

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

На правах рукописи

Репп Владислав Алексеевич

Моделирование автоматизированных систем составления расписания

Направление подготовки 01.04.02 «Прикладная математика и информатика»

АВТОРЕФЕРАТ
МАГИСТЕРСКОЙ ДИССЕРТАЦИИ

Работа выполнена в ФГБОУ ВО
«Комсомольский-на-Амуре государственный университет»

Научный руководитель: кандидат физико-математических наук,
доцент, зав. кафедрой «Прикладная
математика»
Григорьева Анна Леонидовна

Рецензент: кандидат физико-математических наук,
доцент кафедры информационной
безопасности, информационных систем и
физики ФГБОУ ВО «АмГПУ»
Анисимов Антон Николаевич

Защита состоится 19 июня 2026 года в 9 часов 0 мин на заседании государственной экзаменационной комиссии по направлению подготовки 01.04.02 «Прикладная математика и информатика» в Комсомольском-на-Амуре государственном университете по адресу: 681000, г. Комсомольск-на-Амуре, пр. Ленина, 27, ауд. 203/5.

Автореферат разослан 5 июня 2026 г.

Секретарь ГЭК

Д.В. Чернышова

ОБЩАЯ ХАРАКТЕРИСТИКА ДИСЕРТАЦИОННОЙ РАБОТЫ

Актуальность исследования определяется практической потребностью образовательных организаций в оптимизации процессов составления расписания с учетом большого числа установленных требований к нему.

Цель работы заключается в разработке математической модели, алгоритма и программного комплекса автоматизированного построения расписания учебных занятий в вузе с учетом временных, ресурсных, организационных и предпочтительных ограничений, устанавливаемых пользователем.

Для достижения поставленной цели решаются следующие *задачи*:

1. Анализ существующих подходов к автоматизированному составлению расписаний учебных занятий в образовательных организациях.
2. Формализация задачи, определение основных параметров.
3. Разработка математической модели и алгоритмической схемы автоматизированного формирования расписания.
4. Разработка программного модуля.
5. Анализ полученных результатов.

Объектом исследования является процесс планирования и автоматизированного построения расписания учебных занятий для образовательных организаций.

Предметом исследования выступают математические модели, алгоритмы дискретной оптимизации, структуры данных и программные средства, обеспечивающие формирование расписания с учетом ограничений и критериев качественного типа.

Научная новизна

Предлагается математическая модель автоматизированного построения расписания с учетом большого количества требований различного типа, конфигурация которых определяется пользователем.

Достоверность и обоснованность результатов исследования.

Достоверность результатов обеспечивается валидацией исходных данных, а также набором тестов для проверки качества решения.

Практическая значимость

Практическая значимость заключается в создании программного комплекса, позволяющего автоматизировать трудоемкий процесс составления учебного расписания с большим количеством ограничений.

Структура и объем.

Магистерская диссертация состоит из введения, пяти глав, заключения и списка литературы. Объем работы - 74 страницы, в том числе 24 рисунков и 5 таблиц.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении раскрывается актуальность темы, определяются цели и задачи исследования, объект, предмет, указываются научная новизна, практическая значимость, достоверность и обоснованность результатов исследования.

Первая глава содержит теоретические основы моделирования и оптимизации расписания для образовательных организаций, приводится обзор существующих результатов исследований. Отмечаются основные проблемы автоматического составления расписания в образовательных организациях, а также причины высокой вычислительной сложности. Рассматриваются критерии качества расписания. Расписание представляется как сложная ресурсно-временная система. Главный элемент – учебное занятие. Вводятся основные объекты, такие как: группы, преподаватели, аудитории, временные слоты. Описываются жесткие ограничения, накладываемые на основные ресурсы, а также мягкие ограничения, задаваемые пользователем программного комплекса и напрямую влияющие на качество составленного расписания.

Во второй главе разрабатывается математическая модель построения расписания. Модель соответствует большому количеству требований соответствующих специфики образовательных организаций.

