ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ПЕТРА ВЕЛИКОГО»**
ФИЗИКО-МЕХАНИЧЕСКИЙ ИНСТИТУТ
ВЫСШАЯ ШКОЛА ПРИКЛАДНОЙ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ФИЗИКИ

Отчет о прохождении преддипломной практики
на тему: «Децентрализованное решение задачи о назначениях целей группе роботов»

Кромачев Максим Александрович , гр. 5030102/10201

Направление подготовки: 01.03.02 Прикладная математика и информатика.
Место прохождения практики: СПБПУ, ФизМех, ВШПМиВФ.
Сроки практики: с 02.02.25 по 28.05.25.
Руководитель практики от ФГАОУ ВО «СПбПУ»: Фамилия Имя Отчество, должность, степень.
Консультант практики от ФГАОУ ВО «СПбПУ»: Фамилия Имя Отчетство, должность, степень.
Оценка: __________

Руководитель практики
от ФГАОУ ВО «СПбПУ» И.О. Фамилия

Консультант практики
от ФГАОУ ВО «СПбПУ» И.О. Фамилия

Обучающийся И.О. Фамилия

Дата: дд.мм.гггг

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Развитие робототехники и распределенных систем управления в последние десятилетия привело к значительному росту интереса к задачам оптимального распределения ресурсов в условиях ограниченной коммуникации. Такие задачи, известные как задачи назначения, требуют эффективного распределения ограниченного набора ресурсов (например, задач или объектов) между агентами (роботами) для максимизации общей выгоды. В условиях ограниченных областей связи, когда не все роботы могут взаимодействовать друг с другом или отсутствует единый центр управления, традиционные алгоритмы, такие как венгерский метод, сталкиваются с вычислительными трудностями. Это подчеркивает необходимость разработки новых алгоритмов, способных учитывать топологические ограничения и обеспечивать высокую эффективность.

Актуальность исследования обусловлена растущей потребностью в автоматизации сложных систем управления в робототехнике, где ограниченные области связи создают дополнительные проблемы для координации роботов. По данным исследований [1], аукционные алгоритмы демонстрируют высокую эффективность для задач назначения, однако их применение в робототехнике с учетом топологических ограничений остается недостаточно изученным. Это определяет актуальность данной работы, направленной на разработку и исследование новых подходов к решению задач назначения.

Объект исследования — распределенные системы управления роботами, функционирующие в условиях ограниченной коммуникации.

Предмет исследования — алгоритмы назначения, обеспечивающие оптимальное распределение задач между роботами с учетом ограниченных областей связи.

Цель исследования — разработать модифицированный аукционный алгоритм для задачи назначения в распределенных системах роботов с ограниченными областями связи, исследовать его и сравнить его эффективность с венгерским алгоритмом.

Задачи исследования:

A. Изучить существующие алгоритмы решения задачи назначения, включая аукционный и венгерский методы.

B. Проанализировать влияние ограниченных областей связи на эффективность алгоритмов назначения.

C. Разработать модификацию аукционного алгоритма, учитывающую топологические ограничения в системах роботов.
D. Реализовать предложенный алгоритм и венгерский алгоритм в программной среде.
E. Провести сравнительное тестирование алгоритмов на модельных задачах с различными характеристиками коммуникационных ограничений.

Теоретическая база исследования включает работы по задачам назначения и групповому управлению роботами. Основой послужили исследования Д. Бертсекаса [1], описывающие аукционный алгоритм и его преимущества, а также работы Х. Куна [2] по венгерскому методу. Значительное внимание уделено работам по распределённым системам управления [3; 4], а также проблемам оптимального распределения задач в мультиробототных системах [5]. В процессе подготовки исследования были изучены такие дисциплины, как «Алгоритмы и структуры данных», «Робототехника» и «Теория оптимизации».

