостроения, Санкт-Петербург. Сфера научных интересов: имитационное моделирование; мультиагентные системы; системы поддержки принятия решений; разработка распределенных систем. Автор 7 опубликованных научных работ.

Электронный адрес: alexey_bykovoff@mail.ru

Чернышев Станислав Андреевич

кандидат технических наук, доцент кафедры интеллектуальных систем и защиты информации. Санкт-Петербургский государственный университет промышленных технологий и дизайна; доцент кафедры информатики. Санкт-Петербургский государственный экономический университет, Санкт-Петербург. Сфера научных интересов: имитационное моделирование; мультиагентные системы; системы поддержки принятия решений; разработка распределенных систем. Автор более 40 опубликованных научных работ.

Электронный адрес: chernyshev.s.a@bk.ru

учебного заведения. Также стоит отметить возможность добавления новых ограничений к системе с минимальными изменениями общего алгоритма работы.

Мультиагентная система

Мультиагентный подход подразумевает систему, состоящую из множества взаимодействующих друг с другом самостоятельных вычислительных частей, – так называемых агентов. Мультиагентные системы (далее – МАС) могут решать проблемы, которые сложно или невозможно решить монолитной системой. Каждый агент может включать методические, функциональные, процедурные подходы, алгоритмический поиск или обучение с подкреплением [4; 10].

Агенты в таких системах обладают несколькими важными характеристиками:

- Автономность – агенты частично независимы, самосознательны, автономны.
- Локальные представления – ни у одного агента нет полного представления о системе, иначе эта система слишком сложна для агента, чтобы использовать знания о ней.
- Децентрализация – ни один агент не назначается контролирующим (или система фактически сводится к монолитной системе).

Решение задач в мультиагентной системе достигается за счет распределения подзадач на каждого отдельного агента, а также возможности обмена информацией между ними.

МАС используются для графических приложений, таких как компьютерные игры, широко применяются в сетевых и мобильных технологиях для автоматической и динамической балансировки нагрузки, высокой масштабируемости и самовосстановления сетей. Кроме того, они находят применение в логистике, производстве и научных исследованиях, например, моделировании.

Обзор и анализ аналогов

Чтобы определиться с вектором разработки мультиагентной системы поддержки принятия решения при составлении расписания в учебном заведении, необходимо рассмотреть и проанализировать существующие решения, а также выявить их сильные и слабые стороны.

Для анализа выбрано три проекта, выполняющих задачу составления и оптимизации расписания: *Magellan*, «Ректор – вуз» и «1С: Автоматизированное составление расписания. Университет».

Magellan. Система *Magellan* позволяет управлять всем учебным процессом, в том числе составлением расписания в различных режимах, а также учитывать план и последовательность преподавания [2]. В программе возможно расставлять приоритеты для занятий, но при формировании расписания не будет учитываться удовлетворенность расписанием преподавателей и студентов. Работает только на системах Windows, что может вызвать трудности при ее использовании.

«Ректор – вуз». Данная программа позволяет составить расписание для учебного процесса. При составлении расписания учитываются только требуемые нагрузки и учебные планы, а удобство расписания и пожелания студентов и преподавателей не учитываются. Также в программе возможно ручное внесение изменений в расписание [3].

Программное обеспечение позволяет распределять часовую нагрузку на недельный период и в дальнейшем формировать отчеты о загруженности кафедры. При этом учитывается расписание по преподавателям, аудиториям, видам занятий и др. Кроме регулировки загруженности часов на неделю других ограничений нет. Из-за этого возможны проблемы с неравномерным распределением занятий, проведение их в неудобные часы.

Главный минус программы заключается в том, что экспортировать расписание можно только в формате docx, xls и html, что может затруднить процесс импорта расписания в другие сервисы учебного заведения. Кроме того, поддержка приложения, как и у предыдущей программы, осуществляется только в системе Windows.

