

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ	4
ВВЕДЕНИЕ	5
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ РАЗРАБОТКИ, СРАВНЕНИЕ С АНАЛОГАМИ	6
1.1 Описание предметной области с анализом существующих ре- шений и их недостатков исходя из целей работы и критериев сравнения, которые должны соответствовать искомым реше- ниям поставленных задач.	6
1.2 Разработка возможных направлений проведения исследова- ний и методов решений отдельных задач, обоснование выбора оптимального варианта	6
1.3 Обоснование применяемых технологий и инструментов. . . .	9
1.4 Уточненная постановка задачи и требования к прототипу ре- шения. Выводы	11

ПЕРЕЧЕНЬ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

ПО – программное обеспечение

EAM – Enterprise Asset Management

ВВЕДЕНИЕ

Актуальность.

ЕАМ системы - чрезвычайно важная часть производственного процесса предприятий. Это инструмент управления производительностью, рисками и затратами, связанными с физическими (производственными) активами организаций.

Для современных производственных процессов актуальными являются задачи оптимизации: оптимизация маршрутов логистики, рабочего времени, производственных затрат и т. д. Масштаб таких оптимизационных задач может быть непосилен для решения сугубо человеческими ресурсами, даже при использовании ЕАМ систем, поэтому необходимо включать в область решения данных задач программно-аппаратные комплексы, решающие подобные задачи. **рассказать про индастри 4.0 что-нибудь**

Степень разработанности темы, научно-техническая проблема.

?

Объект работы

?

Цель работы и решаемые задачи исходя из темы.

Цель данной работы - создание(?) открытого мультиплатформенного решения, предназначенного для ресурсного планирования в системах управления физическими активами. Решаемые задачи - ?

Планируемый н/т и практический уровень разработки, связь данной работы с другими.

?

Методы или методология проведения работы.

?

Структура работы.

?

1 Анализ предметной области разработки, сравнение с аналогами

1.1 Описание предметной области с анализом существующих решений и их недостатков исходя из целей работы и критериев сравнения, которые должны соответствовать искомому решению поставленных задач.

Одной из областей в операционных исследованиях является составление и оптимизация расписаний. Когда количество работ, человеческих ресурсов и различных механизмов невелико, то составление расписания вполне посильная для человека задача. Но когда количество объектов, учитываемых при планировании, возрастает, например, до уровня предприятия, то задача перестаёт быть тривиальной даже для целого отдела планирования. В таких случаях становится необходимым использование средств автоматизации планирования, примером которых являются мультиагентные системы или программные комплексы комбинаторной оптимизации - отдельные или в рамках ЕАМ-систем.

Про МАС ничего; Рассказываю про существующие на рынке решения аналоги Ор-Тулса, про опенсорс, привожу таблички сравнительные, таблички бенчмарков из статей,

1.2 Разработка возможных направлений проведения исследований и методов решений отдельных задач, обоснование выбора оптимального варианта

В целом можно разделить существующие методы построения и оптимизации подобных моделей на два основных подтипа - Мультиагентные системы (МАС) и Исследование операций (Operations research - далее OR) в рамках целочисленного программирования (программирования в ограничениях).

МАС

Мультиагентная система – система, состоящая из множества взаимодействующих друг с другом агентов. Каждый агент в такой системе независим и не имеет представления о системе в целом, но знает свои задачи и потребности, а также задачи и потребности агентов на одном с ним уровне. Для простоты понимания можно условно сказать, что агенты представляют собой нечто схожее с NPC (Неигровой персонаж (от англ. Non-Player Character)) в компьютерных играх.

Мультиагентные системы начали широко развиваться в девяностых годах и достигли своего пика к середине двухтысячных годов, после чего парадигма создания агентных сетей как чего-то, требующего отдельных фреймворков умерла в связи развитием и упрощением программирования в целом.