Методологическая база включает общенаучные методы (анализ, моделирование, эксперимент) и конкретно-научные методы (методы линейного программирования, аукционные методы, методы графов). В работе применены подходы параллельных вычислений, описанные в [1], а также методы сравнительного анализа алгоритмов.

Информационная база включает материалы учебных дисциплин, данные из научных публикаций, а также результаты моделирования, полученные в ходе выполнения данной ВКР.

Степень научной разработанности проблемы характеризуется значительным вниманием к задачам назначения в работах Д. Бертсекаса, Х. Куна и других авторов [1; 2; 5; 6]. Исследования группового управления роботами, включая работы В.Х. Пшихопова [3] и И.А. Каляева [4], подчеркивают важность учета коммуникационных ограничений. Однако адаптация аукционных алгоритмов к ограниченным областям связи в робототехнике остается малоисследованной, что определяет необходимость разработки новых подходов.

Научная новизна заключается в разработке модифицированного аукционного алгоритма, адаптированного для распределенных систем роботов с ограниченными областями связи. Новизна проявляется в учете топологических ограничений коммуникации, что позволяет повысить эффективность алгоритма по сравнению с традиционными методами. Также новизна состоит в сравнительном анализе аукционного и венгерского алгоритмов в контексте робототехнических приложений.

Практическая значимость заключается в возможности применения разработанного алгоритма для оптимизации распределения задач в системах управления роботами, работающих в условиях ограниченной коммуникации, таких как складская логистика, автономные транспортные системы и роевые робототехнические платформы. Результаты работы могут быть использованы для повышения эффективности реальных систем и дальнейших исследований в области распределенного управления.

Апробация результатов включает представление результатов на защите ВКР и размещение работы на портале СПбПУ.

Введение определяет рамки исследования, связывая поставленные задачи с главами работы. Первая глава посвящена анализу существующих алгоритмов назначения, вторая — разработке модифицированного аукционного алгоритма, третья — программной реализации, четвертая — тестированию и сравнению с венгерским алгоритмом.

ГЛАВА 1. АНАЛИЗ СУЩЕСТВУЮЩИХ АЛГОРИТМОВ НАЗНАЧЕНИЯ

1.1. Введение в предметную область

Задача назначения — одна из ключевых проблем комбинаторной оптимизации, широко применяемая в робототехнике, логистике и управлении ресурсами. Согласно [1], задача заключается в распределении n агентов (например, роботов) по n объектам (например, задачам) для максимизации суммарной выгоды, заданной матрицей $\{a_{ij}\}$, где a_{ij} — выгода от назначения агента i объекту j. Математически цель задачи:

$$\sum_{i=1}^{n} a_{ij_i} \to \max,$$

где j_i — объект, назначенный агенту i, и все j_i различны.

В робототехнике задача назначения актуальна для распределения задач между роботами в условиях ограниченной коммуникации, когда роботы обмениваются информацией только с соседями. Такие ограничения, обусловленные топологией сети, требуют алгоритмов, эффективных в распределенных системах. Примеры приложений: складская логистика, автономные транспортные системы, роевые платформы [4; 5].

Цель главы — проанализировать аукционный и венгерский алгоритмы, включая их математические основы, сходимость и оптимальность, и оценить их применимость к робототехнике. Анализ обосновывает необходимость модифицированного аукционного алгоритма для ограниченной коммуникации.

1.2. Аукционный алгоритм

Аукционный алгоритм, предложенный Д. Бертсекасом [1], представляет собой итеративный метод для решения задачи назначения, моделирующий процесс аукциона, где агенты делают ставки на объекты, а цены корректируются для достижения оптимального распределения.