Рассматриваются элементы модели, принадлежащие следующим множествам:

$g \in G$ – множество учебных групп;

$t \in T$ – множество преподавателей;

$r \in R$ – множество аудиторий;

$u \in U$ – множество учебных требований;

$e \in E$ – множество событий расписания.

Учебные требования представляют систему ограничений, включающую сведения о том, какая дисциплина должна быть проведена, для каких учебных

групп она предназначена, в каком объеме должна быть реализована, какими преподавателями может обеспечиваться и при каких условиях допускается ее проведение. Однако непосредственно в расписание помещается не требование, а событие.

Событие расписания – это одно конкретное занятие, которое должно быть поставлено в расписание ровно один раз.

$$E = \bigcup_{u \in U} E(u),$$

где $E(u)$ – множество событий, порожденных требованием u .

Временные параметры модели определяются:

$W = \{odd, even\}$ – множество типов недель;

D – множество учебных дней;

P – множество номеров пар внутри дня.

Полное множество временных позиций задается как

$$S = W \times D \times P$$

Каждый элемент $s \in S$ – это один конкретный слот расписания, например «четная неделя, вторник, 3-я пара».

Для каждого события $e \in E$ задаются:

$G(e) \subset G$ – группы, участвующие в занятии;

$T(e) \subset T$ – допустимые преподаватели;

$\delta(e)$ – длительность занятия в числе слотов;

$n(e)$ – число обучающихся;

$F(e) \subset S$ – запрещенные слоты;

$\rho(e)$ – требования к аудитории.

Для каждого события $e \in E$ строится множество допустимых размещений $A(e)$. Элемент $a \in A(e)$ задает одно возможное размещение.

Переменные определяют

$$x_{e,a} = \begin{cases} 1, & \text{если для события } e \text{ выбрано размещение } a, \\ 0, & \text{иначе.} \end{cases}$$

$$z_{e,r} = \begin{cases} 1, & \text{если событию } e \text{ назначена аудитория } r, \\ 0, & \text{иначе.} \end{cases}$$

В модели задаются ограничения двух типов, жесткие и мягкие. К жестким ограничениям относятся:

- уникальность события, состоящая в том, что каждое событие должно быть назначено ровно в одно допустимое размещение:

$$\sum_{a \in A(e)} x_{e,a} = 1, \forall e \in E;$$

- ограничение непересечения занятий учебной группы, состоящее в том, что одна и та же группа не может участвовать в двух событиях одновременно:

$$\sum_{e: g \in G(e)} \sum_{a \in A(e): s \in \Sigma(a)} x_{e,a} \leq 1, \forall g \in G, \forall s \in S.$$

Для события e определяется множество допустимых аудиторий $R(e) \subseteq R$.

- требование однозначного соответствия аудитории, состоящее в том, что каждому событию должна быть назначена ровно одна допустимая аудитория

$$\sum_{r \in R(e)} z_{e,r} = 1, \forall e \in E;$$

- требование однозначного соответствия занятости аудитории временному интервалу, состоящее в том, что в каждый момент времени одной аудитории может соответствовать не более одного события

$$\sum_{e: r \in R(e), s \in \Sigma^*(e)} z_{e,r} \leq 1, \forall r \in R, \forall s \in S.$$

где $\Sigma^*(e)$ – множество временных позиций, занятых событием e после завершения первого этапа.

Мягкие ограничения обеспечивают повышение качества формируемого расписания и в практической реализации задаются пользователем. Для этого в программном комплексе предусмотрена возможность выбора учитываемых правил и задания их приоритетов, выражаемых через весовые коэффициенты в целевой функции:

- Минимизация поздних занятий. Чем позже занятие поставлено в течение дня, тем больший штраф оно получает.

- Минимизация разрывов в расписании учебных групп. Штраф возникает, если у группы между двумя занятиями есть один или несколько пустых слотов.

В общем виде целевая функция этапа расстановки событий имеет вид:

$$F_{course}(x) = \sum_i w_i \Phi_i(x) \rightarrow \min,$$

где $w_i > 0$ – веса мягких ограничений.