«1С: Автоматизированное составление расписания. Университет». Данная программа позволяет производить составление и оптимизацию расписания в вузе. Программа учитывает учебный план и нагрузки, а также может учитывать наличие некоторых параметров, например, количество окон между парами. Однако она не позволяет учитывать индивидуальные предпочтения участников учебного процесса относительно параметров первых и последних пар, а также окон [1].

Анализ аналогов и формирование требований к разрабатываемой системе

После анализа существующих программ для составления и оптимизации расписания вуза был сделан вывод, что все рассматриваемые программы обладают большим набором функции относительно составления различных расписаний, – для сессии, учебного процесса, организации приема в вуз. Однако при составлении учебного расписания ни одна из программ не учитывает индивидуальные предпочтения участников учебного процесса, то есть учитываются только общие критерии. Кроме того, эти приложения не предоставляют возможность экспортировать полученное расписание в современные форматы данных, такие как JSON, XML и реляционные базы данных, что затрудняет его использование в других возможных приложениях учебного заведения, например, на сайте для студентов с расписанием.

Разрабатываемое программное приложение должно решать проблему с форматами и учитывать степень удовлетворенности составленным расписанием групп обучающихся и

преподавателей. Именно поэтому в проекте применяется мультиагентный подход для составления и оптимизации расписания. Его использование позволяет рассматривать каждую группу студентов и каждого преподавателя как отдельного агента со своими требованиями, ограничениями, учитывать их индивидуальный критерий удовлетворенности.

Используемый стек технологий и архитектура приложения

Для решения поставленной задачи был выбран язык Python из-за его возможности быстрого создания прототипа приложения, кроссплатформенности, широкого набора модулей, позволяющих реализовать мультиагентный подход [6].

Поскольку расписание в вузе имеет много данных и связей, например, группа – преподаватель, группа – предмет, предмет – аудитория, было принято решение использовать две системы управления базами данных – SQLite [5] и Neo4j [8]. SQLite выбрана для хранения готового расписания для каждой из групп, а Neo4j – для хранения зависимостей между объектами предметной области, что позволяет увеличить скорость ее работы.

Для реализации мультиагентной системы на языке программирования Python использовался модуль The Spian, позволяющий создавать параллельные, распределенные и отказоустойчивые системы на основе модели актора [9]. Графический пользовательский интерфейс был реализован средствами библиотеки PyQt [7].

Модули приложения. Разработанное программное обеспечение состоит из девяти основных модулей, каждый из которых решает определенные задачи.

- Environment – предназначен для инициализации всех участников системы и параллельного запуска мультиагентной системы, а также для установления параметров работы системы, таких как количество итераций работы и минимальный уровень удовлетворенности группы;
- Actorbase – описывает стандартные интерфейсы взаимодействия акторов друг с другом и их методы;
- Group_actor – модуль, наследуемый от модуля Actorbase и реализующий его методы;
- Buffer – выполняет обработку взаимодействия акторов и принимает решения о замене пар в расписании;
- StateMachine – реализует паттерн машины состояний для агента группы и буфера;
- Messages – описывает возможные сообщения между группами и их сигнатуру;
- Database – позволяет выполнять взаимодействие с базой данных SQLite посредством заготовленных допустимых запросов;
- Neo4j – позволяет выполнять взаимодействие с базой данных Neo4j посредством заготовленных допустимых запросов;
- Logger – позволяет вести журнал обо всех происходящих событиях в системе для их дальнейшего анализа.

На Рисунке 1 представлена структурная схема взаимодействия основных компонентов системы.

Описание акторов системы. В системе создается количество акторов, равное количеству групп в вузе. Каждый актор хранит в себе значения коэффициентов, отражающих предпочтения студентов относительно наличия у них количества первых пар, количества поздних пар, количества учебных дней и окон, а также свое начальное расписание и состояние в системе.

Актор может быть в 6 состояниях: инициализация, старт, анализ своей удовлетворенности, поиск возможных замен, попытка замены пары, ожидание ответа от буфера и состояние остановки.