1.2.1. Постановка задачи

Задача назначения (Assignment Problem) заключается в нахождении оптимального соответствия между n агентами и n объектами с учетом матрицы выгод $\{a_{ij}\}$, где a_{ij} — выгода агента i при назначении ему объекта j. Цель — максимизировать суммарную выгоду:

$$\max \sum_{i=1}^{n} a_{ij_i},$$

где j_i — объект, назначенный агенту i, при условии, что каждые агент и объект участвуют ровно в одном назначении (биективное соответствие). Эта задача эквивалентна задаче линейного программирования с ограничениями:

$$\begin{cases} \sum_{i=1}^{n} x_{ij} = 1, & \forall j = 1, \dots, n, \\ \sum_{j=1}^{n} x_{ij} = 1, & \forall i = 1, \dots, n, \\ x_{ij} \in \{0, 1\}, & \forall i, j, \end{cases}$$

где $x_{ij} = 1$, если агент i назначен объекту j, и $x_{ij} = 0$ в противном случае. Аукционный алгоритм решает эту задачу приближенно с заданной точностью $\varepsilon > 0$, достигая почти оптимального решения через итеративную корректировку цен.

1.2.2. Описание алгоритма

Согласно [1], алгоритм работает с матрицей выгод $\{a_{ij}\}$ и начинается с произвольного распределения и цен $\{p_j\}$. Ключевые определения:

– *Почти счастье*: Агент i, назначенный объекту j_i, почти счастлив, если:

$$a_{ij_i} - p_{j_i} \geqslant \max_{j=1,\dots,n} \{a_{ij} - p_j\} - \varepsilon,$$

где $\varepsilon > 0$ — параметр точности.

– *Почти равновесие*: Распределение и цены, при которых все агенты почти счастливы.

Шаги алгоритма [1]:

A. Проверить, все ли агенты почти счастливы. Если да, завершить.

B. Выбрать агента i, не почти счастливого, и найти объект j_i:

$$j_i \in \arg\max_{j=1,\dots,n} \{a_{ij} - p_j\}.$$

C. Переназначить: агент i получает j_i, а агент, ранее назначенный на j_i, получает объект, принадлежавший i.

D. Увеличить цену p_{j_i} на:

$$\gamma_i = v_i - w_i + \varepsilon,$$

где $v_i = \max_j\{a_{ij} - p_j\}$, $w_i = \max_{j \neq j_i}\{a_{ij} - p_j\}$.

E. Повторить с шага 1.

1.2.3. Сходимость алгоритма

Теорема 1.1 (Сходимость аукционного алгоритма [1])**.** *Алгоритм завершается за конечное число шагов с распределением и ценами, почти в равновесии.*

Доказательство. Если объект j получает ставку, агент i, назначенный на j, становится почти счастливым, так как j максимизирует $a_{ij} - p_j$. После увеличения p_j на $\gamma_i = v_i - w_i + \varepsilon$:

$$a_{ij} - (p_j + \gamma_i) = a_{ij} - p_j - (v_i - w_i + \varepsilon) = w_i - \varepsilon \leqslant \max_{k \neq j}\{a_{ik} - p_k\} - \varepsilon,$$

что удовлетворяет условию почти счастья. Агент i остается почти счастливым, пока удерживает j, так как цены p_k $(k \neq j)$ не уменьшаются, а $a_{ij} - p_j$ может только уменьшаться.

Агенты, не почти счастливые, назначены на объекты без ставок. Если все объекты получат хотя бы одну ставку, все агенты станут почти счастливы, и алгоритм завершится. Предположим, некоторые объекты никогда не получают ставок. Объект j с m ставками имеет цену $p_j \geqslant m\varepsilon$. При большом m $a_{ij} - p_j$ становится меньше $a_{ik} - p_k$ для объекта k без ставок ($p_k = 0$), и агент выберет k. Таким образом, все объекты получат ставку, или алгоритм завершится ранее, когда все агенты почти счастливы. Конечность шагов следует из конечности числа объектов.

1.2.4. Оценка итераций

Утверждение 1.1 (Оценка итераций [1])**.** *Количество итераций пропорционально C/ε, где $C =$* $\max_{i,j} |a_{ij}|$.

