

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование)

02.03.03 Математическое обеспечение и администрирование информационных систем

(код и наименование направления подготовки, специальности)

WEB-дизайн и мультимедиа

(направленность (профиль))

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВАРСКАЯ РАБОТА)

на тему «Анализ и оптимизация сетевых графиков в условиях неопределенности»

Студент

М.В. Севрюгин

(И.О. Фамилия)

(личная подпись)

Руководитель

к.т.н., доцент, Н.А. Сосина

(ученая степень, звание, И.О. Фамилия)

Аннотация

Тема выпускной квалифицированной работы: «Анализ и оптимизация сетевых графиков в условиях неопределенности».

Целью бакалаврской работы являются анализ и оптимизация сетевых графиков в условиях неопределенности.

Актуальность бакалаврской работы заключается в разработке программы для автоматизации анализа, планирования в условиях неопределенности и оптимизации сетевого графика. Во введении раскрывается актуальность исследования и дается краткое описание проделанной работы.

В первой главе рассматриваются уже существующие методы управления проектами, благодаря анализу которых будут поставлены задачи на реализацию.

Вторая глава содержит описательный характер выбранного алгоритма, его концептуальная и техническая реализация, а также математическая постановка задачи.

В третьей главе описывается процесс разработки программного обеспечения. На предоставленном примере показан результат работы.

Результатом работы служит программный продукт, не имеющий недостатков существующих аналогов.

ABSTRACT

The topic of the present graduation work is *Analyzing and optimizing the network graphs under conditions of uncertainty*.

The goal of the investigation is to analyze and optimize the network graphs under conditions of uncertainty.

The relevance of the graduation work is to develop a programme for automating the analysis, planning under conditions of uncertainty and optimizing the network graph.

The introduction reveals the relevance of the research and provides a brief description of the work done.

The first chapter of the research considers the existing methods of project management, thanks to the analysis of which the implementation tasks are set.

The second chapter of the research describes the chosen algorithm, its conceptual and technical implementation, as well as features the mathematical problem statement.

The third chapter of the research is devoted to developing the software. The provided example shows the result of the work that is a software product. It is noted that this product does not have the disadvantages of the existing analogues.

Оглавление

Введение.....	5
Глава 1 Обзор и анализ методов сетевого планирования на графиках	7
1.1 Метод PERT	7
1.2 Диаграмма Ганта	11
1.3 Система сетевого планирования и управления.....	14
Глава 2 Методика решения задачи анализа и оптимизация сетевого графика в условиях неопределенности.....	20
Глава 3 Разработка программы анализа и оптимизации сетевых графиков	28
3.1 Обзор и анализ программного обеспечения для сетевого планирования	28
3.1.1 Программный продукт «1С: Управление проектным офисом»	28
3.1.2 Онлайн-сервис PERT Chart Generator	29
3.1.3 Онлайн-калькулятор по сетевому планированию	31
3.2 Реализация программы анализа и оптимизации сетевых графиков	33
3.3 Решение задачи анализа и оптимизации сетевых графиков.....	37
Заключение	51
Список используемой литературы и используемых источников.....	52
Приложение А Фрагмент программного кода	54

Введение

В последнее время наблюдается повышенный интерес к сетевому планированию в различных областях производственной и проектной деятельности, прежде всего в области решения задач оптимизации и планирования сетевых графиков для различных объектов - от отдельных производственных единиц до географически распределенных цепочек поставок. Как показывает практика, значительная часть задач в этой области сосредоточена на разработке детерминированных моделей, в которых предполагается, что данные о проблемах известны заранее.

В действительности, однако, может существовать неопределенность в отношении ряда факторов, таких как время обработки и стоимость.

Для решения таких задач требуется применение более сложных методов анализа и оптимизации сетевых графиков.

Таким образом, исследование методов анализа и оптимизации сетевых графиков в условиях неопределенности является актуальным и вызывает научно-практический интерес.

Объектом исследования бакалаврской работы является сетевое планирование.

Предметом исследования бакалаврской работы является сетевые графики в условиях неопределенности.

Целью выпускной квалификационной работы являются анализ и оптимизация сетевых графиков в условиях неопределенности.

Для достижения данной цели необходимо выполнить следующие задачи:

- произвести обзор и анализ методов сетевого планирования на графиках;
- описать методику решения задачи анализа и оптимизация сетевого графика в условиях неопределенности;
- разработать программу для анализа и автоматизации сетевого планирования в условиях неопределенности, а также оптимизации сетевого графика.

Методы исследования – методы сетевого планирования и разработки математического обеспечения информационных систем.

Практическая значимость бакалаврской работы заключается в разработке программы для автоматизации и анализа сетевого планирования в условиях неопределенности и оптимизации сетевого графика.

Данная работа состоит из введения, трех глав, заключения, списка используемой литературы и приложений.

Первая глава посвящена обзору и анализу методов сетевого планирования на графиках.

Во второй главе описана методика решения задачи анализа и оптимизация сетевого графика в условиях неопределенности.

В третьей главе описан процесс разработки программы автоматизации и анализа сетевого планирования в условиях неопределенности и оптимизации сетевого графика.

В заключении описываются результаты выполнения выпускной квалификационной работы.

Приложения содержат фрагменты программного кода приложения.

Бакалаврская работа состоит из 56 страниц текста с приложением, 16 рисунков, 8 таблиц и 20 источников.

Глава 1 Обзор и анализ методов сетевого планирования на графиках

«Сетевая модель или сетевой график представляет собой модель, на которой представлена последовательность выполнения всех работ и их взаимосвязи. В основу построения сети закладываются три основных понятия: работа, событие и путь.

Расчет сетевого графика заключается в определении его расчетных параметров: продолжительности критического пути; раннего начала и окончания всех работ, позднего начала и окончания работ; резервов времени.

Методы сетевого планирования применяются для оптимизации планирования и управления сложными разветвленными комплексами работ, требующими участия большого числа исполнителей и затрат ограниченных ресурсов.

Основная цель сетевого планирования - сокращение до минимума продолжительности проекта.

Задача сетевого планирования состоит в том, чтобы графически, наглядно на системном уровне отобразить и оптимизировать последовательность и взаимозависимость работ, действий или мероприятий, обеспечивающих своевременное и планомерное достижение конечных целей.

С помощью сетевого графика руководитель работ или операции имеет возможность системно и масштабно представлять весь ход работ или оперативных мероприятий, управлять процессом их осуществления, а также маневрировать ресурсами» [7].

Рассмотрим популярные методы сетевого планирования.

1.1 Метод PERT

В управлении проектами метод анализа оценки проекта, или PERT (Project Evaluation Review Technique), используется для определения времени, необходимого для завершения конкретной задачи или действия [20].

Это система, которая помогает в правильном планировании и координации всех задач в рамках проекта. Это также помогает отслеживать прогресс или отсутствие такового по проекту в целом. В 1950-х годах ВМС США разработали методику оценки проекта для управления программой создания подводных лодок «Поларис» их Управления специальных проектов.

Знание времени, необходимого для выполнения проекта, имеет решающее значение, поскольку оно помогает руководителям проектов принимать решение о других факторах, таких как бюджет и делегирование задач. Независимо от того, насколько велик или мал проект, оценки могут быть слишком оптимистичными или пессимистичными, но использование диаграммы PERT поможет определить реалистичные оценки.

Результатом сетевого планирования по методу PERT является сетевая диаграмма.

Узлы диаграммы представляют собой события, обозначающие начало или конец действий или задач. Директивные линии указывают на задачи, которые необходимо выполнить, а стрелки показывают последовательность действий.

Есть четыре определения времени, которые используются для оценки требований проекта ко времени:

- оптимистичное время (O) - наименьшее количество времени, которое может занять выполнение задачи;
- пессимистическое время (P) - максимальное время, необходимое для выполнения задачи;
- наиболее вероятное время (M) - при условии отсутствия проблем, наилучшая или наиболее разумная оценка того, сколько времени потребуется для выполнения задачи;
- ожидаемое время - при наличии проблем наилучшая оценка времени, необходимого для выполнения задачи.

Как правило, для проведения анализа по методу PERT используют трех-временные оценки (оптимистическая, пессимистическая и наиболее вероятная) для каждого действия на критическом пути.

Затем используют эти оценки в приведенной ниже формуле, чтобы рассчитать, сколько времени требуется для каждого этапа проекта [15]:

$$CP = (P+4M+O)/6 \quad (1)$$

На рисунке 1 представлены пример табличного представления сетевого графика по методу PERT.

Task	Optimistic (O)	Most Likely (M)	Pessimistic (P)
Task A	2 Wks	4 Wks	5 Wks
Task B	1 Wks	2 Wks	3 Wks
Task C	2 Wks	3 Wks	4 Wks
Task B	3 Wks	5 Wks	8 Wks
Completion	8 Wks	14 Wks	20 Wks

Рисунок 1 – Пример трех-временного графика

На рисунке 2 представлены пример узловой PERT-диаграммы критического пути (выделен пунктиром). Для данного случая критический путь равен 4 дням.

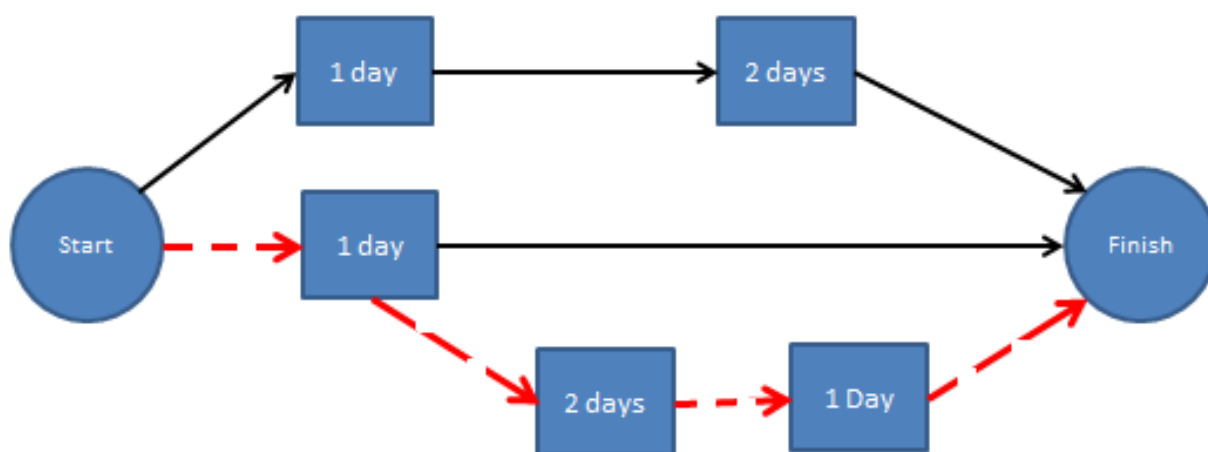


Рисунок 2 – Пример узловой PERT-диаграммы критического пути

В диаграммах PERT в основном используются следующие термины [13]:

- Float / Slack (Плавающий / Медленный) - указывает количество

времени, на которое задача может быть отложена, не вызывая общей задержки в завершении других задач или проекта;

- критический путь - указывает максимально возможный непрерывный путь от начала до конца задачи или события;
- активность критического пути - относится к деятельности без задержек;
- время выполнения - количество времени, необходимое для завершения задачи, не влияя на последующие задачи;
- время задержки - самое раннее время, когда последующее событие / задача может следовать за предыдущим событием / задачей;
- быстрое отслеживание - относится к параллельному выполнению задач или действий;
- Crashing Critical Path (Неудачный критический путь) - сокращение времени на выполнение критической задачи.

Методика построения диаграммы PERT состоит из следующих шагов:

- определите различные задачи, необходимые для завершения проекта. Обязательно добавьте их в правильном порядке и укажите продолжительность каждой задачи;
- создайте схему сети. Используйте стрелки для обозначения действий и используйте узлы для обозначения вехи (работа с нулевой продолжительностью, обозначает важное, значимое событие в проекте);
- определите критический путь и возможный резерв пути (разница между временной длительностью проекта и длиной пути на графике).

Преимущества использования метода PERT в управлении проектами:

- помогает максимально использовать ресурсы;
- делает планирование проекта более управляемым;
- полезен, даже при недостаточности или отсутствии данных о предыдущем графике;
- позволяет руководителям проектов лучше оценить или определить

более точную дату их завершения.