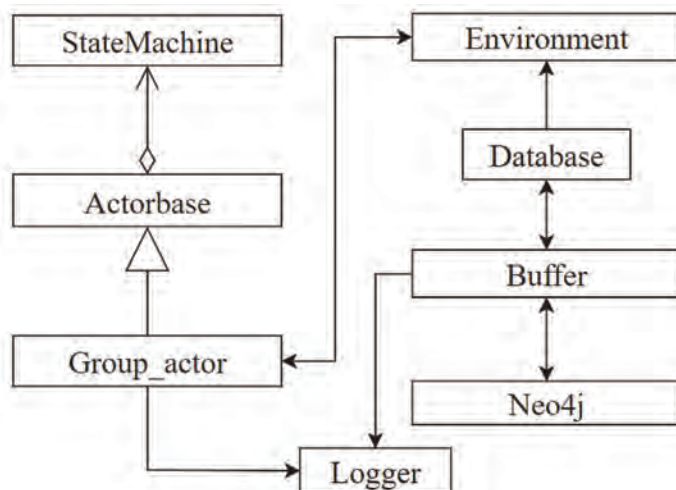


Рисунок 1. Структурная схема взаимодействия модулей

Кроме того, в системе присутствуют акторы преподавателей, служащие ограничителем при замене пары, – для каждого преподавателя устанавливаются рабочие дни и пары. Без подтверждения возможности проведения пары со стороны преподавателя ни одна замена не будет принята.

Схема взаимодействия акторов системы представлена на Рисунке 2.

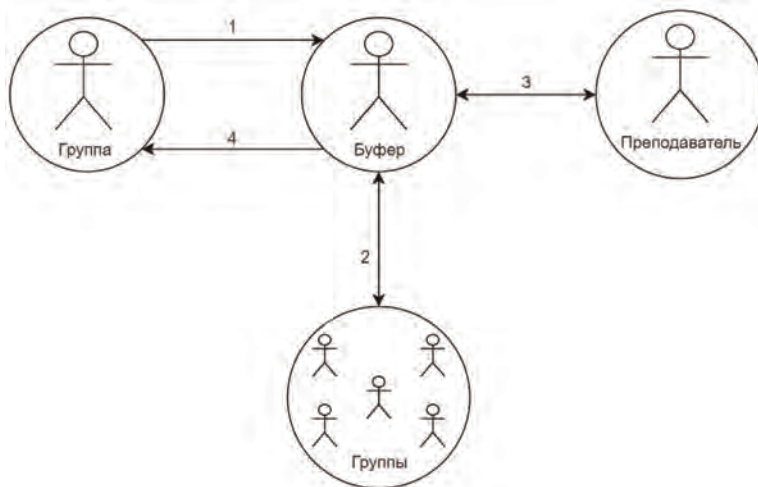


Рисунок 2. Схема взаимодействия акторов

Из рисунка видно, что общий алгоритм взаимодействия состоит из 4 основных стадий.

1. Группа, проанализировав свое расписание, посылает буферу лучшую замену на текущей итерации работы.

2. Буфер, получив информацию о замене, проверяет наличие пар в день переносимой пары, запрашивая эту информацию у других групп; в случае пересечения пары с другой группой происходит попытка обмена парами между группами.

3. Если возможность замены была подтверждена группами, происходит согласование с преподавателем о замене.

4. Результат о попытке замены пары отправляется группе.

Описание сообщений в МАС. Для взаимодействия акторов друг с другом и системой имеется 6 основных типов сообщений:

- об инициализации актора, которое служит для передачи актору группы начальных данных;
- для запуска актора и начала его работы;
- для проверки наличия в своем расписании той или иной пары, которое служит для проверки группами свободной пары для замены в своем расписании;
- сообщение о замене пары, подтверждающее возможность переноса пары на другое время;
- сообщение о результатах замены конфликтной пары, например, когда группы могут обмениваться временем проведения пары для улучшения удовлетворения обоих;
- сообщение об остановке актора.