Доказательство. Каждая ставка увеличивает цену объекта минимум на ε. Максимальная выгода C ограничивает рост цен. В худшем случае цены достигают порядка C, и число ставок (итераций) составляет $O(C/\varepsilon)$.

1.2.5. Оптимальность

Теорема 1.2 (Оптимальность аукционного алгоритма [1])**.** *Если алгоритм завершается с почти равновесным распределением, суммарная выгода находится в пределах $n\varepsilon$ от оптимальной. При целочисленных a_{ij} и $\varepsilon < 1/n$ распределение оптимально.*

Доказательство. Оптимальная выгода (примитивная задача):

$$A^* = \max_{\{k_i\}} \sum_{i=1}^{n} a_{ik_i}, \quad k_i \neq k_m \text{ для } l \neq m.$$

Двойственная задача:

$$D^* = \min_{p_j} \left\{ \sum_{j=1}^{n} p_j + \sum_{i=1}^{n} \max_j \{a_{ij} - p_j\} \right\}.$$

Для любого распределения $\{(i, k_i)\}$ и цен $\{p_j\}$:

$$\sum_{i=1}^{n} a_{ik_i} \leqslant \sum_{j=1}^{n} p_j + \sum_{i=1}^{n} \max_j \{a_{ij} - p_j\},$$

так как $\max_j \{a_{ij} - p_j\} \geqslant a_{ik_i} - p_{k_i}$. Следовательно, $A^* \leqslant D^*$.
При почти равновесии для распределения $\{(i, j_i)\}$:

$$a_{ij_i} - p_{j_i} \geqslant \max_j \{a_{ij} - p_j\} - \varepsilon.$$

Суммируем по i:

$$\sum_{i=1}^{n} (a_{ij_i} - p_{j_i}) \geqslant \sum_{i=1}^{n} \max_j \{a_{ij} - p_j\} - n\varepsilon.$$

Добавим $\sum_{j=1}^{n} p_j$:

$$\sum_{i=1}^{n} a_{ij_i} \geqslant \sum_{j=1}^{n} p_j + \sum_{i=1}^{n} \max_j \{a_{ij} - p_j\} - n\varepsilon.$$

Поскольку $D^* \leqslant \sum_{j=1}^n p_j + \sum_{i=1}^n \max_j\{a_{ij} - p_j\}$, то:

$$\sum_{i=1}^{n} a_{ij_i} \geqslant D^* - n\varepsilon \geqslant A^* - n\varepsilon.$$

Так как $\sum_{i=1}^n a_{ij_i} \leqslant A^*$, получаем:

$$A^* - n\varepsilon \leqslant \sum_{i=1}^{n} a_{ij_i} \leqslant A^*.$$

Для целочисленных a_{ij} и $\varepsilon < 1/n$, $A^* - \sum_{i=1}^n a_{ij_i} < 1$, и, так как разность целочисленная, $A^* = \sum_{i=1}^n a_{ij_i}$.

1.2.6. Свойства, преимущества и недостатки

Свойства [1]:

- Сходимость за конечное число шагов: алгоритм завершается, как только все объекты получают хотя бы одну ставку или все люди становятся почти счастливыми (теорема 1.1).
- Оптимальность в пределах $n\varepsilon$: итоговая выгода назначения находится в пределах $n\varepsilon$ от оптимального значения, а при $\varepsilon < 1/n$ и целых a_{ij} назначение оптимально (теорема 1.2).
- Производительность зависит от начальных цен: цены, близкие к оптимальным, значительно сокращают число раундов, особенно при использовании ε-масштабирования.

Преимущества:

- Интуитивная экономическая модель: алгоритм имитирует реальный аукцион, упрощая понимание и интерпретацию.
- Гибкость для адаптации: легко модифицируется для асимметричных задач, транспортных проблем и задач минимальной стоимости потока [1].
- Хорошо подходит для параллельных и распределенных систем: допускает асинхронные и параллельные реализации, эффективен для разреженных задач [5].