Недостатки метода PERT:

- сложность интерпретации метода для крупных проектов;
- сложность обновления, изменения и поддержки PERT- диаграмм;
- субъективный временной анализ мероприятий, что в некоторых случаях может повлиять на качество сетевой модели проекта.

Следует отметить, что диаграммы PERT полезны для анализа «что, если».

Понимание возможностей, касающихся потока ресурсов проекта и основных этапов, позволяет руководству достичь наиболее эффективного и полезного пути проекта.

1.2 Диаграмма Ганта

Диаграмма Ганта - это инструмент управления проектами, помогающий в планировании и составлении графиков проектов любого размера, хотя они особенно полезны для упрощения сложных проектов [19].

Сроки и задачи управления проектом преобразуются в горизонтальную линейчатую диаграмму, на которой отображаются даты начала и окончания, а также зависимости, график и сроки, в том числе, какая часть задачи выполнена на данном этапе и кто является владельцем задачи.

Это полезно для отслеживания задач, когда есть большая команда и несколько заинтересованных сторон при изменении объема работы.

Поскольку диаграмма Ганта представлена в формате гистограммы, она обеспечивает визуальное отображение:

- всего проекта;
- сроков и предельных дат выполнения всех задач;
- отношений и зависимостей между различными видами деятельности;
- фаз проекта.

Решения для управления проектами, которые интегрируют диаграммы Ганта, позволяют менеджерам видеть рабочие нагрузки команды, а также

текущую и будущую доступность, что позволяет более точно составлять график выполнения проекта.

Основная концепция диаграммы Ганта состоит в том, чтобы обозначить, какие задачи могут выполняться параллельно, а какие - последовательно.

Если объединить это с ресурсами проекта, можно установить компромисс между объемом (выполнение большего или меньшего количества работы), стоимостью (использование большего или меньшего количества ресурсов) и временными масштабами проекта.

Добавляя дополнительные ресурсы или сокращая объем, менеджер проекта может увидеть влияние на дату окончания.

Чтобы создать диаграмму, необходимо знать все отдельные задачи, необходимые для завершения проекта, оценить, сколько времени займет каждая задача и какие задачи зависят от других.

Структура и пример диаграммы Ганта представлена на рисунках 3 и 4, соответственно [1].

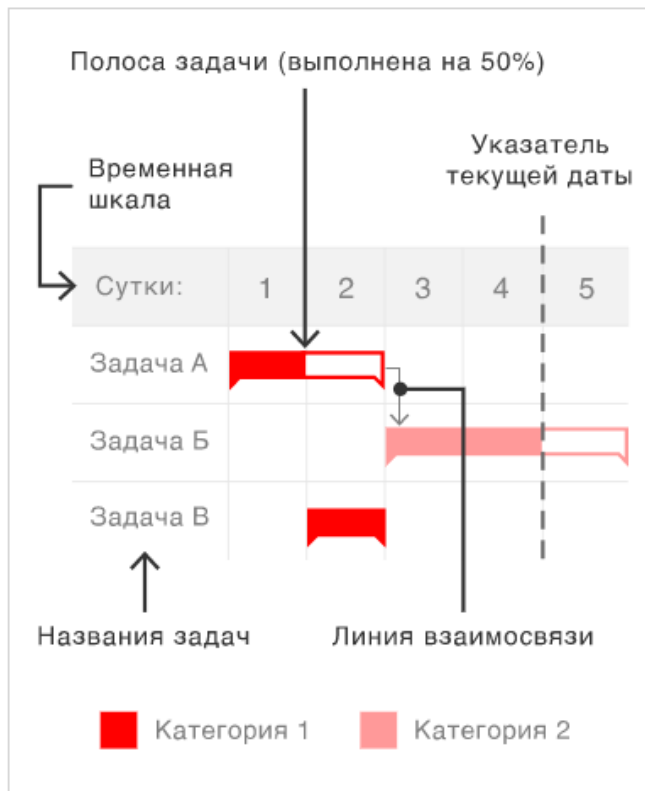


Рисунок 3 - Структура диаграммы Ганта

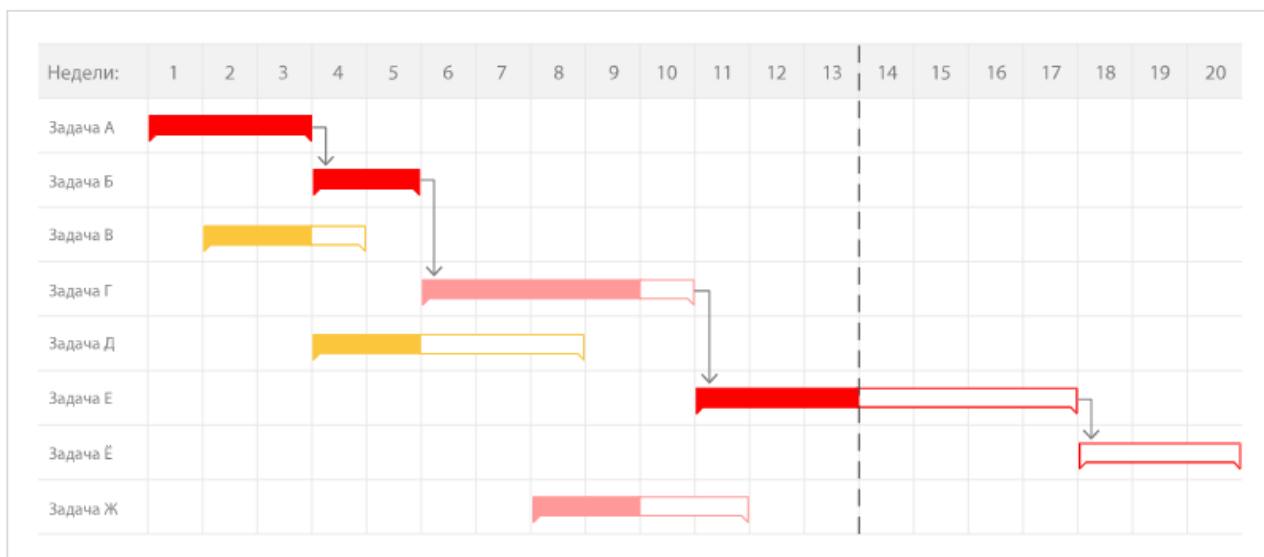


Рисунок 4 – Диаграмма Ганта

Сам процесс сбора этой информации помогает руководителю проекта сосредоточиться на основных частях проекта и начать устанавливать реалистичные временные рамки для завершения.

Недостатки диаграмм Ганта:

- они могут стать чрезвычайно сложными. За исключением самых простых проектов, для завершения проекта будет выполнено большое количество задач и задействованы ресурсы.;
- размер полосы не указывает на объем работы. Каждая полоса на диаграмме указывает период времени, в течение которого будет выполнен определенный набор задач;
- их нужно постоянно обновлять. У разработчика должна быть возможность легко и часто менять диаграмму. Для этого понадобится специализированное программное обеспечение;
- трудно разметить на одном листе бумаги. Программные продукты, которые создают эти диаграммы, обычно разбивают ее на сегменты, чтобы иметь возможность видеть весь проект.

Диаграмма Ганта используется для следующих задач:

- создания первоначальный графика проекта;
- распределения ресурсов;

- внесите корректировок в проект;
- отслеживания и установления прогресса;
- контроля и распространения графиков;
- представления ключевых событий;
- выявление и распространение проблем.

Поскольку все отображается визуально, можете сразу увидеть, что должно было быть достигнуто к определенной дате, и, если проект отстает от графика, можно принять меры, чтобы вернуть его в нормальное русло.

1.3 Система сетевого планирования и управления

Система сетевого планирования и управления (СПУ) была разработана в России на основе метода PERT.

«Основными характерными чертами системы СПУ являются обоснованность разрабатываемого плана и оперативность руководства.

Обоснованность достигается благодаря точному расчету сетевого графика, возможности его корректировки и пересмотра принятых решений с учетом фактического хода работ.

В СПУ используются следующие понятия:

- Сетевой график – наглядное отображение плана работ.
- Событие – состояние, момент достижения промежуточной или конечной цели разработки.
- Работа – протяжённый во времени процесс, необходимый для совершения события» [11].

Рассмотрим основные правила построения сетевой модели метода «вершина – работа».

1. Элементы сетевого графика включают работы и зависимости (связи) между ними. События на графике не отражаются за исключением вех, представляющих собой основные наиболее важные события, изображаемых в форме «ромбиков», работ с нулевой продолжительностью.

«2. Работа представляет собой неделимый элемент процесса, требующий времени и других ресурсов для выполнения, изображается в виде прямоугольника, вытянутого по горизонтали. Исходя из этого правила, длина прямоугольника может свидетельствовать о продолжительности операции.

3. Разработка графика начинается с размещения на нем исходной работы в крайней левой точке и завершается включением в него конечной операции, закрывающей проект. Календарно начальный момент исходного действия обозначает старт проекта.

4. Зависимости (связи) между работами оформляются стрелками, направленными слева направо под разными углами наклона. На основе данного правила связи между операциями определяются одной из форм отношений зависимости.

5. Сетевой график имеет только одну работу, в которую входят зависимости, но не выходит ни одна, и одну работу, не имеющей ни одной входящей зависимости.

6. Сетевой график не может иметь циклов, т.е. зависимости не должны связывать работы по кругу» [2].

Пример модели сетевого графика по методу «вершина - работа» представлен на рисунке 5.



Рисунок 5 - Модель сетевого графика по методу «вершина - работа»

Рассмотрим расчет сетевого графика аналитическим методом СПУ.

На рисунке 6 изображен базовый алгоритм построения сетевого графика.

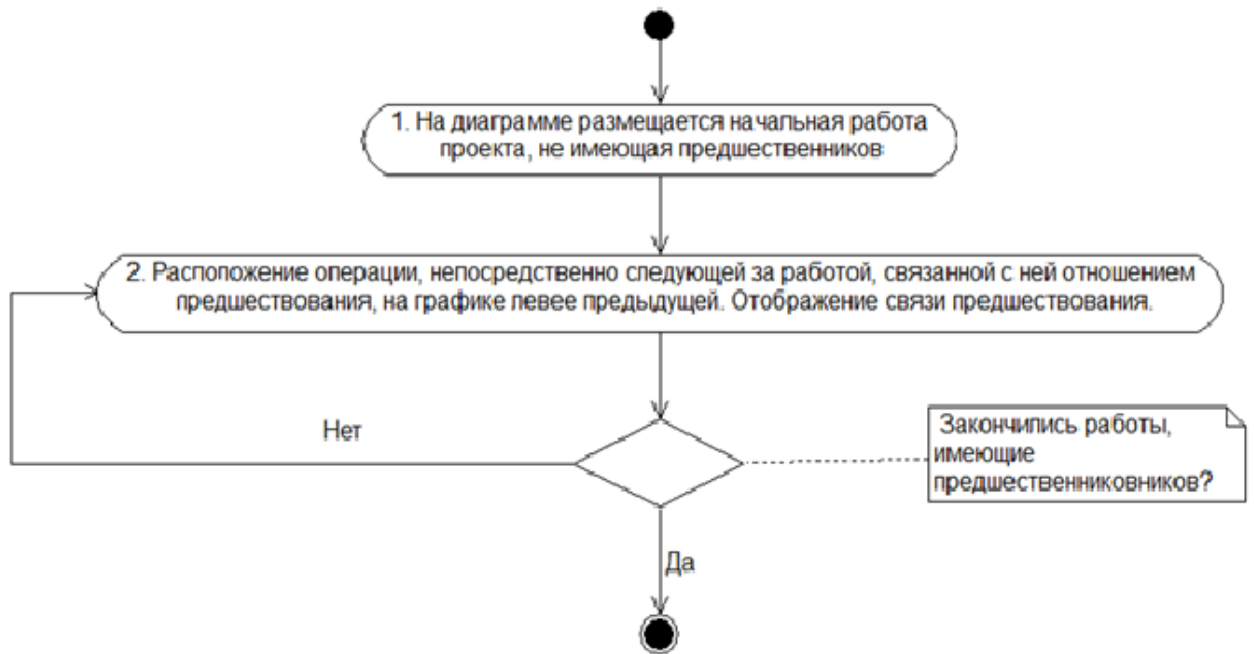


Рисунок 6 - Базовый алгоритм построения сетевого графика

Начало и окончание любой работы описываются двумя событиями, которые называются начальным и конечным событиями, соответственно (рисунок 7).