Демонстрация работы приложения

Интерфейс приложения представлен на Рисунке 3.

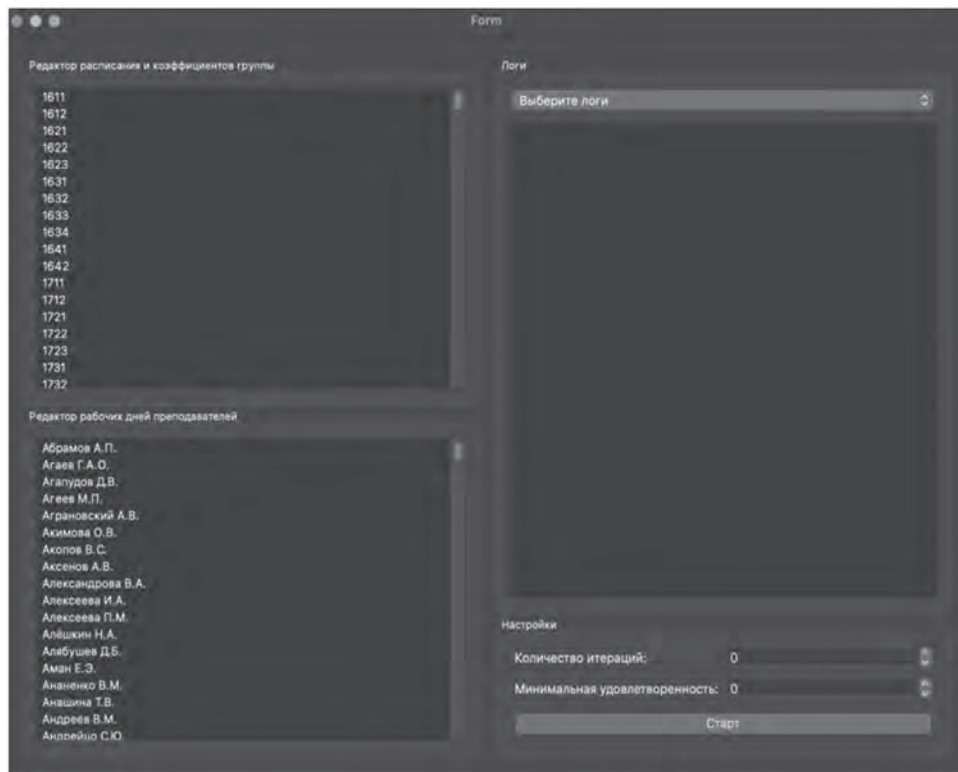


Рисунок 3. Интерфейс разработанного приложения

Как видно из рисунка, пользователю предоставляется интерфейс для настройки параметров программы, таких как количество итераций выполнения программы и минималь-

ная удовлетворенность для каждой из групп и преподавателей, при которой выполнение программы также будет завершено.

Также пользователь имеет возможность настроить коэффициенты удовлетворенности для каждой группы и редактировать рабочие дни преподавателя. Для настройки коэффициентов группы пользователь выбирает требуемую группу из списка, после чего появляется новое окно для редактирования коэффициентов, которое представлено на Рисунке 4.