Недостатки:

- Число итераций зависит от C/ε: для больших $C = \max_{i,j} |a_{ij}|$ и малого ε требуется больше раундов, примерно пропорционально C/ε.

- Чувствительность к выбору ε: большое ε снижает точность (итоговая выгода дальше от оптимальной), малое ε увеличивает число итераций.
- Чувствительность к начальным ценам: плохие начальные цены замедляют сходимость, хотя ε-масштабирование смягчает эту проблему.

1.3. Венгерский алгоритм

Венгерский алгоритм, разработанный Х. Куном [2], решает задачу назначения через редукцию матрицы выгод или эквивалентную задачу максимального паросочетания в двудольном графе [7].

1.3.1. Описание через матрицу

Алгоритм работает с матрицей выгод $\{a_{ij}\}$ размера $n \times n$ [2]:

A. Для каждого ряда вычесть минимальный элемент: $a_{ij} \leftarrow a_{ij} - \min_j a_{ij}$.
B. Для каждого столбца вычесть минимальный элемент: $a_{ij} \leftarrow a_{ij} - \min_i a_{ij}$.
C. Найти минимальное число строк и столбцов, покрывающих все нули.
D. Если число линий равно n, найти назначение (нули соответствуют оптимальным парам) и завершить.
E. Иначе найти минимальный непокрытый элемент δ, вычесть δ из непокрытых элементов, прибавить к элементам на пересечении линий, вернуться к шагу 3.

1.3.2. Графовый подход

Как указано в [7], задача назначения эквивалентна нахождению максимального паросочетания в двудольном графе $G = (V_1 \cup V_2, E)$, где V_1 — агенты, V_2 — объекты, а ребра (i, j) имеют вес a_{ij}. Алгоритм:

A. Построить матрицу $\{a_{ij}\}$ и редуцировать ее, как выше.
B. Сформировать граф, где ребра соответствуют нулям в матрице.
C. Найти максимальное паросочетание (например, с помощью алгоритма Куна или Форда-Фалкерсона).
D. Если паросочетание покрывает n вершин, оно оптимально. Иначе корректировать матрицу (δ) и обновить граф.

1.3.3. Доказательство оптимальности

Алгоритм основан на двойственности линейного программирования [2]. Примитивная задача:

$$\max \sum_{i,j} a_{ij} x_{ij}, \quad \sum_{j} x_{ij} = 1, \sum_{i} x_{ij} = 1, x_{ij} \geqslant 0.$$

Двойственная задача:

$$\min \sum_{i} u_i + \sum_{j} v_j, \quad u_i + v_j \geqslant a_{ij},$$

где u_i, v_j — двойственные переменные. Редукция матрицы и корректировка δ обеспечивают выполнение условий $u_i + v_j \geqslant a_{ij}$, а равенство $u_i + v_j = a_{ij}$ для выбранных пар дает оптимальность.

1.3.4. Свойства, преимущества и недостатки

Свойства [2; 7]:

- Сходимость за конечное число шагов.
- Оптимальность: всегда находит максимальную выгоду.
- Сложность: $O(n^3)$ для матричной реализации.

Преимущества:

- Гарантированная оптимальность.
- Простота реализации [7].
- Поддержка в библиотеках.

Недостатки:

- Сложность $O(n^3)$ для больших n.
- Требует полной матрицы $\{a_{ij}\}$, что проблематично при ограниченной коммуникации [5].
- Нет параллелизма [6].

1.4. Применение алгоритмов в робототехнике

В робототехнике задача назначения используется для распределения задач между роботами в условиях ограниченной коммуникации [3; 4]. Аукционный алгоритм [1] поддерживает параллелизм, что подходит для роевых систем, но не учитывает топологические ограничения [5]. Венгерский алгоритм [2; 7] эффективен при полном доступе к данным, но неприменим без центрального управления [4]. Исследования [3; 5] подчеркивают необходимость адаптации алгоритмов для динамических сетей.