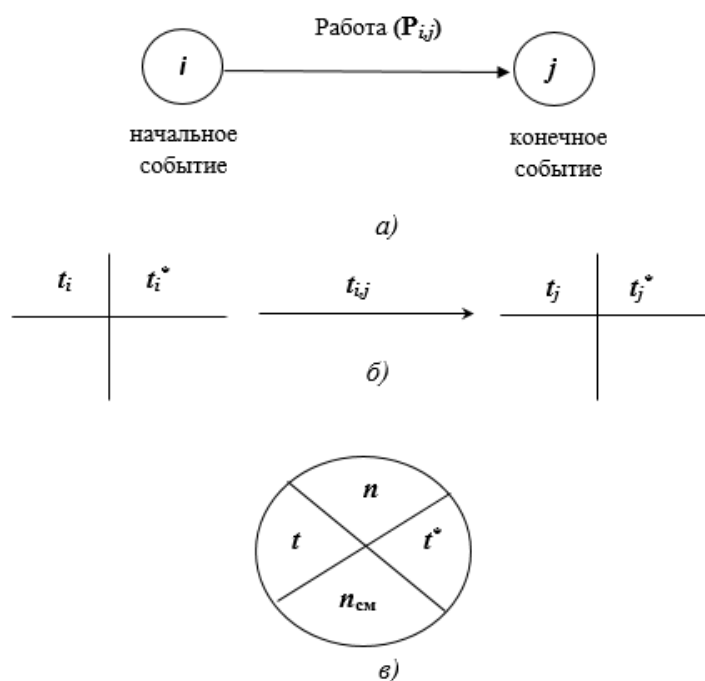


Рисунок 7 – Параметры сетевого графика

Здесь:

а) P_{ij} - работа, где i и j – номера начального и конечного события,

соответственно (рисунок 5, а).

б) t_{ij} – продолжительность работы $P_{i,j}$, t – ранний срок (ожидаемое время) возникновения события, t^* – поздний срок (предельное время) возникновения события.

в) n – номер события, $n_{см}$ – номер смежного события.

Ранний срок возникновения j -го события представляет собой момент времени, раньше которого событие произойти не может и рассчитывается по формуле:

$$t_j = \max (t_i + t_{ij}) \quad (2)$$

Поздний срок возникновения i -го события показывает максимальную задержку во времени наступления данного события:

$$t_i = \min (t_j - t_{ij}) \quad (3)$$

Критический путь – последовательность работ между начальными и конечными событиями сети, имеющих наибольшую продолжительность во времени.

Длина критического пути равна раннему сроку свершения завершающего события, т.е. $t_{кр} = t_{п} = t_{п}^*$.

Любая из работ пути L на его участке, не совпадающем с критическим путем (замкнутым между двумя событиями критического пути), обладает резервом времени.

Полный резерв времени работы $P_{i,j}$ показывает, на сколько можно увеличить время выполнения данной работы при условии, что срок выполнения комплекса работ не изменится.

Для расчета данного резерва используем следующую формулу:

$$M_{ij} = t_j^* - t_i - t_{ij} \quad (4)$$

Свободный резерв времени работы $P_{i,j}$ представляет собой часть полного резерва времени, на которую можно увеличить продолжительность работы, не

изменив при этом раннего срока ее конечного события.

Для расчета данного резерва используем следующую формулу:

$$M_{ij} = t_j - t_i - t_{ij} \quad (5)$$

Таким образом, метод СПУ позволяет рассчитывать основные параметры сетевого графика, опираясь на системный подход к вопросам организации управления.

Главным недостатком метода СПУ является его строгая линейность, инвариантность действий и последовательности их осуществления.

Для выбора метода анализа и оптимизации сетевого графика в условиях неопределенности используем таблицу 1, составленную на основе анализа источников по данной тематике.

Критерии оценивания:

- 0 – полное несоответствие требованиям;
- 1 – значительное несоответствие требованиям;
- 2 – незначительное несоответствие требованиям;
- 3 – полное соответствие требованиям.

Таблица 1 – Сравнительный анализ методов сетевого планирования

Характеристика/балл	PERT	Диаграмма Ганта	СПУ
табличное представление сетевой модели	3	1	3
графическое представление сетевой модели	2	3	2
анализ сетевой модели	3	2	3
оптимизация сетевой модели	3	2	3
простота реализации	2	2	2
Итого	13	10	13

Таким образом, по результатам сравнительного анализа выбираем в качестве метода сетевого планирования PERT и основанный на нем метод СПУ.

Выводы к главе 1

Первая глава ВКР посвящена постановке обзора и анализу методов сетевого планирования.

Результаты проделанной работы позволили сделать следующие выводы:

1. Наиболее востребованными методами сетевого планирования являются PERT, диаграмма Ганта и СПУ.

2. Основными характерными чертами системы СПУ являются обоснованность разрабатываемого плана и оперативность руководства.

3. Сравнительный анализ методов показал, что лучшими характеристиками для решения задач сетевого планирования обладают метод PERT и основанный на нем метод СПУ.

Глава 2 Методика решения задачи анализа и оптимизация сетевого графика в условиях неопределенности

На рисунке 8 изображена упрощенная блок-схема функциональных этапов управления проектом.



Рисунок 8 - Блок-схема управления проектом

Анализ сетевого графика начинается с анализа топологии сети, включающего контроль построения сетевого графика, установление целесообразности выбора работ, степени их расчленения.

Затем проводятся классификация и группировка работ по величинам резервов. Следует отметить, что величина полного резерва времени далеко не всегда может достаточно точно характеризовать, насколько напряженным является выполнение той или иной работы не критического пути.

Все зависит от того, на какую последовательность работ распространяется вычисленный резерв, какова продолжительность этой последовательности.

Для оптимизации сетевых графиков используются следующие методы:

- метод критического пути;
- анализ расписания по методу PERT.

«При использовании метод критического пути производится последовательный расчет наиболее ранних и самых поздних сроков работ по проекту. Далее устанавливается размер полного резерва, при этом критическими работами считаются действия, имеющие нулевой полный резерв.

Наконец, рассчитываются временные резервы для выполнения операций, и в качестве самого длинного пути сети выбирается критический путь. Критических путей может быть несколько.

Для оптимизации сетевого графика методом критического пути применяется следующих моделей диаграммы:

- для моделей с дискретным и непрерывным временем;
- для моделей с простым отношением предшествования;
- в сетях с обобщенными связями.

Основные отличия метода PERT от метода критического пути основаны на допущении, что длительность работ носит случайный характер, и для целей оценки и анализа сроков следует учитывать неопределенность временного параметра работ проекта.

Также делается допущение о независимости всех случайных величин

продолжительности критического пути.

Для этих целей на основе метода математической статистики и теории вероятности используется β -распределение и оценка параметров распределения с трех экспертных позиций: оптимистичной, пессимистичной и наиболее вероятной.

Для анализа и оптимизации сетевого графика используем метод PERT.

В управлении неопределенность – это состояние недостаточности знаний, при котором невозможно точно описать существующее состояние или будущие результаты, а также определить, какой из нескольких возможных результатов будет получен» [18].

Тем не менее, можно определить неопределенность, присвоив вероятность каждому возможному состоянию или исходу, чтобы оценить его вероятность.

Как показывает практика, как правило, продолжительность работы по сетевому графику заранее не известна и может принимать лишь одно из ряда возможных значений [14].

Продолжительность работы t_{ij} является случайной величиной, характеризующейся своим законом распределения, а значит, своими числовыми характеристиками – \bar{T}_{ij} средним значением, или математическим ожиданием, а также дисперсией σ^2_{ij} .

«Чем больше неопределенность в определении продолжительности выполнения каждой работы в отдельности, тем больше неопределенность и по сети в целом, т.к. дисперсия сроков наступления событий по мере продвижения по сети к конечному событию накапливается.

Чем больше объем сети и чем больше работ входит в ее состав, тем более неопределенной становится оценка срока окончания разработки» [11].

Алгоритм разработки сетевого графика в условиях неопределенности состоит из следующих шагов:

Шаг 1. Определить числовые характеристики работы P_{ij} .

Для этого предварительно проводится опрос ответственных исполнителей

и экспертов проекта, на основании которого определяются:

- оптимистическая оценка a_{ij} ;
- пессимистическую оценку b_{ij} ;
- наиболее вероятную оценку m_{ij} .

«Указанные три оценки являются основой для расчета средней ожидаемой продолжительности работы и ее дисперсии.

При этом используется гипотеза об определенном законе распределения длительностей работ или β -распределение.

Исходя из формулы (1), получим:

$$\bar{t}_{ij} = \frac{a_{ij} + 4m_{ij} + b_{ij}}{6} \quad (6)$$

$$\sigma_{ij}^2 = \left(\frac{b_{ij} - a_{ij}}{6} \right)^2 \quad (7)$$

Шаг 2. Определить общую продолжительность пути L .

Общая продолжительность пути имеет нормальный закон распределения со средним значением, равным сумме средних значений продолжительности составляющих его работ и дисперсией, равной сумме соответствующих дисперсий:

$$\bar{t}(L) = \sum_{P_{ij} \in L} \bar{t}_{ij} \quad (8)$$

$$\sigma^2(L) = \sum_{P_{ij} \in L} \sigma_{ij}^2 \quad (9)$$

Шаг 3. Оценить вероятность T .

Задача: Требуется оценить вероятность того, что срок выполнения проекта $t_{кр}$ не превзойдет заданного директивного срока T .

Рассмотрим $t_{кр}$, как случайную величину, имеющую нормальный закон распределения» [8].

Тогда получим:

$$P(t_{кр} \leq T) = \Phi \left(\frac{T - \bar{t}_{кр}}{\sigma_{кр}} \right), \quad (10)$$

где Φ - значение интеграла вероятностей Лапласа.

«Если величина $P(t_{кр} \leq T)$ мала, то опасность срыва заданного срока выполнения комплекса велика и необходимо принятие ряд дополнительных мер для выхода из ситуации (например, перераспределение ресурсов по сети, пересмотр состава работ и т. п.).

Если величина $P(t_{кр} \leq T)$ велика, то, с достаточной степенью надежности можно прогнозировать выполнение проекта в установленный срок.

Шаг 4. Проанализировать сетевой график.

Рассмотрим анализ коэффициента сложности:

$$K_{сл} = n_{раб} / n_{соб}, \quad (11)$$

где:

$n_{раб}$ - количество работ;

$n_{соб}$ – количество событий.

Сетевые графики, у которых $1,0 \leq K_{сл} \leq 1,5$ считаются простыми.

Сетевые графики, у которых $1,51 \leq K_{сл} \leq 2$ имеют среднюю сложность.

Сетевые графики, у которых $K_{сл} > 2$ являются сложными.

Рассмотрим анализ коэффициента напряженности работы $K_{н}$:

$$K_{н} = \frac{t(L_{max}) - t'_{кр}}{t_{кр} - t'_{кр}}, \quad (12)$$

где:

$t(L_{max})$ – продолжительность максимального пути, проходящего через работу P_{ij} ;

$t_{кр}$ – продолжительность (длина) критического пути;

$t'_{кр}$ – продолжительность отрезка рассматриваемого максимального пути, совпадающего с критическим путем;

Чем ближе к 1 K_n работы P_{ij} , тем сложнее выполнить данную работу в установленные сроки» [8].

Чем ближе K_n работы P_{ij} к 0, тем большим относительным резервом обладает максимальный путь, проходящий через данную работу.

«Вычисленные коэффициенты напряженности позволяют дополнительно классифицировать работы по зонам.

В зависимости от величины K_n выделяют три зоны:

- критическую ($K_n > 0,8$);
- подкритическую ($0,6 < K_n < 0,8$);
- резервную ($K_n < 0,6$).

Шаг 5. Оптимизировать сетевой график.

Оптимизация сетевого графика в зависимости от полноты решаемых задач может быть условно разделена на частную и комплексную.

Видами частной оптимизации сетевого графика являются: минимизация времени выполнения комплекса работ при заданной его стоимости; минимизация стоимости комплекса работ при заданном времени выполнения проекта.

Комплексная оптимизация представляет собой нахождение оптимального соотношения величин стоимости и сроков выполнения проекта в зависимости от конкретных целей, ставящихся при его реализации.

Для оптимизации сетевого графика в условиях неопределенности используем метод «время-стоимость».

При использовании этого метода предполагается, что при уменьшении продолжительности работы возрастает ее стоимость» [5].