Согласно статьи Virginia Dignum и Frank Dignum, [1] несмотря на существование целых организаций по изучению мультиагентных систем, ни за время существования этих организаций, ни за время их работы, МАС не смогли проявить себя ни в одной из заявленных областей, кроме как моделирования поведения персонажей в компьютерных играх [2]. Исследователи связывают неудачу МАС со слабой способностью конкурировать с другими компьютерными технологиями того времени, в первую очередь вызванной тем, что из-под пера ассоциации или заинтересованных исследователей так и не вышло платформы МАС, которую могли бы использовать все желающие за пределами «закрытого» сообщества. Второй проблемой при попытке применить исследования агентов на практике, по их мнению, стала неспособность охватить или учесть существенные характеристики проблемы, для решения которой используются агенты. Аналогичные выводы были представлены Городецким и Скобелевым в статье 2017 года «МНОГО-АГЕНТНЫЕ ТЕХНОЛОГИИ ДЛЯ ИНДУСТРИАЛЬНЫХ ПРИЛОЖЕНИЙ: РЕАЛЬНОСТЬ И ПЕРСПЕКТИВА» [3].

В целом всё-таки можно сказать, что МАС – достаточно перспективное направление, второе дыхание которому могут дать современные разработки в области искусственного интеллекта, нейросетей и программирования.

Главным плюсом (из которого в общих случаях следует и главный минус) мультиагентных систем является скорость реакции на изменения в системе и внешние факторы. Предполагается, что такая система фактически моментально должна реагировать на «происшествия» и сразу же выдавать удовлетворяющее решение с приемлемым качеством. Данное свойство мультиагентных систем делает их максимально пригодными для использования в качестве систем решения задач реального времени или в качестве модели-двойника. Под моделью-двойником представляется виртуальный клон реальной системы, где все объекты взаимодействия являются агентами. Предполагается, что такие модели будут использоваться, например, для симулирования внештатных ситуаций в системе и оценки реагирования системы и агентов на эти ситуации.

Так как МАС предлагают в целом «приемлемые решения», то в областях, где важна каждая мелочь и требуется наиболее полный точный результат, такой подход не подойдет. МАС вполне можно использовать на этапах конкретной эксплуатации, где как раз необходимы оперативные принятия решений, а вот в задаче установки оптимальных начальных условий такой подход скорее всего будет неприменим.

Исследование операций

Для получения наиболее точных и оптимальных решений применяется дисциплина под названием «Исследование операций». Согласно Википедии [4], «ИО - дисциплина, занимающаяся разработкой и применением методов нахождения оптимальных решений на основе математического моделирования, статистического моделирования и различных эвристических подходов в различных областях человеческой деятельности. Иногда используется название математические методы исследования операций.»

Методы исследования операций являются основными в решениях таких задач как планирование, транспортные потоки, маршрутизация, упаковка и задачи о назначениях. В варианте, представленном в статье, предлагается использовать методы комбинаторной оптимизации, а именно – целочисленное программирование в ограничениях.

Программирование в ограничениях — это теория решения комбинаторных задач, опирающаяся на большое количество знаний из областей искусственного интеллекта, информатики и исследования операций. В программировании в ограничениях декларативно задаются ограничения для набора переменных входных данных. Ограничения не определяют последовательность действий для получения искомого результата, а указывают свойства и особенности для нахождения конечного результата.

Оптимизация с ограничениями, или программирование с ограничениями (англ.- *constraint programming*, далее CP), — это название, данное идентификации выполнимых решений из очень большого набора возможных решений. CP основывается на выполнимости (нахождении выполнимого решения), а не на оптимизации (нахождении оптимального решения) и фокусируется на ограничениях и переменных, а не на целевой функции. Фактически, задача CP может даже не иметь целевой функции - цель может заключаться просто в том, чтобы сузить большое множество возможных решений до более управляемого подмножества путем добавления ограничений к задаче.

1.3 Обоснование применяемых технологий и инструментов.

Реализовывать различные методы комбинаторной оптимизации представляется достаточно сложной задачей, поэтому предполагается использование готовых библиотек и фреймворков, реализующих методы исследования операций.

Одними из самых популярных фреймворков для решения задач исследования операций являются открытые GLPK [5], LP_Solve [6], MiniZinc [7], Google OR-Tools [8], CHOCO [9] и проприетарные IBM CPLEX [10] и Gurobi [11].

По результатам бенчмарков [12] [13] можно сказать, что между фреймворками Google OR-Tools, IBM CPLEX и Gurobi существует условный паритет по скорости и качеству решения оптимизационных задач, и в целом предугадать, кто окажется впереди в условиях реальной задачи невозможно.