1.5. Постановка проблемы

Аукционный алгоритм [1] не учитывает ограничения связи, снижая эффективность в распределенных системах [5]. Венгерский алгоритм [2; 7] требует полной матрицы выгод, что неосуществимо без централизации [4]. Необходим модифицированный аукционный алгоритм, который:

- Учитывает ограниченные области связи.
- Сохраняет параллелизм.
- Обеспечивает эффективность и близость к оптимальности.

Цель — разработать такой алгоритм и сравнить с венгерским методом.

1.6. Выводы

Рассмотрены аукционный [1] и венгерский [2; 7] алгоритмы. Аукционный алгоритм поддерживает параллелизм, но требует адаптации для ограниченной коммуникации. Венгерский алгоритм оптимален, но непригоден для распределенных систем. Необходим модифицированный аукционный алгоритм, учитывающий топологические ограничения.

ГЛАВА 2. РАЗРАБОТКА МОДИФИЦИРОВАННОГО АУКЦИОННОГО АЛГОРИТМА

В данной главе представлена разработка модифицированного аукционного алгоритма для распределенных систем роботов с ограниченными областями связи. На основе анализа аукционного алгоритма Бертсекаса [1] предлагается модификация, учитывающая топологические ограничения сети, при которых роботы обмениваются информацией только с соседями, объединенными в группы на основе радиуса связи. Каждый робот имеет доступ ко всем задачам, что упрощает задачу назначения, но требует координации внутри групп. Описывается математическая модель задачи с учетом локальной доступности данных, а также алгоритмические изменения, обеспечивающие параллельную обработку и устойчивость к ограниченной коммуникации. Формулируются гипотезы об эффективности и близости решения к оптимальному.

2.1. Математическая модель задачи

Задача назначения формулируется следующим образом. Дано множество из n роботов $R = \{r_1, r_2, \ldots, r_n\}$ и множество из m задач $T = \{t_1, t_2, \ldots, t_m\}$. Каждый робот r_i имеет координаты $p_i = (x_i, y_i)$, а каждая задача t_j — координаты $q_j = (x_j, y_j)$. Выгода от назначения робота r_i задаче t_j задана матрицей $\{\alpha_{ij}\}$, где α_{ij} зависит от расстояния между p_i и q_j, например, обратно пропорциональна евклидову расстоянию:

$$d(p_i, q_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}.$$

Все роботы имеют доступ ко всем задачам, поэтому α_{ij} определена для любых i и j.

Матрица видимости роботов $V = \{v_{ij}\}$ определяет топологию сети: $v_{ij} = 1$, если $d(p_i, p_j) \leqslant r_v$ (роботы r_i и r_j могут обмениваться информацией), и $v_{ij} = 0$ в противном случае. Роботы объединяются в группы (связные компоненты графа V), внутри которых проводится аукцион.

Цель задачи — максимизировать суммарную выгоду:

$$\sum_{i=1}^{n} \alpha_{ij_i} \to \max,$$

где j_i — задача, назначенная роботу r_i, при условиях:

- Задача может быть назначена нескольким роботам в процессе аукциона, но в итоговом решении для каждой задачи t_j учитывается только робот с максимальной выгодой α_{ij}, определяемой как наибольшая эффективность (например, минимальное время достижения цели). Остальные назначения для этой задачи отбрасываются.
- Роботы обмениваются информацией о ценах и назначениях только внутри связных компонент графа видимости V.

2.2. Модификация аукционного алгоритма

Модифицированный аукционный алгоритм адаптирует подход Бертсекаса [1] для учета ограниченной коммуникации между роботами. Ключевые изменения включают:

- Полная видимость задач: Все задачи доступны каждому роботу, поэтому α_{ij} определена для всех i, j.
- Разделение на связные компоненты: Роботы группируются по графу видимости V с использованием поиска в ширину, что позволяет проводить аукцион независимо для каждой компоненты.
- Параллельная обработка: Каждая связная компонента обрабатывается одновременно, моделируя распределенную систему.
- Разрешение конфликтов: Если несколько роботов из разных компонент выбирают одну задачу, учитывается только робот с максимальной выгодой α_{ij}, а остальные назначения отбрасываются.