Каждая работа $P_{i,j}$ характеризуется продолжительностью $t_{i,j}$, удовлетворяющей условию:

$$a_{ij} \leq t_{i,j} \leq b_{ij}, \quad (13)$$

где:

a_{ij} – минимально возможная (экстренная) продолжительность работы $P_{i,j}$, которую только можно осуществить в условиях разработки;

b_{ij} – нормальная продолжительность выполнения работы P_{ij} .

Затраты на ускорение работы P_{ij} (по сравнению с нормальной продолжительностью) на единицу времени рассчитываются по формуле:

$$h_{ij} = \frac{C_{\max ij} - C_{\min ij}}{b_{ij} - a_{ij}}, \quad (14)$$

где:

h_{ij} – коэффициент затрат на ускорение работы P_{ij} ;

$C_{\min ij}$ и $C_{\max ij}$ - значения стоимости при нормальной и экстренной продолжительности работы, соответственно.

При частной оптимизации сетевого графика продолжительность каждой работы, имеющей резерв времени, увеличивают до тех пор, пока не будет исчерпан этот резерв или пока не будет достигнуто верхнее значение продолжительности b_{ij} .

Стоимость первоначального варианта сетевого графика равна сумме стоимостей всех работ (включая работы, не имеющие резервов и вычисляется по формуле:

$$C = \sum_{P_{ij}} C_{ij} \quad (15)$$

Соответственно, стоимость выполнения проекта после оптимизации уменьшится на величину:

$$\Delta C = \sum_{P_{ij}} \Delta C_{ij} = \sum_{P_{ij}} (b_{ij} - t_{ij}) h_{ij} \quad (16)$$

Как следует из вышеизложенного, построение, анализ и оптимизация графика связаны со сложными расчетами и представляют собой трудоемкий процесс.

Поэтому для анализа и оптимизации сетей необходимо использовать

специальное программное обеспечение.

Выводы к главе 2

Вторая глава ВКР посвящена анализу и оптимизация сетевого графика в условиях неопределенности.

Результаты проделанной работы позволили сделать следующие выводы:

1. Анализ сетевого графика начинается с анализа топологии сети, включающего контроль построения сетевого графика, установление целесообразности выбора работ, степени их расчленения.
2. Для оптимизации сетевых графиков используются метод критического пути и анализ расписания по методу PERT.
3. Как показал анализ, метод PERT более подходит для решения задач анализа и оптимизация сетевого графика в условиях неопределенности.
4. Построение, анализ и оптимизация графика связаны со сложными расчетами и представляют собой трудоемкий процесс. Поэтому для анализа и оптимизации сетей необходимо использовать специальное программное обеспечение.

Глава 3 Разработка программы анализа и оптимизации сетевых графиков

3.1 Обзор и анализ программного обеспечения для сетевого планирования

Рассмотрим и сравним программы, обеспечивающие автоматизацию построения, анализа и оптимизации сетевых графиков.

3.1.1 Программный продукт «1С: Управление проектным офисом»

Программный продукт (ПП) «1С: Управление проектным офисом» относится к категории информационных систем управления проектами.

«Ключевым механизмом контура управления проектами является механизм актуализации проекта: на основании данных о фактическом выполнении (или же срыве выполнения) проектных работ нижнего уровня декомпозиции система производит полный перерасчёт календарных сроков всех элементов проекта. Актуализация проекта, выполняемая регулярно, позволяет руководителю получать достоверную информацию о состоянии работ и оперативно реагировать на возникающие проблемы.

На уровне проекта система позволяет производить:

- план-фактный анализ проектных задач по срокам, длительности, стоимости, затратам ресурсов;
- анализ контрольных событий проекта, расчёт прогнозируемых и фактических финансовых санкций» [6].

Визуальный инструментарий системы позволяет руководителю получить информацию о проекте в разных представлениях: табличные представления, дерево данных, диаграммы Ганта, карта проектных вех и сетевой график (рисунок 9).

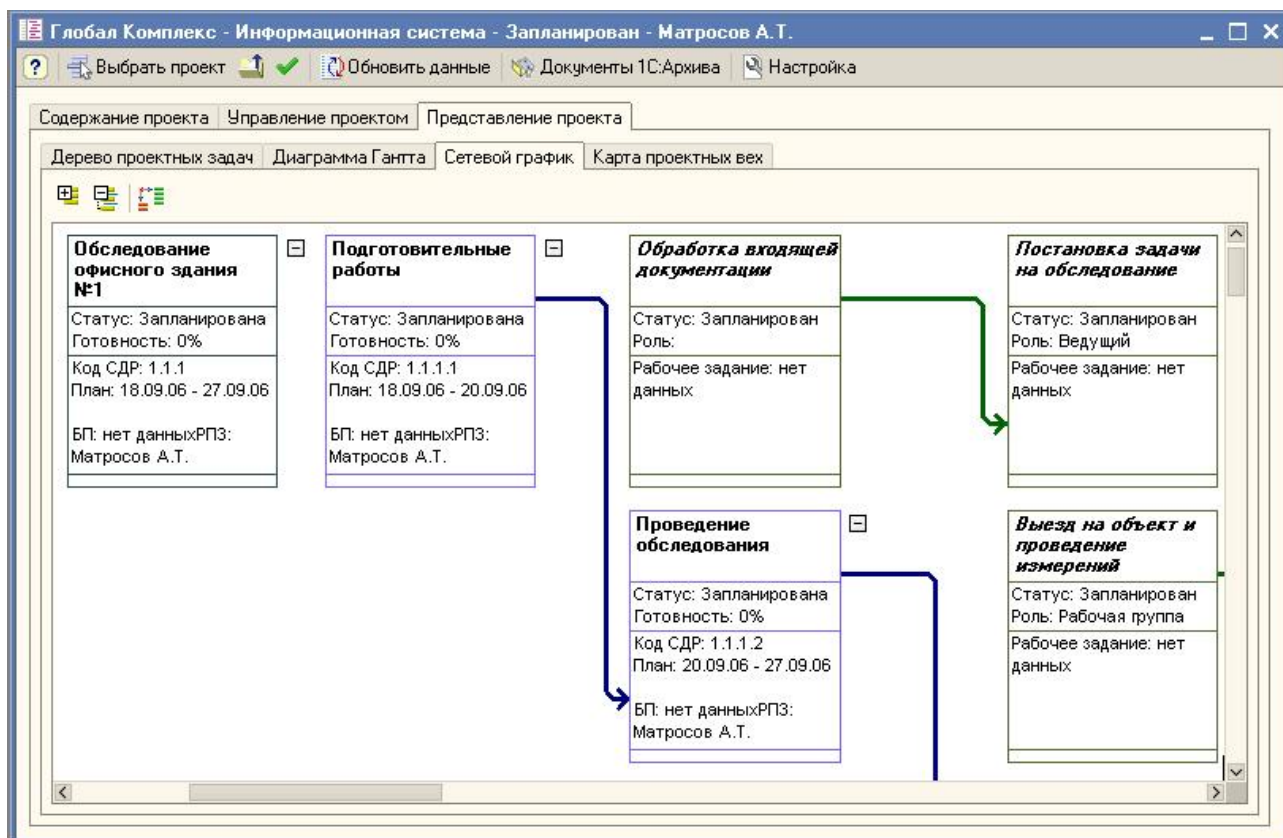


Рисунок 9 – Экран представления сетевого графика проекта

ПП «1С:Управление проектным офисом» может быть интегрирован в рамках единой информационной базы с любым типовым решением фирмы на основе «1С: Предприятие 8.x».

3.1.2 Онлайн-сервис PERT Chart Generator

Онлайн-сервис Visme PERT Chart Generator для упрощенного управления проектами предназначен для построения диаграмм по методу PERT [16].

Онлайн-сервис позволяет создать собственную диаграмму PERT с помощью онлайн-инструмента диаграммы PERT прямо на панели инструментов дизайна Visme (рисунок 10).

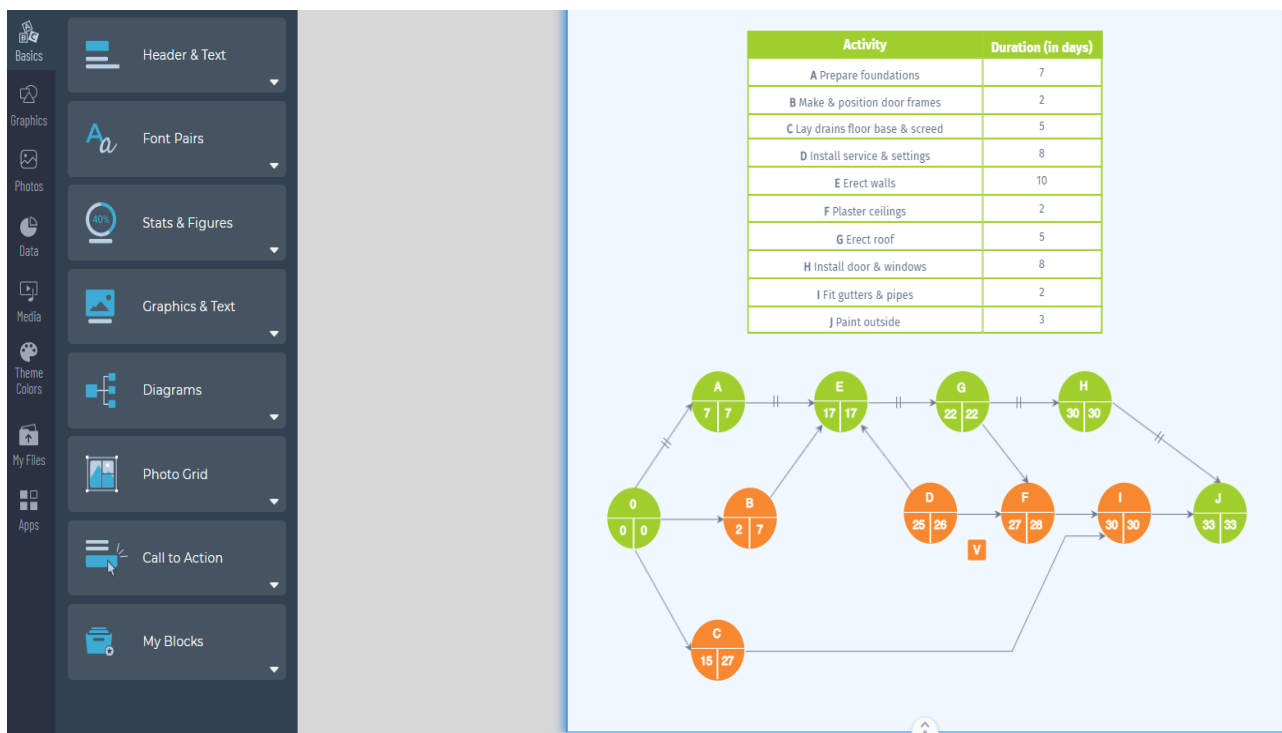


Рисунок 10 – Экран среды для построения диаграмм PERT

PERT Chart Generator сочетает в себе возможности и простоту использования, чтобы создать платформу, позволяющую каждому выполнять свою работу наилучшим образом.

Функциональные возможности онлайн-сервиса:

- простой в использовании генератор диаграмм PERT;
- работает по принципу «Соединяйте различные формы и линии».
- позволяет настраивать текст, цвета, размеры и позиции на графике.

По мнению разработчиков сервиса, с помощью программного обеспечения Visme PERT для создания диаграмм легко упростить самые сложные проекты.

Одним из достоинств онлайн-сервиса является наличие бесплатного тарифа, для доступа к которому достаточно указать зарегистрироваться под учетной записью в популярной социальной сети.

К недостаткам можно отнести ограниченные возможности для анализа и оптимизации, а также нелокализованный интерфейс, что создает некоторые проблемы при разработке диаграммы.

3.1.3 Онлайн-калькулятор по сетевому планированию

Бесплатный онлайн-калькулятор math.semestr.ru предлагает пользователям сервисы для решения задач по математике, информатике, ЭММ, теории вероятностей, статистике, эконометрике и другим дисциплинам, в том числе по сетевому планированию [3].

Все выкладки оформляются в формате Word и Excel.