Для этапа расстановки аудиторий по событиям используется функция

$$F_{room}(z) = \sum_i \lambda_i \vartheta_i \rightarrow \min.$$

Оптимизация мягких ограничений организована поэтапно: на первом шаге формируется допустимое решение, после чего последовательно выполняется улучшение отдельных показателей качества расписания.

В третьей главе описывается алгоритм построения расписания на основе построенной математической модели. На первом этапе выполняется проверка корректности, полноты и согласованности исходных данных. Одновременно проверяется логическая непротиворечивость ограничений и возможность преобразования каждого учебного требования в допустимое событие, имеющее хотя бы одно возможное размещение. После этого исходные данные преобразуются во внутреннюю форму, удобную для вычислений. Далее формируется множество событий расписания. Каждое учебное требование разворачивается в одно или несколько атомарных событий, которые в дальнейшем рассматриваются как самостоятельные единицы планирования. На следующем этапе для каждого события формируется множество допустимых размещений. Сохраняются только те варианты, которые удовлетворяют локальным ограничениям. В результате для каждого события строится конечное множество допустимых размещений. После этого строится расписание в виде временных слотов, заполняемых событиями. Для каждого события выбирается одно допустимое размещение при соблюдении всех глобальных ограничений. Сначала строится допустимое решение, удовлетворяющее жестким ограничениям. Затем выполняется последовательное улучшение решения с учетом мягких ограничений. После определения времени проведения событий и назначения преподавателей выполняется отдельный этап распределения аудиторий. Каждому событию ставится в соответствие аудитория. В результате формируется полное расписание, содержащее время, преподавателя и аудиторию для каждого

события. На заключительном этапе формируется итоговый результат и производится проверка качества. Определяются показатели полноты построенного расписания и вычисляются основные метрики его качества. Общая схема алгоритма представлена на рисунке 2.

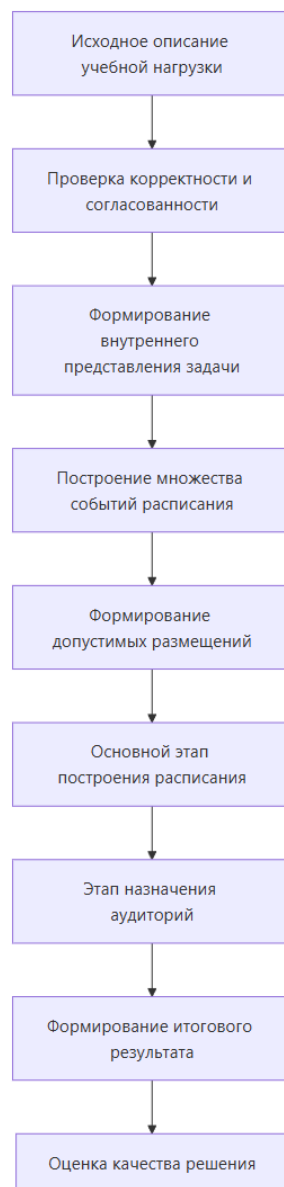


Рисунок 2 – Общая схема алгоритма

В четвертой главе рассматривается программная реализация алгоритма автоматизированного построения расписания учебных занятий.

Программный модуль включает четыре базовые составляющие: серверная часть, пользовательский интерфейс, локальное хранилище данных и настольную оболочку. Хранение наборов данных, задач расчета и готовых расписаний выполняется в локальной базе данных.

Каждый этап вычислительного процесса выносится в отдельный модуль.

Интерфейс программы позволяет загружать и выбирать наборы исходных данных, запускать построение расписания с указанием временного ограничения, открывать ранее сохраненные результаты и выполнять их последующий анализ. Общий вид главного окна и основные области работы пользователя представлены на рисунке 5.1.

В программе предусмотрен просмотр уже рассчитанных расписаний без необходимости повторного запуска вычислений. Для выбранного результата отображаются сводные показатели, табличное представление расписания и дополнительные диагностические сведения. Пример отображения рассчитанного решения приведен на рисунке 5.2.