2.2.1. Описание алгоритма

Алгоритм состоит из следующих шагов:

A. **Инициализация**:
   - Задать матрицу выгод $\{\alpha_{ij}\}$, координаты роботов $\{p_i\}$ и задач $\{q_j\}$, радиус видимости r_v.
   - Построить матрицу видимости роботов V: $v_{ij} = 1$, если $d(p_i, p_j) \leqslant r_v$, и 0 иначе.
   - Инициализировать цены задач $\{p_j = 0\}$ и назначения $\{j_i = -1\}$ (роботы изначально не назначены).

B. **Разделение на компоненты**:
   - Используя поиск в ширину на графе V, найти связные компоненты $C_1, C_2, \ldots, C_k$, где роботы в одной компоненте могут обмениваться данными.

C. **Параллельный аукцион**:
   - Для каждой компоненты C_l:
     1. Для каждого робота $r_i \in C_l$ вычислить текущую прибыль: $\alpha_{ij_i} - p_{j_i}$, если $j_i \neq -1$, иначе 0.

2. Проверить условие почти счастья: $\alpha_{ij_i} - p_{j_i} \geqslant \max_j\{\alpha_{ij} - p_j\} - \varepsilon$, где $\varepsilon > 0$.
3. Если робот не почти счастлив, найти задачу t_{j_i}: $j_i = \arg\max_j\{\alpha_{ij} - p_j\}$, включая фиктивную задачу ($\alpha_{i,-1} = 0$).
4. Вычислить $v_i = \max_j\{\alpha_{ij} - p_j\}$, $w_i = \max_{j \neq j_i}\{\alpha_{ij} - p_j\}$.
5. Если $j_i = -1$, переназначить робота на фиктивную задачу без изменения цен.
6. Иначе переназначить: робот r_i получает задачу t_{j_i}, прежний владелец t_{j_i} (если есть) становится неназначенным.
7. Увеличить цену: $p_{j_i} += v_i - w_i + \varepsilon$.
8. Повторять, пока все роботы в C_l не станут почти счастливы.

D. **Разрешение конфликтов**:
   - Для каждой задачи t_j собрать всех роботов $\{r_i\}$, назначенных на t_j из разных компонент.
   - Выбрать робота r_i с максимальной выгодой α_{ij}, определяемой как наибольшая эффективность (например, минимальное время достижения цели). Остальные роботы, назначенные на t_j, переназначаются на фиктивную задачу ($j_i = -1$).

E. **Вычисление результата**:
   - Вычислить суммарную выгоду: $\sum_{i=1}^{n} \alpha_{ij_i}$, где $j_i \neq -1$, учитывая только назначения, выбранные после разрешения конфликтов.
   - Вернуть назначения $\{j_i\}$ и суммарную выгоду.

2.2.2. Сходимость алгоритма

Теорема 2.1 (Сходимость модифицированного аукционного алгоритма)**.** *Модифицированный аукционный алгоритм завершается за конечное число шагов, при котором все роботы в каждой связной компоненте удовлетворяют условию почти счастья, определенному в главе 1.*

Доказательство. Модифицированный алгоритм разделяет роботов на связные компоненты $C_1, C_2, \ldots, C_k$ на основе матрицы видимости V. В каждой компоненте C_l аукцион проводится независимо, следуя структуре оригинального аукционного алгоритма, описанного в главе 1 [1]. Согласно теореме 1.1 (глава 1), оригинальный аукционный алгоритм сходится за конечное число шагов, обеспечивая почти равновесие, при котором все агенты почти счастливы, то есть для каждого агента i, назначенного объекту j_i, выполняется:

$$a_{ij_i} - p_{j_i} \geqslant \max_{j=1,\dots,n} \{a_{ij} - p_j\} - \varepsilon,$$

где $\varepsilon > 0$ — параметр точности, a_{ij} — выгода, p_j — цена объекта.