На рисунке 11 представлено окно главной страницы онлайн-калькулятора

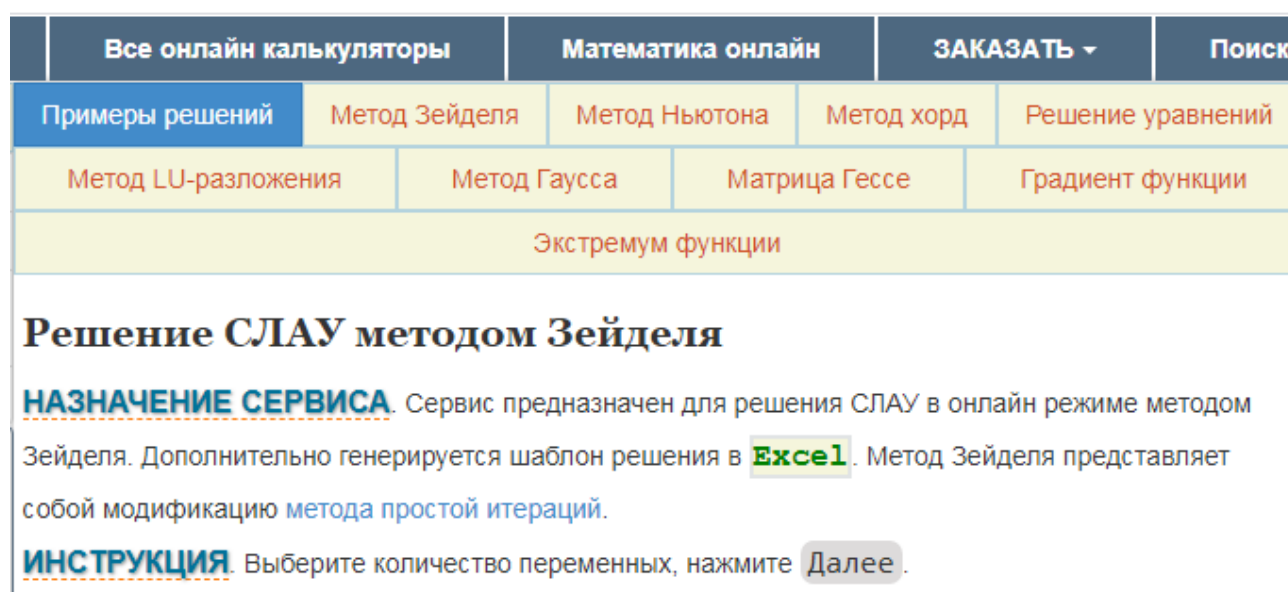


Рисунок 11-Экран главного окна онлайн-калькулятора math.semestr.ru

По умолчанию все вычисления оформляются в файле формата MS Word.

Однако следует учесть, что такая возможность имеется только для браузеров, поддерживающих Javascript.

Если поддержка Javascript отключена, необходимо включить ее в настройках браузера.

В остальных случаях всегда доступен просмотр решения непосредственно в браузере.

К достоинствам данного онлайн-сервиса следует отнести то, что помимо калькулятора пользователю по каждой дисциплине для ознакомления

предлагается соответствующий теоретический материал и примеры решения задач.

Кроме того, онлайн-калькулятор имеет простой в использовании и многоязычный интерфейс.

Недостатками решения являются ограниченные возможности для построения графиков и некоторая избыточность информации в рабочих окнах, что усложняет поиск конкретной дисциплины или определенного метода решения задачи.

Для выбора программы используем таблицу 2, составленную на основе анализа источников по данной тематике.

Критерии оценивания:

- 0 – полное несоответствие требованиям;
- 1 – значительное несоответствие требованиям;
- 2 – незначительное несоответствие требованиям;
- 3 – полное соответствие требованиям.

Таблица 2 - Сравнительный анализ программ автоматизации сетевого планирования

Характеристика/балл	1С: Управление проектным офисом	PERT Chart Generator	math.semestr.ru
табличное представление сетевой модели	3	0	3
графическое представление сетевой модели	1	3	2
анализ сетевой модели	2	1	3
оптимизация сетевой модели	2	1	3
удобство использования	2	2	1
простота интеграции	1	0	0
низкая стоимость владения	0	1	3
Итого	11	8	15

Анализ показал, что наилучшими характеристиками обладает онлайн-калькулятор сетевого планирования. Однако сложность интеграции и избыточность интерфейса создают некоторые проблемы с использованием данного онлайн-сервиса.

Поэтому используем данный онлайн-калькулятор в качестве основы для реализации собственной программы анализа и оптимизации сетевых графиков.

3.2 Реализация программы анализа и оптимизации сетевых графиков

В качестве средства средств разработки программы используем бесплатный онлайн-сервис math.semestr.ru и технологию Electron [10, 12].

Electron - это платформа для создания собственных приложений с использованием HTML, CSS и JavaScript.

Electron - это платформа с открытым исходным кодом, которая упрощает задачу создания сложных настольных приложений, удаляя сложные части кодирования, так что разработчики могут сосредоточиться на фактическом создании приложения, а не на других трудоемких аспектах кодирования и программирования.

Он использует Node.js, а также браузер Chromium, т.е. знакомые веб-разработчикам технологии для создания обычных приложений.

Electron позволяет разрабатывать настольные приложения с графическим интерфейсом пользователя, используя внешние и внутренние компоненты, изначально разработанные для веб-приложений.

Приложения Electron состоят из нескольких процессов. Есть «главный» процесс и несколько процессов «рендеринга». Основной процесс запускает логику приложения, а затем может запускать несколько процессов визуализации, отображая окна, которые появляются на экране пользователя, визуализируя HTML и CSS.

С помощью технологии Electron создавать кроссплатформенные приложения, которые работают на Mac или Windows.

Преимуществами технологии Electron являются:

- высокая безопасность данных;
- доступность;
- высокая производительность;

- простое управление;
- многофункциональный фреймворк;
- совместимость;
- взаимодействие с веб-инструментами UI / UX;
- сокращение времени и затрат на разработку программы.

К недостаткам технологии относят значительные накладные расходы по сравнению с собственными приложениями с аналогичной функциональностью.

Приложения, созданные с помощью Electron могут занимать больше места для хранения и оперативной памяти и могут работать медленнее, чем аналогичные приложения, созданные с использованием технологий, присущих операционной системе.

Компоненты технологии Electron:

- JavaScript - это динамический язык программирования, который используется для веб-разработки, в веб-приложениях, для разработки игр и многого другого. Он позволяет разработчикам реализовывать динамические функции на веб-страницах, которые нельзя реализовать только с помощью HTML и CSS [9];
- Chromium - это проект браузера с открытым исходным кодом, целью которого является создание более безопасного, быстрого и стабильного способа работы в Интернете для всех пользователей. Вся проектная документация, обзоры архитектуры, информация о тестировании и другие материалы, позволяющие научиться создавать и работать с исходным кодом Chromium представлены на сайте проекта [17];
- Node.js - это кроссплатформенная среда выполнения с открытым исходным кодом для разработки серверных и сетевых приложений. Приложения Node.js написаны на JavaScript и могут запускаться в среде выполнения Node.js в OS X, Microsoft Windows и Linux. Node.js также предоставляет богатую библиотеку различных модулей JavaScript, которая в значительной степени упрощает разработку веб-приложений с использованием Node.js [4].

Процесс разработки в технологии Electron состоит из следующих этапов:

- 1) используем URL онлайн-калькулятора;
- 2) добавляем коды на JavaScript и CSS для скрытия лишних элементов интерфейса и обеспечения работы программы как десктоп-приложения. Для работы программы необходим доступ в Интернет.

На рисунке 12 представлен жизненный цикл приложения Electron.

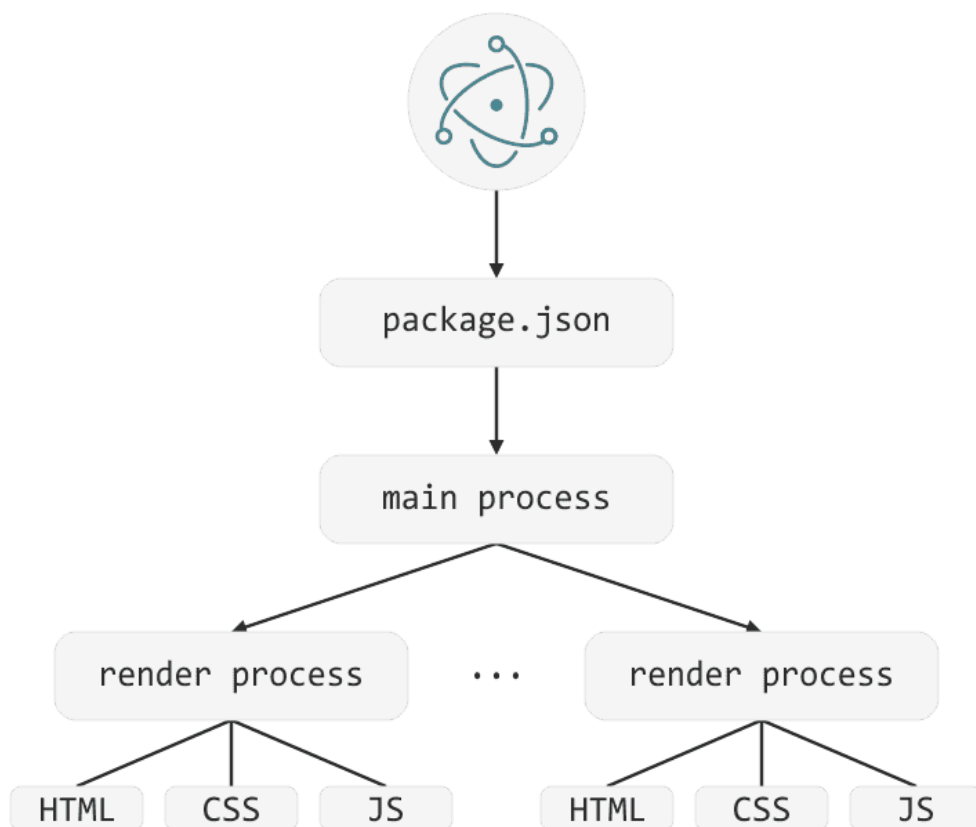


Рисунок 12 – Жизненный цикл приложения Electron

Разработана логическая модель программы.

На рисунке 13 изображена диаграмма вариантов использования программы.