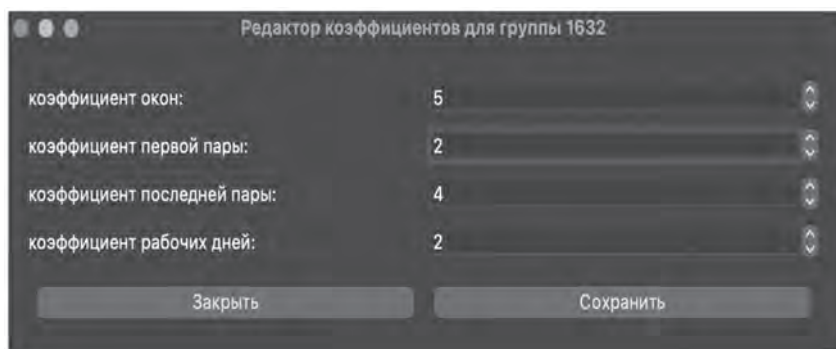


Рисунок 4. Окно настройки параметров для группы

Для настройки рабочих дней преподавателя требуется также выбрать требуемый пункт из списка, после чего появляется окно для редактирования рабочих дней (см. Рисунок 5). Следует отметить наличие возможности добавления рабочих дней и их удаления.

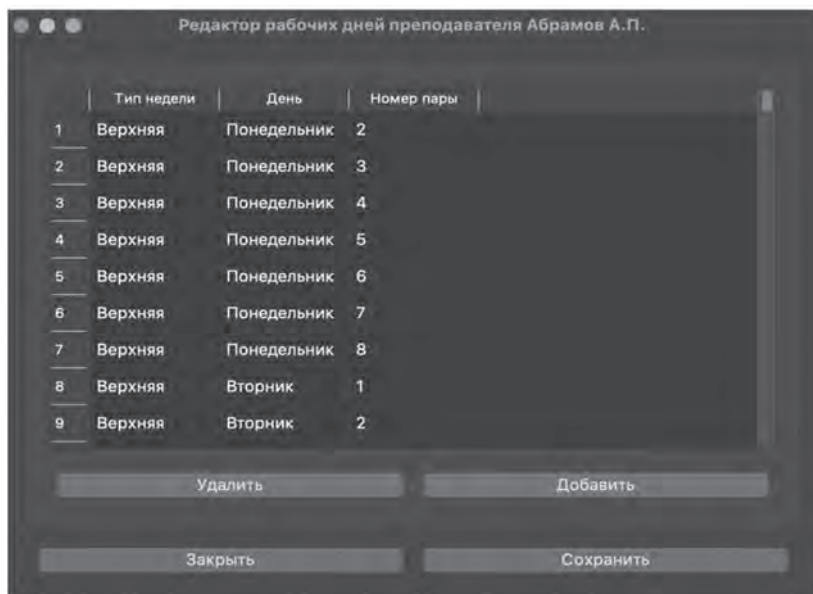


Рисунок 5. Окно настройки рабочих дней преподавателя

Для отслеживания работы приложения для каждого актора в системе создается файл логирования, который можно посмотреть в интерфейсе программы, выбрав требуемый файл в соответствующем пункте.

Для проверки работы программы было взято расписание Санкт-Петербургского государственного университета аэрокосмического приборостроения.

На Рисунке 6 представлено исходное расписание для группы, а на Рисунке 7 – расписание по итогам работы мультиагентной системы. Оно содержит меньше рабочих дней и вечерних пар – субботние пары ушли, как и поздняя пара в пятницу. На вход системы подавалось 457 групп с их расписанием за семестр, число итераций устанавливалось 4, а время работы – 5 минут.