В процессе построения расписания пользователь может наблюдать текущее состояние вычислений. В окне программы отображаются этап выполнения, статус задачи и степень завершенности расчета. Пример промежуточного состояния решения показан на рисунке 5.3.

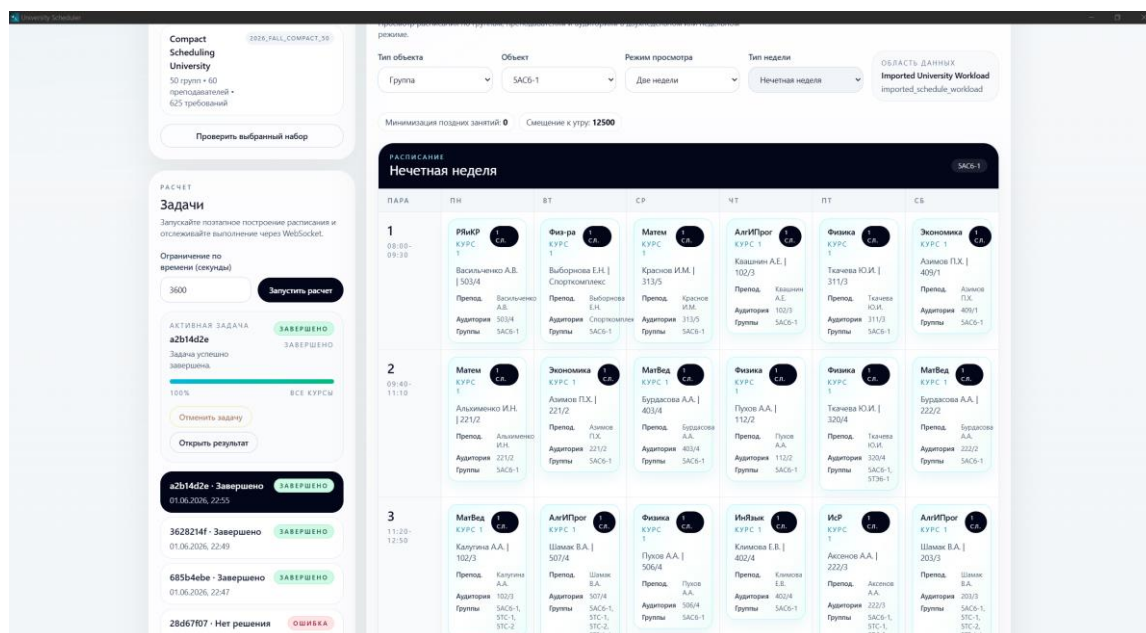


Рисунок 4.1 – Главное окно программного комплекса

После завершения расчета пользователь получает доступ к готовому расписанию. Для удобства анализа результат в интерфейсе представлен в двух

основных формах: в виде сводных карточек и в виде общей таблицы расписания.