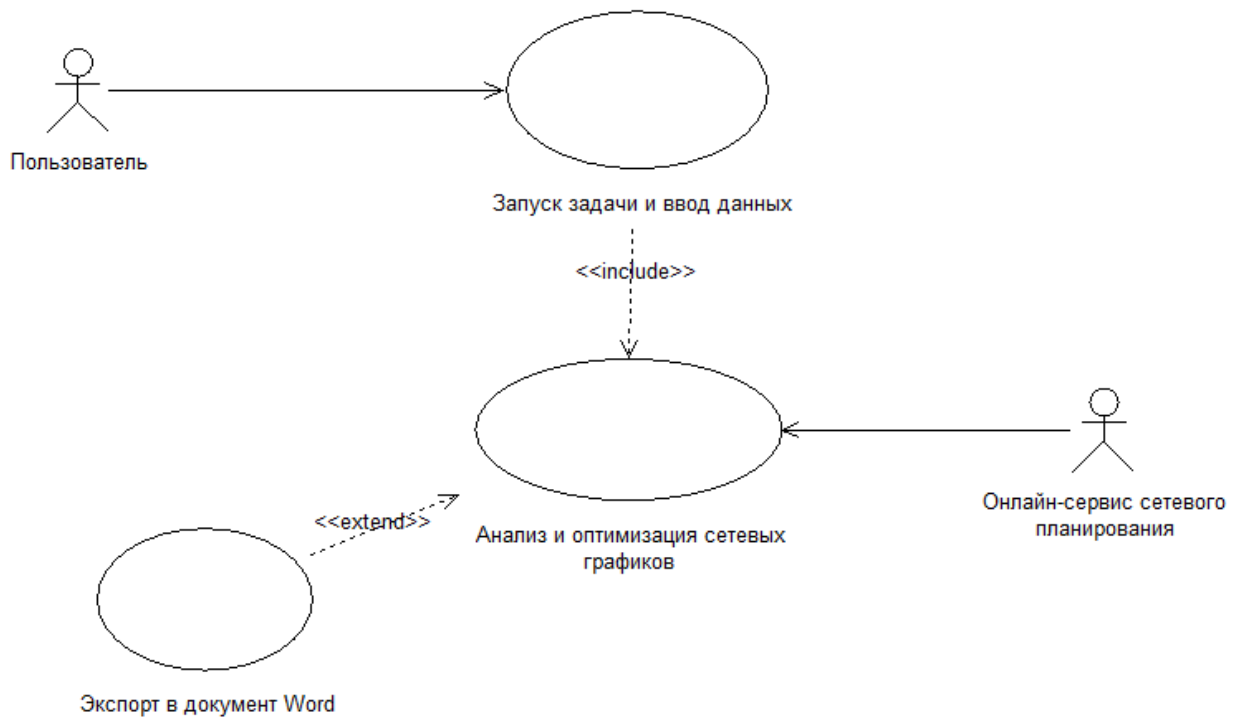


Рисунок 13 – Диаграмма вариантов использования программы

На рисунке 14 изображена диаграмма компонентов программы

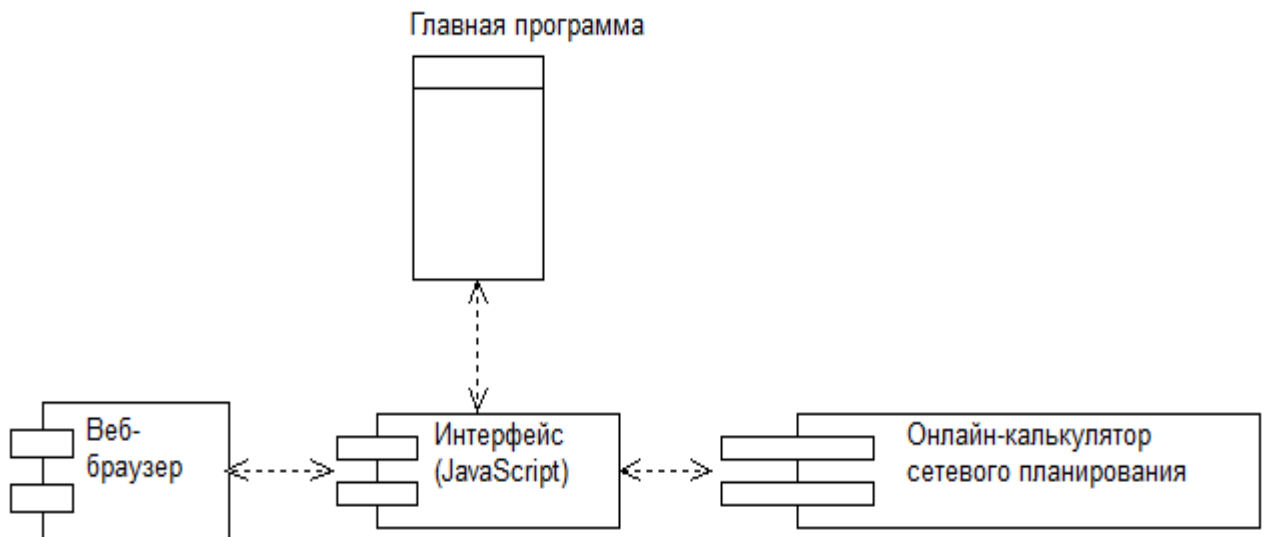


Рисунок 14 – Диаграмма компонентов программы

Фрагмент кода программы на языке JavaScript приведен в Приложении А.

3.3 Автоматизация решения задачи анализа и оптимизации сетевых графиков

Для проверки работоспособности программы рассмотрим пример решения задачи анализа, планирования в условиях неопределенности и оптимизации сетевого графика.

Параметры сетевой модели заданы в таблице 3.

Таблица 3 – Параметры сетевой модели

Код работы (i,j)	Продолжительность		Нормальная стоимость (д.е.)	Срочная стоимость (д.е.)
	$t_{\min}(i,j)$	$t_{\max}(i,j)$		
1,2	5	10	50	80
1,4	2	7	20	30
1,5	1	6	70	100
2,3	2	4,5	90	100
2,8	9	19	100	130
3,4	1	3,5	30	35
3,6	9	19	50	65
4,7	4	6,5	70	90
5,7	2	7	100	120
6,8	7	12	40	50
7,8	5	7,5	75	90

Ввод параметров сетевой модели выполняется с помощью формы, представленной на рисунке 15.

Рисунок 15 – Окно ввода параметров сетевой модели

«В нашем задании продолжительность выполнения работы задаётся двумя оценками – минимальной и максимальной.

Минимальная оценка $t_{\min}(i,j)$ характеризует продолжительность выполнения работы при наиболее благоприятных обстоятельствах, а максимальная $t_{\max}(i,j)$ – при наиболее неблагоприятных условиях.

Продолжительность работы в этом случае рассматривается, как случайная величина, которая в результате реализации может принять любое значение в заданном интервале. Такие оценки называются вероятностными (случайными), и их ожидаемое значение $t_{ож}(i,j)$ оценивается по формуле:

$$t_{ож}(i,j) = (3 t_{\min}(i,j) + 2 t_{\max}(i,j)) / 5 \quad (17)$$

Для характеристики степени разброса возможных значений вокруг ожидаемого уровня используется показатель дисперсии:

$$S^2(i,j) = 0,04(t_{\max}(i,j) - t_{\min}(i,j))^2 \quad (18)$$

Рассчитаем ожидаемое значение и показатель дисперсии» [3].

$$t_{ож}(1,2) = (3 * 5 + 2 * 5) / 5 = 7$$

$$t_{ож}(1,4) = (3 * 2 + 2 * 2) / 5 = 4$$

$$t_{ож}(1,5) = (3 * 1 + 2 * 1) / 5 = 3$$

$$t_{ож}(2,3)=(3*2+2*2)/5=3$$

$$t_{ож}(2,8)=(3*9+2*9)/5=13$$

$$t_{ож}(3,4)=(3*1+2*1)/5=2$$

$$t_{ож}(3,6)=(3*9+2*9)/5=13$$

$$t_{ож}(4,7)=(3*4+2*4)/5=5$$

$$t_{ож}(5,7)=(3*2+2*2)/5=4$$

$$t_{ож}(6,8)=(3*7+2*7)/5=9$$

$$t_{ож}(7,8)=(3*5+2*5)/5=6$$

$$S^2(1,2)=0.04(10-5)^2=1$$

$$S^2(1,4)=0.04(7-2)^2=1$$

$$S^2(1,5)=0.04(6-1)^2=1$$

$$S^2(2,3)=0.04(4.5-2)^2=0.25$$

$$S^2(2,8)=0.04(19-9)^2=4$$

$$S^2(3,4)=0.04(3.5-1)^2=0.25$$

$$S^2(3,6)=0.04(19-9)^2=4$$

$$S^2(4,7)=0.04(6.5-4)^2=0.25$$

$$S^2(5,7)=0.04(7-2)^2=1$$

$$S^2(6,8)=0.04(12-7)^2=1$$

$$S^2(7,8)=0.04(7.5-5)^2=0.25$$

Полученные данные занесем в таблицу 4.

Таблица 4 – Ожидаемые значения и показатели дисперсии

Работа (i,j)	$t_{min}(i,j)$	$t_{max}(i,j)$	$m(i,j)$	Ожидаемая продолжительность $t_{ож}(i,j)$	Дисперсия $S^2(i,j)$
1,2	5	10	50	7	1
1,4	2	7	20	4	1
1,5	1	6	70	3	1
2,3	2	4.5	90	3	0.25
2,8	9	19	100	13	4
3,4	1	3.5	30	2	0.25
3,6	9	19	50	13	4
4,7	4	6.5	70	5	0.25
5,7	2	7	100	4	1
6,8	7	12	40	9	1
7,8	5	7.5	75	6	0.25

«Используя полученные данные, мы можем найти основные характеристики сетевой модели табличным методом, критический путь и его продолжительность.

Важнейшим показателем сетевого графика являются резервы времени.

Резервы времени каждого пути показывают, на сколько может быть увеличена продолжительность данного пути без ущерба для наступления завершающего события. Поскольку каждый не критический путь сетевого графика имеет свой полный резерв времени, то и каждое событие этого пути имеет свой резерв времени (таблица 5)» [3].

Таблица 5 – Обозначение элементов сетевой модели

Элемент сети	Наименование параметра	Условное обозначение параметра
Событие <i>i</i>	Ранний срок свершения события	$tr(i)$
	Поздний срок свершения события	$t(i)$
	Резерв времени события	$R(i)$
Работа (<i>i, j</i>)	Продолжительность работы	$t(i,j)$
	Ранний срок начала работы	$t^{PH}(i,j)$
	Ранний срок окончания работы	$t^{PO}(i,j)$
	Поздний срок начала работы	$t^{PN}(i,j)$
	Поздний срок окончания работы	$t^{PO}(i,j)$
	Полный резерв времени работы	$R^H(i,j)$
Путь <i>L</i>	Продолжительность пути	$t(L)$
	Продолжительность критического пути	$t_{кр}$
	Резерв времени пути	$R(L)$

«Резерв времени события показывает, на какой допустимый период времени можно задержать наступление этого события, не вызывая при этом увеличения срока выполнения комплекса работ.

Для определения резервов времени по событиям сети рассчитывают наиболее ранние tr и наиболее поздние tp сроки свершения событий. Любое событие не может наступить прежде, чем свершиться все предшествующие ему события и не будут выполнены все предшествующие работы» [3]. Поэтому ранний (или ожидаемый) срок $tr(i)$ свершения *i*-ого события определяется продолжительностью максимального пути, предшествующего этому событию:

$$t^P(i) = \max(t(L_{ni})) \quad (19)$$

где L_{ni} – любой путь, предшествующий *i*-ому событию, то есть путь от исходного до *i*-ого события сети.

«Если событие j имеет несколько предшествующих путей, а, следовательно, несколько предшествующих событий i , то ранний срок свершения события j удобно находить по формуле:

$$t^p(j) = \max[t^p(i) + t(i,j)] \quad (20)$$

Задержка свершения события i по отношению к своему раннему сроку не отразится на сроке свершения завершающего события (а значит, и на сроке выполнения комплекса работ) до тех пор, пока сумма срока свершения этого события и продолжительности (длины) максимального из следующих за ним путей не превысит длины критического пути. Поэтому поздний (или предельный) срок $t^p(i)$ свершения i -ого события равен:

$$T^p(i) = t_{kp} - \max(t(L_{ci})) \quad (21)$$

где L_{ci} - любой путь, следующий за i -ым событием, т.е. путь от i -ого до завершающего события сети.

Если событие i имеет несколько последующих путей, а, следовательно, несколько последующих событий j , то поздний срок свершения события i удобно находить по формуле:

$$t^p(i) = \min[t^p(j) - t(i,j)] \quad (22)$$

Резерв времени $R(i)$ i -ого события определяется как разность между поздним и ранним сроками его свершения:

$$R(i) = t^p(i) - t^p(i) \quad (23)$$

Резерв времени события показывает, на какой допустимый период времени можно задержать наступление этого события, не вызывая при этом увеличения срока выполнения комплекса работ» [3].

Критические события резервов времени не имеют, так как любая задержка в свершении события, лежащего на критическом пути, вызовет такую же задержку в свершении завершающего события. Таким образом, определив ранний срок наступления завершающего события сети, мы тем самым определяем длину критического пути.

При определении ранних сроков свершения событий $t^p(i)$ двигаемся по сетевому графику слева направо и используем формулы (1), (2).

Расчет сроков свершения событий.

Для $i=1$ (начального события), очевидно $tp(1)=0$.

$$i=2: t^p(2) = t^p(1) + t(1,2) = 0 + 7 = 7.$$

$$i=3: t^p(3) = t^p(2) + t(2,3) = 7 + 3 = 10.$$

$$i=4: \max(t^p(1) + t(1,4); t^p(3) + t(3,4)) = \max(0 + 4; 10 + 2) = 12.$$

$$i=5: t^p(5) = t^p(1) + t(1,5) = 0 + 3 = 3.$$

$$i=6: t^p(6) = t^p(3) + t(3,6) = 10 + 13 = 23.$$

$$i=7: \max(t^p(4) + t(4,7); t^p(5) + t(5,7)) = \max(12 + 5; 3 + 4) = 17.$$

$$i=8: \max(t^p(2) + t(2,8); t^p(6) + t(6,8); t^p(7) + t(7,8)) = \max(7 + 13; 23 + 9; 17 + 6) = 32.$$

Длина критического пути равна раннему сроку свершения завершающего события 8: $t_{кр}=tp(8)=32$

«При определении поздних сроков свершения событий $t_n(i)$ движемся по сети в обратном направлении, то есть справа налево и используем формулы (3), (4).