week	day_of_week	pair_...	pair_...	pair_name	building	audience	teacher
Нижняя	Вторник	5	ЛР	Оптические системы связи	Б.М.	32-02	Пресленев Л.Н.
Нижняя	Вторник	6	Л	Оптические системы связи	Б.М.	51-06-03	Пресленев Л.Н.
Нижняя	Среда	1	Л	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Нижняя	Среда	2	ЛР	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Нижняя	Среда	3	ЛР	Оптические системы связи	Б.М.	51-06-03	Казakov В.И.
Нижняя	Среда	4	ЛР	Теория и проектирование акустооптоэлектронных ус...	Б.М.	51-06-05	Петров П.Н.
Нижняя	Четверг	4	ЛР	Технология производства лазерных систем	Б.М.	13-17	Лебедев В.Ф.
Нижняя	Четверг	5	ЛР	Технология производства лазерных систем	Б.М.	51-06-03	Лебедев В.Ф.
Нижняя	Пятница	5	Л	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.
Нижняя	Суббота	5	ЛР	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.
Верхняя	Пятница	5	ЛР	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.
Верхняя	Понедельник	2	ЛР	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Верхняя	Вторник	1	Л	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Вторник	2	ЛР	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Вторник	3	ЛР	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Вторник	4	ЛР	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Среда	2	ЛР	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Верхняя	Среда	3	ЛР	Оптические системы связи	Б.М.	51-06-03	Казakov В.И.
Верхняя	Среда	4	ЛР	Теория и проектирование акустооптоэлектронных ус...	Б.М.	51-06-05	Петров П.Н.
Верхняя	Четверг	4	ЛР	Технология производства лазерных систем	Б.М.	13-17	Лебедев В.Ф.
Верхняя	Четверг	5	Л	Технология производства лазерных систем	Б.М.	51-06-03	Лебедев В.Ф.
Верхняя	Пятница	3	ЛР	Теория и проектирование акустооптоэлектронных ус...	Б.М.	13-15	Петров П.Н.
Верхняя	Пятница	4	Л	Теория и проектирование акустооптоэлектронных ус...	Б.М.	51-06-05	Петров П.Н.
Верхняя	Пятница	6	ЛР	Проектирование лазерных технологических комплек...	Б.М.	14-06	Новиков А.Е.
Верхняя	Суббота	5	ЛР	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.

Рисунок 6. Исходное расписание группы

week	day_of_week	pair_...	pair_...	pair_name	building	audience	teacher
Нижняя	Вторник	5	ЛР	Оптические системы связи	Б.М.	32-02	Пресленев Л.Н.
Нижняя	Вторник	6	Л	Оптические системы связи	Б.М.	51-06-03	Пресленев Л.Н.
Нижняя	Среда	1	Л	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Нижняя	Среда	2	ЛР	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Нижняя	Среда	3	ЛР	Оптические системы связи	Б.М.	51-06-03	Казakov В.И.
Нижняя	Среда	4	ЛР	Теория и проектирование акустооптоэлектронных ус...	Б.М.	51-06-05	Петров П.Н.
Нижняя	Четверг	4	ЛР	Технология производства лазерных систем	Б.М.	13-17	Лебедев В.Ф.
Нижняя	Четверг	5	ЛР	Технология производства лазерных систем	Б.М.	51-06-03	Лебедев В.Ф.
Нижняя	Пятница	5	Л	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.
Верхняя	Понедельник	3	ЛР	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.
Верхняя	Пятница	5	ЛР	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.
Верхняя	Понедельник	2	ЛР	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Верхняя	Вторник	1	Л	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Вторник	2	ЛР	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Вторник	3	ЛР	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Вторник	4	ЛР	Лазерные системы локаций и навигации	Гаст.	14-02	Митко А.В.
Верхняя	Среда	2	ЛР	Лазерные информационные системы координатных апп...	Б.М.	51-06-03	Шакин О.В.
Верхняя	Среда	3	ЛР	Оптические системы связи	Б.М.	51-06-03	Казakov В.И.
Верхняя	Среда	4	ЛР	Теория и проектирование акустооптоэлектронных ус...	Б.М.	51-06-05	Петров П.Н.
Верхняя	Четверг	4	ЛР	Технология производства лазерных систем	Б.М.	13-17	Лебедев В.Ф.
Верхняя	Четверг	5	Л	Технология производства лазерных систем	Б.М.	51-06-03	Лебедев В.Ф.
Верхняя	Пятница	3	ЛР	Теория и проектирование акустооптоэлектронных ус...	Б.М.	13-15	Петров П.Н.
Верхняя	Пятница	4	Л	Теория и проектирование акустооптоэлектронных ус...	Б.М.	51-06-05	Петров П.Н.
Верхняя	Понедельник	4	ЛР	Проектирование лазерных технологических комплек...	Б.М.	14-06	Новиков А.Е.
Верхняя	Понедельник	5	ЛР	Лазерные системы специального назначения	Гаст.	14-02	Рыжиков М.Б.