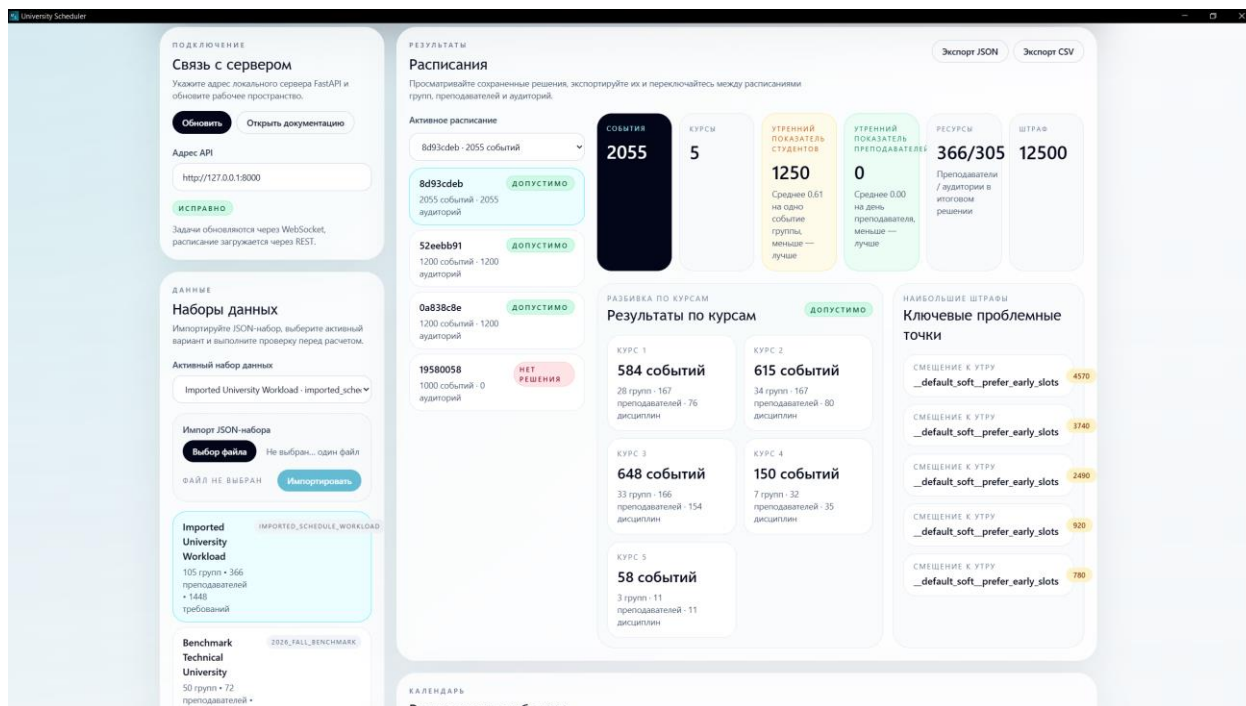


Рисунок 4.2 – Информация о исходных данных и рассчитанном расписании

Таблица может строиться для различных типов объектов: для учебной группы, для преподавателя и для аудитории.

После завершения вычислений задача получает конечный статус, а результат автоматически становится доступным в списке сохраненных расписаний.

В пятой главе проводится оценка результатов работы программного комплекса.

Оценка результатов проводилась на наборе данных, который содержит 105 учебных групп, 366 преподавателей, 312 аудиторий, 1448 учебных требований и 2055 событий расписания.

Результаты сравнения представлены в таблице 5.1.

Таблица 5.1 – Результаты сравнения

Показатель	Программный комплекс	Реальное расписание
Количество окон в расписании	0	169
Средний недельный дисбаланс	1.171	2.310
Средний дневной дисбаланс	0.843	1.740

Показатель	Программный комплекс	Реальное расписание
Средний номер стартовой пары	0,150	1.578
Средний резерв вместимости аудитории	67.4	79.5

Средний недельный дисбаланс для каждой группы вычисляется как абсолютная разность между числом занятых временных слотов в четной и нечетной неделях, после чего полученные значения усредняются по всем группам.

Средний дневной дисбаланс для каждой группы вычисляется как разность между числом занятых временных слотов в наиболее загруженный и наименее загруженный день, после чего полученные значения усредняются по всем группам.

Средний резерв вместимости аудитории для каждого занятия вычисляется как разность между вместимостью назначенной аудитории и числом студентов, посещающих занятие, после чего полученные значения усредняются по всем занятиям.

В заключении приводятся основные результаты работы. Дается оценка полученных результатов, в сравнении с существующими подходами и получаемыми результатами на их основе.

СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ИССЛЕДОВАНИЯ

1 Репп, В.А. Математическая модель составления расписания учебных занятий в университете на основе бинарного целочисленного программирования/ В.А. Репп, А.Л. Григорьева // материалы 9-й всероссийской национальной научной конференции молодых учёных. 2026. – С. 343-346.

2 Репп, В.А. Обзор подходов к постановке задачи корректировки расписания в учебных заведениях/ В.А. Репп, А.Л. Григорьева // материалы 9-й всероссийской национальной научной конференции молодых учёных. 2026. – С. 346-349.