Для $i=8$ (завершающего события) поздний срок свершения события должен равняться его раннему сроку (иначе изменится длина критического пути): $t^n(8)=t^p(8)=32$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 7. Просматриваются все строчки, начинающиеся с номера 7» [3].

$$i=7: t^n(7) = t^n(8) - t(7,8) = 32 - 6 = 26.$$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 6. Просматриваются все строчки, начинающиеся с номера 6.

$$i=6: t^n(6) = t^n(8) - t(6,8) = 32 - 9 = 23.$$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 4. Просматриваются все строчки, начинающиеся с номера 4.

$$i=4: t^n(4) = t^n(7) - t(4,7) = 26 - 5 = 21.$$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 3. Просматриваются все строчки, начинающиеся с номера 3.

$$i=3: \min(t^p(4) - t(3,4); t^p(6) - t(3,6)) = \min(21 - 2; 23 - 13) = 10.$$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 5. Просматриваются все строчки, начинающиеся с номера 5.

$$i=5: t^p(5) = t^p(7) - t(5,7) = 26 - 4 = 22.$$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 2. Просматриваются все строчки, начинающиеся с номера 2.

$$i=2: \min(t^p(3) - t(2,3); t^p(8) - t(2,8)) = \min(10 - 3; 32 - 13) = 7.$$

Далее просматриваются строки, оканчивающиеся на номер предпоследнего события, т.е. 1. Просматриваются все строчки, начинающиеся с номера 1.

$$i=1: \min(t^p(2) - t(1,2); t^p(4) - t(1,4); t^p(5) - t(1,5)) = \min(7 - 7; 21 - 4; 22 - 3) = 0.$$

Таблица 6 - Расчет резерва событий

Номер события	Сроки свершения события: ранний $t^p(i)$	Сроки свершения события: поздний $t^p(i)$	Резерв времени, $R(i)$
1		0	0
2	7	7	0
3	10	10	0
4	12	21	9
5	3	22	19
6	23	23	0
7	17	26	9
8	32	32	0

Сформируем таблицу анализа сетевой модели по времени.

«Перечень работ и их продолжительность перенесем во вторую и третью графы. При этом работы следует записывать в графу 2 последовательно: сначала начиная с номера 1, затем с номера 2 и т.д.

Во второй графе поставим число, характеризующее количество непосредственно предшествующих работ (КПР) тому событию, с которого начинается рассматриваемая работа» [3].

Так, для работы (4,7) в графу 1 поставим число 2, т.к. на номер 4 оканчиваются 2 работы: (1,4),(3,4).

Графу 4 получаем из таблицы 1 ($t^p(i)$). Графу 7 получаем из таблицы 1 ($t^p(i)$).

Значения в графе 5 получаются в результате суммирования граф 3 и 4.

«В графе 6 позднее начало работы определяется как разность позднего окончания этих работ и их продолжительности (из значений графы 7 вычитаются данные графы 3). Содержимое графы 8 ($R(ij)$) равно разности граф 6 и 4 или граф 7 и 5. Если $R(ij)$ равен нулю, то работа является критической. Полный резерв пути показывает, на сколько в сумме может быть увеличена продолжительность всех работ, принадлежащих данному пути, при условии, что срок выполнения всего комплекса работ не изменится. Образовывается, когда предшествующие работы закончатся в свой наиболее ранний срок» [3].

«Находим полный резерв $R^{\Pi}_{i-j} = T_{\Pi j} - t_{i-j} - T_{p_i}$

$$R^{\Pi}_{(1,2)} = 7 - 7 - 0 = 0$$

$$R^{\Pi}_{(1,4)} = 21 - 4 - 0 = 17$$

$$R^{\Pi}_{(1,5)} = 22 - 3 - 0 = 19$$

$$R^{\Pi}_{(2,3)} = 10 - 3 - 7 = 0$$

$$R^{\Pi}_{(2,8)} = 32 - 13 - 7 = 12$$

$$R^{\Pi}_{(3,4)} = 21 - 2 - 10 = 9$$

$$R^{\Pi}_{(3,6)} = 23 - 13 - 10 = 0$$

$$R^{\Pi}_{(4,7)} = 26 - 5 - 12 = 9$$

$$R^{\Pi}_{(5,7)} = 26 - 4 - 3 = 19$$

$$R^{\Pi}_{(6,8)} = 32 - 9 - 23 = 0$$

$$R^{\Pi}_{(7,8)} = 32 - 6 - 17 = 9» [3]$$

Свободный резерв времени также можно найти и по формуле:

$$R^C_{i-j} = T_{\Pi i} - t_{i-j} - T_{p_i} \quad (24)$$

$$R^C_{(1,2)} = 7-7-0 = 0$$

$$R^C_{(1,4)} = 12-4-0 = 8$$

$$R^C_{(1,5)} = 3-3-0 = 0$$

$$R^C_{(2,3)} = 10-3-7 = 0$$

$$R^C_{(2,8)} = 32-13-7 = 12$$

$$R^C_{(3,4)} = 12-2-10 = 0$$

$$R^C_{(3,6)} = 23-13-10 = 0$$

$$R^C_{(4,7)} = 17-5-12 = 0$$

$$R^C_{(5,7)} = 17-4-3 = 10$$

$$R^C_{(6,8)} = 32-9-23 = 0$$

$$R^C_{(7,8)} = 32-6-17 = 9$$

Независимый резерв времени также можно найти и по формуле:

$$R^H_{i-j} = T_{pj} - t_{i-j} - T_{\Pi i} \quad (25)$$

$$R^H_{(1,2)} = 7-7-0 = 0$$

$$R^H_{(1,4)} = 12-4-0 = 8$$

$$R^H_{(1,5)} = 3-3-0 = 0$$

$$R^H_{(2,3)} = 10-3-7 = 0$$

$$R^H_{(2,8)} = 32-13-7 = 12$$

$$R^H_{(3,4)} = 12-2-10 = 0$$

$$R^H_{(3,6)} = 23-13-10 = 0$$

$$R^H_{(4,7)} = 17-5-21 = -9$$

$$R^H_{(5,7)} = 17-4-22 = -9$$

$$R^H_{(6,8)} = 32-9-23 = 0$$

$$R^H_{(7,8)} = 32-6-26 = 0$$

Сведем данные анализа в таблицу 7

Таблица 7 - Анализ сетевой модели по времени

Работа (i,j)	Количество предшествующих работ	Продолжительность t_{ij}	Ранние сроки : начало $t_{ij}^{P.H.}$	Ранние сроки: окончание $t_{ij}^{P.O.}$	Поздние сроки: начало $t_{ij}^{П.H.}$	Поздние сроки: окончание $t_{ij}^{П.O.}$	Резервы времени: полный $R_{ij}^{П}$	Независимый резерв времени R_{ij}^{H}	Частный резерв I рода, R_{ij}^I	Частный резерв II рода, R_{ij}^C
(1,2)	0	7	0	7	0	7	0	0	0	0
(1,4)	0	4	0	4	17	21	17	8	17	8
(1,5)	0	3	0	3	19	22	19	0	19	0
(2,3)	1	3	7	10	7	10	0	0	0	0
(2,8)	1	13	7	20	19	32	12	12	12	12
(3,4)	1	2	10	12	19	21	9	0	9	0
(3,6)	1	13	10	23	10	23	0	0	0	0
(4,7)	2	5	12	17	21	26	9	-9	0	0
(5,7)	1	4	3	7	22	26	19	-9	0	10
(6,8)	1	9	23	32	23	32	0	0	0	0
(7,8)	2	6	17	23	26	32	9	0	0	9

Следует отметить, что кроме полного резерва времени работы, выделяют еще три разновидности резервов.

«Частный резерв времени первого вида R_1 - часть полного резерва времени, на которую можно увеличить продолжительность работы, не изменив при этом позднего срока ее начального события. R_1 находится по формуле:

$$R_1(i,j) = R^п(i,j) - R(i) \quad (26)$$

Частный резерв времени второго вида, или свободный резерв времени R_c работы (i,j) представляет собой часть полного резерва времени, на которую можно увеличить продолжительность работы, не изменив при этом раннего срока ее конечного события. R_c находится по формуле:

$$R_c(i,j) = R^п(i,j) - R(j) \quad (27)$$

Значение свободного резерва времени работы указывает на расположение резервов, необходимых для оптимизации.

Независимый резерв времени R_n работы (i,j) - часть полного резерва, получаемая для случая, когда все предшествующие работы заканчиваются в поздние сроки, а все последующие начинаются в ранние сроки» [3].

R_n находится по формуле:

$$R_n(i,j) = R^п(i,j) - R(i) - R(j) \quad (28)$$

Критический путь: (1,2)(2,3)(3,6)(6,8)

Продолжительность критического пути: 32

Анализ сетевого графика

$$K_c = 11 / 8 = 1.38$$

Поскольку $K_c < 1.5$, то сетевой график является простым.

Расчет коэффициента напряженности приведен в таблице 8.

Таблица 8 - Расчет коэффициента напряженности

Работа	Путь	Максимальный путь, $t(L_{max})$	Совпадающие работы	t_{1kp}	Расчет	K_n
(1,2)	(1,2)(2,3)(3,6)(6,8)	32	(1,2)(2,3)(3,6)(6,8)	32	-	-
(1,4)	(1,4)(4,7)(7,8)	15	(1,1)	0	$(15-0)/(32-0)$	0.469
(1,5)	(1,5)(5,7)(7,8)	13	(1,1)	0	$(13-0)/(32-0)$	0.406
(2,3)	(1,2)(2,3)(3,6)(6,8)	32	(1,2)(2,3)(3,6)(6,8)	32	-	-
(2,8)	(1,2)(2,8)	20	(1,2)	7	$(20-7)/(32-7)$	0.52
(3,4)	(1,2)(2,3)(3,4)(4,7)(7,8)	23	(1,2)(2,3)	10	$(23-10)/(32-10)$	0.591
(3,6)	(1,2)(2,3)(3,6)(6,8)	32	(1,2)(2,3)(3,6)(6,8)	32	-	-
(4,7)	(1,2)(2,3)(3,4)(4,7)(7,8)	23	(1,2)(2,3)	10	$(23-10)/(32-10)$	0.591
(5,7)	(1,5)(5,7)(7,8)	13	(1,1)	0	$(13-0)/(32-0)$	0.406
(6,8)	(1,2)(2,3)(3,6)(6,8)	32	(1,2)(2,3)(3,6)(6,8)	32	-	-
(7,8)	(1,2)(2,3)(3,4)(4,7)(7,8)	23	(1,2)(2,3)	10	$(23-10)/(32-10)$	0.591

Вычисленные коэффициенты напряженности позволяют дополнительно классифицировать работы по зонам. В зависимости от величины K_n выделяют три зоны: критическую ($K_n > 0,8$); подкритическую ($0,6 < K_n < 0,8$); резервную ($K_n < 0,6$).

При этом стоимость строительства павильона в нормальном режиме выполнения работ составит:

$$50 + 20 + 70 + 90 + 100 + 30 + 50 + 70 + 100 + 40 + 75 = 695$$

Сократим критический путь на 1 день за минимальную доплату к ранее найденной стоимости в нормальном режиме выполнения работ.

Прежде всего, установим, за какую минимальную доплату можно завершить комплекс работ за 31 дня вместо 32 дней. Сокращение срока на день возможно только за счет уменьшения на один день продолжительности критического пути.

Это, в свою очередь, можно достичь, ускоряя на день какие-либо критические работы, лежащие на этих путях.

«Для того, чтобы установить какие критические работы в совокупности ускорять дешевле, нужно вычислить средние стоимости 1 дня ускорения всех выявленных критических работ.

В итоге самым дешевым вариантом для сокращения срока на 1 день является ускорение на один день работ, что приведет к новому критическому сроку 31 дня с минимальной доплатой = 0 д.е.

Выполним запланированные ускорения работ и сделаем перерасчет временных характеристик сетевого графика» [3].

Отсюда итоговая стоимость составит $695 + 0 = 695$ д.е.

Программа содержит опцию построения сетевых графиков.

На рисунке 16 изображен сетевой график, построенный в результате решения представленной задачи.