Одним из наиболее доступных фреймворков является Google Or-Tools. OR-Tools - это программное обеспечение с открытым исходным кодом для комбинаторной оптимизации, которая направлена на поиск наилучшего решения проблемы из очень большого набора возможных решений. Последние несколько лет именно OR-Tools среди всех открытых библиотек занимает первые места в соревнованиях MiniZinc Challenge по программированию в ограничениях [14]. Также, в отличие от большинства остальных фреймворков имеет большее количество официально написанных интерфейсов под разные языки программирования – C++ (изначально написан именно на нём), C#, Java, Python.

Фреймворк Google OR-Tools поддерживает решатели под задачи линейной оптимизации (решения проблемы, смоделированной как набор линейных зависимостей.), целочисленной оптимизации (Смешанное целочисленное программирование - Mixed Integer Programming – MIP; некоторые переменные обязательно представляют собой целые числа) и программирование в ограничениях. Все эти решатели могут быть использованы для решения таких задач, как: решение sudoku, диета Стиглера[15], шахматные задачи, планирование, составление оптимальных маршрутов и т.д.

В результате обзора и анализа методов решения поставленной задачи, на данный момент наиболее подходящим методом можно признать метод комбинаторной оптимизации и программирования в ограничениях, так как явно предоставляет набор устоявшихся практик и решений, показавших себя во множестве проектов.

Основным средством предлагается использовать фреймворк Google OR-Tools, решающий задачи комбинаторной оптимизации, так как данный фреймворк является бесплатным, открытым, может быть использован в коммерческой разработке и покрывает большую часть областей и задач исследования операций.

Мультиагентные системы на данный момент, напротив, по результатам анализа показали себя непригодными к повсеместному использованию, так

как не предоставляют доступных средств для реализации и не были широко протестированы в реальных условиях.

1.4 Уточненная постановка задачи и требования к прототипу решения. Выводы

Имеются следующие начальные условия:

- Есть множество запланированных работ (ЗР) на заданном временном интервале. Каждая работа имеет расчетные плановую дату/время начала выполнения и плановую дату/время завершения выполнения (разница между этими датами определяет длительность работы).
- Каждая работа имеет перечень исполнителей и механизмов, необходимых для ее выполнения (каждый из этих перечней может быть пустым). Для каждого исполнителя и механизма задана потребность, выраженная в чел.*час. (либо, аналогично, машино*час.). В качестве исполнителя (механизма) может быть указан либо конкретный исполнитель (штатная единица (ШЕ) – для человека или конкретный механизм), либо требуемая квалификация (профессия) исполнителя или необходимый тип механизма.
- Имеется подразделение (цех, участок и т.п.), располагающее определенными людскими ресурсами и определенным набором механизмов. Состав людских ресурсов определяется штатным расписанием (оно имеет структуру), перечень доступных механизмов также задан в этой структуре. Каждая ШЕ (людской ресурс) в составе штатного расписания имеет свою профессию, а механизм – соответствующий ему тип механизма.

Задача:

На основании имеющихся исходных данных постараться построить такой порядок выполнения работ, чтобы в каждый день суммарная потребность в людских ресурсах и механизмах на выполнение всех работ не превышала

объем располагаемых ресурсов и механизмов, заданных в штатном расписании.

Для того, чтобы найти такое решение, система может перемещать исходные ЗР во времени в заданных пределах. При этом должны соблюдаться следующие условия:

- длительность работ и потребность в ресурсах, указанные для каждой ЗР, должны сохраняться;
- порядок выполнения работ одного типа и для одного и того же объекта должен сохраняться;
- могут быть заданы ограничения по изменению сроков выполнения работ: например, нельзя изменять сроки более, чем на 7 дней в любом направлении, либо можно изменять не более, чем на 10 дней в сторону уменьшения, но не более, чем на 2 дня в сторону увеличения.

Результат:

В результате система должна предложить новые плановые даты выполнения каждой ЗР при условии соблюдения заданных ограничений. Если решения нет, то система должна сообщить об этом, желательно при этом указать, какого ресурса или механизма (по мнению системы) недостаточно для выполнения плана.