Рисунок 7. Расписание группы после завершения работы программы

При запуске большого числа итераций время выполнения и количество внесенных изменений увеличивается.

Заключение

В ходе работы были проанализированы существующие решения для автоматизации процесса составления расписания учебного заведения. Сделан вывод, что существующие программные решения хотя и обладают большим набором функции относительно составления различных расписаний для сессии, учебного процесса и организации приема в вуз, однако при составлении учебного расписания ни одна из программ не позволяет учитывать индивидуальные предпочтения групп и преподавателей, то есть учитываются только общие критерии. В связи с этим было решено использовать мультиагентное моделирование, что позволило учитывать индивидуальные ограничения при формировании расписания для каждого участника учебного процесса.

В целом разработанное приложение справляется с задачей, что видно из примеров. Для каждой группы и преподавателя оно формирует расписание относительно их индивидуальных приоритетов. Кроме того, разработанное приложение, в основе которого лежит мультиагентный подход, с минимальными изменениями можно применять для других предметных областей, например, составления расписания в спортзале, графика смен на предприятии и др.

Литература

1. Программа «1С: Автоматизированное составление расписания. Университет» [Электронный ресурс]. URL: <https://solutions.1c.ru> (дата обращения: 01.01.2022).
2. Программа Magellan [Электронный ресурс]. URL: <https://magellanius.ru> (дата обращения: 01.01.2022).
3. Программа «Ректор – вуз» [Электронный ресурс]. URL: <http://rector.spb.ru> (дата обращения: 01.01.2022).
4. Hu J., Bhowmick P., Jang I., Arvin F., Lanzon A. (2021) A Decentralized Cluster Formation Containment Framework for Multirobot Systems. *IEEE Transactions on Robotics*.
5. Kreibich Jay A. (2010) Using SQLite. O'Reilly Media.
6. Mark Reed (2020) Python Programming. Publishing Factory LLC.
7. Mark Summerfield (2015) Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming, Pearson.
8. Rik Van Bruggen (2014) Learning Neo4j. Packt Publishing.
9. The Spian. Официальный сайт [Электронный ресурс]. URL: <https://thespianpy.com> (дата обращения: 15.01.2022).
10. Williams A.B. (2004) Learning to share meaning in a multi-agent system. *Autonomous Agents and Multi-Agent Systems*, No. 8 (2), pp. 165–193.

References

1. *Programma «1S: Avtomatizirovannoe sostavlenie raspisanija. Universitet»* [Programma "1C: Automated scheduling. University"]. Available at: <https://solutions.1c.ru> (date of the application: 01.01.2022) (in Russian).
2. *Programma «Magellan»* [Programma "Magellan"]. Available at: <https://magellanius.ru> (date of the application: 01.01.2022) (in Russian).

3. *Programma «Rector – vuz»* [Programma "Rector – university"]. Available at: <http://rector.spb.ru/> (date of the application: 01.01.2022) (in Russian).
4. Hu J., Bhowmick P., Jang I., Arvin F., Lanzon A. (2021) A Decentralized Cluster Formation Containment Framework for Multirobot Systems. *IEEE Transactions on Robotics*.
5. Kreibich Jay A. (2010) *Using SQLite*. O'Reilly Media.
6. Mark Reed (2020) *Python Programming*. Publishing Factory LLC.
7. Mark Summerfield (2015) *Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming*, Pearson.
8. Rik Van Bruggen (2014) *Learning Neo4j*. Packt Publishing.
9. The Spian. Available at: <https://thespianpy.com> (date of the application: 15.01.2022).
10. Williams A.B. (2004) Learning to share meaning in a multi-agent system. *Autonomous Agents and Multi-Agent Systems*, No. 8 (2), pp. 165–193.