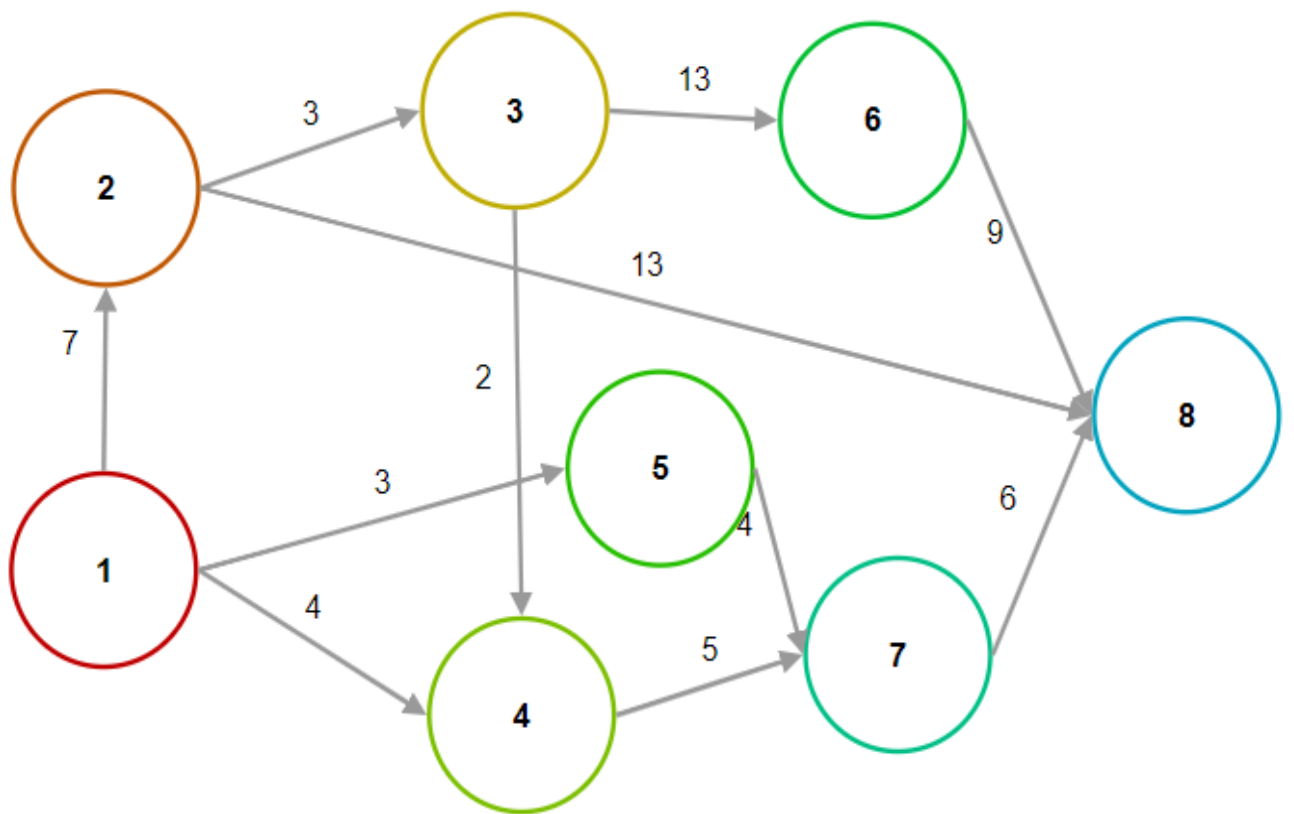


Рисунок 16 – Сетевой график

Таким образом, разработанная программа позволяет решить задачу анализа, планирования в условиях неопределенности и оптимизации сетевого графика.

Выводы к главе 3

Третья глава ВКР посвящена разработке программы анализа, планирования в условиях неопределенности и оптимизации сетевого графика.

Результаты проделанной работы позволили сделать следующие выводы:

Сравнительный анализ известных аналогов программы показал, что наилучшими характеристиками обладает калькулятор сетевого планирования.

Однако сложность интеграции и избыточность интерфейса создают некоторые проблемы с использованием данного онлайн-сервиса.

В качестве средства средств разработки программы выбраны бесплатный онлайн-сервис `math.semestr.ru` и технология Electron.

Проверка работоспособности программы подтвердило возможность решения с ее помощью задач анализа, планирования в условиях неопределенности и оптимизации сетевого графика.

Заключение

Выпускная квалификационная работа посвящена проблеме анализа и оптимизации сетевых графиков в условиях неопределенности.

В процессе работы над ВКР решены следующие задачи:

Произведен обзор и анализ методов сетевого планирования на графиках. Как показал анализ, наиболее востребованными методами сетевого планирования являются PERT, диаграмма Ганта и СПУ. Основными характерными чертами системы СПУ являются обоснованность разрабатываемого плана и оперативность руководства. Сравнительный анализ методов показал, что лучшими характеристиками для решения задач сетевого планирования обладают метод PERT и основанный на нем метод СПУ.

Описана методика решения задачи анализа и оптимизация сетевого графика в условиях неопределенности. Отмечено, что для оптимизации сетевых графиков используются метод критического пути и анализ расписания по методу PERT. Как показал анализ, метод PERT более подходит для решения задач анализа и оптимизация сетевого графика в условиях неопределенности. Построение, анализ и оптимизация графика связаны со сложными расчетами и представляют собой трудоемкий процесс. Поэтому для анализа и оптимизации сетей необходимо использовать специальное программное обеспечение.

Разработана программа для автоматизации анализа и оптимизация сетевого графика в условиях неопределенности. Сравнительный анализ известных аналогов программы показал, что наилучшими характеристиками обладает калькулятор сетевого планирования. Однако сложность интеграции и избыточность интерфейса создают некоторые проблемы с использованием данного онлайн-сервиса. В качестве средства средств разработки программы выбраны бесплатный онлайн-сервис math.semestr.ru и технология Electron.

Проверка работоспособности программы подтвердило возможность решения с ее помощью задач анализа, планирования в условиях неопределенности и оптимизации сетевого графика.

Список используемой литературы и используемых источников

1. Диаграмма Ганта [Электронный ресурс]. URL: https://datavizcatalogue.com/RU/metody/diagramma_ganta.html (дата обращения: 25.05.2021).
2. Моделирование сетевого графика проекта [Электронный ресурс]. URL: <http://projectimo.ru/planirovanie-proekta/setevoj-grafik.html> (дата обращения: 25.05.2021).
3. Онлайн-калькулятор по сетевому планированию [Электронный ресурс]. URL: https://math.semestr.ru/setm/setm_manual.php (дата обращения: 25.05.2021).
4. Платформа Node.js [Электронный ресурс]. URL: <https://nodejs.org/en/> (дата обращения: 25.05.2021).
5. Плескунов М. А. Задачи сетевого планирования : учебное пособие. Екатеринбург : Изд-во Урал. ун-та, 2014. 92 с.
6. Программный продукт «1С: Управление проектным офисом» [Электронный ресурс]. URL: <https://solutions.1c.ru/catalog/project-office/features> (дата обращения: 25.05.2021).
7. Сетевое планирование [Электронный ресурс]. URL: <http://upr-proektom.ru/setevoe-planirovanie> (дата обращения: 25.05.2021).
8. Сетевое планирование в условиях неопределенности [Электронный ресурс]. URL: https://studme.org/80819/ekonomika/setevoe_planirovanie_usloviyah_neopredelennosti (дата обращения: 25.05.2021).
9. Современный учебник JavaScript [Электронный ресурс]. URL: <https://learn.javascript.ru/> (дата обращения: 25.05.2021).
10. Создание десктопного приложения с помощью Electron и веб-технологий [Электронный ресурс]. URL: <https://habr.com/ru/post/272075/> (дата обращения: 25.05.2021).
11. Спиридонов Э.С., Телятникова Н. А. Сетевое планирование в строительстве: Учебное пособие, М.: РУТ (МИИТ), 2018. 75 с.

12. Технология Electron [Электронный ресурс]. URL: <https://electronjs.org> (дата обращения: 25.05.2021).

13. Everything you need to know about PERT charts [Электронный ресурс]. URL: <https://medium.com/nulab/everything-you-need-to-know-about-pert-charts-55978c06b5fc> (дата обращения: 25.05.2021).

14. Liu Jun-yan “Schedule Uncertainty Control: A Literature review” , Physics Procedia 33 (2012) 1842 – 1848.

15. PERT Analysis [Электронный ресурс]. URL: <https://acqnotes.com/acqnote/tasks/pert-analysis> (дата обращения: 25.05.2021).

16. PERT chart generator [Электронный ресурс]. URL: <https://www.visme.co/pert-chart-generator/> (дата обращения: 25.05.2021).

17. The Chromium Projects [Электронный ресурс]. URL: <https://www.chromium.org/> (дата обращения: 25.05.2021).

18. Uncertainty in management [Электронный ресурс]. URL: <http://kolibri.teacherinabox.org.au/modules/en-boundless/www.boundless.com/management/definition/uncertainty/index.html> (дата обращения: 25.05.2021).

19. What is a Gantt chart? [Электронный ресурс]. URL: <https://www.apm.org.uk/resources/find-a-resource/gantt-chart/#:~:text=A%20Gantt%20chart%20is%20a,useful%20for%20simplifying%20complex%20projects.&text=As%20it's%20in%20a%20bar,progress%20with%20a%20quick%20glance> (дата обращения: 25.05.2021).

20. What is Project Evaluation Review Technique [Электронный ресурс]. URL: https://corporatefinanceinstitute.com/resources/knowledge/other/project-evaluation-review-technique-pert_ (дата обращения: 25.05.2021).

Приложение А

Фрагмент программного кода серверной части приложения

```
*** / function(module, exports, __webpack_require__) {
    'use strict';
    __webpack_require__(1);
    var _fs = __webpack_require__(13);
    var _fs2 = _interopRequireDefault(_fs);
    var _path = __webpack_require__(12);
    var _path2 = _interopRequireDefault(_path);
    var _electron = __webpack_require__(14);
    var _loginWindow = __webpack_require__(15);
    var _loginWindow2 = _interopRequireDefault(_loginWindow);
    var _mainWindow = __webpack_require__(16);
    var _mainWindow2 = _interopRequireDefault(_mainWindow);
    var _helpers = __webpack_require__(24);
    var _helpers2 = _interopRequireDefault(_helpers);
    var _inferFlash = __webpack_require__(29);
    var _inferFlash2 = _interopRequireDefault(_inferFlash);
    var _electronDl = __webpack_require__(30);
    var _electronDl2 = _interopRequireDefault(_electronDl);
    function _interopRequireDefault(obj) { return obj && obj.__esModule ?
obj : { default: obj }; }
    var isOSX = _helpers2.default.isOSX; (0, _electronDl2.default)();
    var APP_ARGS_FILE_PATH = _path2.default.join(__dirname, '..',
'nativefier.json');
    var appArgs = JSON.parse(_fs2.default.readFileSync(APP_ARGS_FILE_PATH, 'utf8'));
    var mainWindow = void 0;
    if (typeof appArgs.flashPluginDir === 'string') {
```

Продолжение Приложения А

```
    _electron.app.commandLine.appendSwitch('ppapi-flash-path',
appArgs.flashPluginDir);
  } else if (appArgs.flashPluginDir) {
    var flashPath = (0, _inferFlash2.default)();
    _electron.app.commandLine.appendSwitch('ppapi-flash-path',
flashPath);
  }
  if (appArgs.ignoreCertificate) {
    _electron.app.commandLine.appendSwitch('ignore-certificate-errors');
  }
  var setDockBadge = function setDockBadge() {};
  if (isOSX()) {
    setDockBadge = _electron.app.dock.setBadge;
  }
  _electron.app.on('window-all-closed', function () {
    if (!isOSX() || appArgs.fastQuit) {
      _electron.app.quit();
    }
  });
  _electron.app.on('activate', function (event, hasVisibleWindows) {
    if (isOSX()) {
      // this is called when the dock is clicked
      if (!hasVisibleWindows) {
        mainWindow.show();
      }
    }
  });
  _electron.app.on('before-quit', function () {
```

Продолжение Приложения А

```
    if (isOSX()) {
        // need to force a quit as a workaround here to simulate the osx app
        // hiding behaviour
        //          Somehow          sokution          at
        // https://github.com/atom/electron/issues/444#issuecomment-76492576 does not work,
        // e.preventDefault appears to persist
        // might cause issues in the future as before-quit and will-quit events
        // are not called
        _electron.app.exit(0);
    }
});
_electron.app.on('ready', function () {
    mainWindow = (0, _mainWindow2.default)(appArgs,
    _electron.app.quit, setDockBadge);
});
_electron.app.on('login', function (event, webContents, request, authInfo,
callback) {
    // for http authentication
    event.preventDefault();
    (0, _loginWindow2.default)(callback);
});
_electron.ipcMain.on('notification', function () {
    if (!isOSX() || mainWindow.isFocused()) {
        return;
    }
    setDockBadge('●');
});
/***/ }
```