В модифицированном алгоритме для каждого робота $r_i \in C_l$, назначенного задаче t_{j_i}, обновление цен происходит аналогично: после ставки цена p_{j_i} увеличивается на $\gamma_i = v_i - w_i + \varepsilon$, где $v_i = \max_j\{\alpha_{ij} - p_j\}$, $w_i = \max_{j \neq j_i}\{\alpha_{ij} - p_j\}$. Как показано в доказательстве теоремы 1.1, это обновление обеспечивает, что:

$$\alpha_{ij_i} - (p_{j_i} + \gamma_i) = w_i - \varepsilon \leqslant \max_{k \neq j_i}\{\alpha_{ik} - p_k\} - \varepsilon,$$

удовлетворяя условию почти счастья. Поскольку цены p_k для $k \neq j_i$ не уменьшаются, робот r_i остается почти счастливым, пока удерживает задачу t_{j_i}.

Конечность числа шагов в каждой компоненте следует из теоремы 1.1: число роботов и задач конечно, а цены увеличиваются минимум на ε. Если задача t_j получает m ставок, то $p_j \geqslant m\varepsilon$. При большом m $\alpha_{ij} - p_j$ становится меньше $\alpha_{ik} - p_k$ для задачи k без ставок, побуждая роботов выбрать другие задачи. Параллельная обработка компонент не влияет на сходимость, так как каждая компонента независима. Разрешение конфликтов, при котором для задачи t_j выбирается робот с максимальной выгодой α_{ij}, происходит после завершения аукциона в компонентах и не нарушает почти счастье внутри них, так как переназначение на фиктивную задачу ($j_i = -1$) не требует изменения цен. Таким образом, алгоритм завершается за конечное число шагов, обеспечивая почти равновесие в каждой компоненте.

2.2.3. Оценка итераций

Утверждение 2.1 (Оценка итераций)**.** *Общее число итераций модифицированного аукционного алгоритма ограничено $O\left(\frac{C}{\varepsilon}\right)$, где $C = \max_{i,j} |\alpha_{ij}|$.*

Доказательство. Модифицированный алгоритм выполняет аукцион в каждой связной компоненте C_l, аналогичный оригинальному аукционному алгоритму, описанному в главе 1 [1]. Согласно утверждению 1.1 (глава 1), число итераций в оригинальном алгоритме пропорционально C/ε, где $C = \max_{i,j} |a_{ij}|$. В компоненте C_l с $|C_l|$ роботами и m задачами аукцион следует той же логике: каждая ставка увеличивает цену задачи минимум на ε, и рост цен ограничен $C = \max_{i,j} |\alpha_{ij}|$. Таким образом, число итераций в C_l составляет $O(C/\varepsilon)$. Поскольку компоненты обрабатываются параллельно, общее число итераций определяется максимумом по всем компонентам, то есть $O(C/\varepsilon)$.

2.2.4. Оптимальность

Теорема 2.2 (Оптимальность модифицированного алгоритма)**.** *В каждой связной компоненте C_l модифицированный аукционный алгоритм при целочисленных выгодах α_{ij} и $\varepsilon < 1/n$ даёт оптимальное решение задачи аукциона, как определено в главе 1. Однако общее решение после объединения компонент не гарантирует глобальной оптимальности из-за ограниченной коммуникации.*

Доказательство. В каждой связной компоненте C_l модифицированный алгоритм выполняет аукцион, аналогичный оригинальному, описанному в главе 1 [1]. Для робота $r_i \in C_l$, назначенного задаче t_{j_i}, достигается почти равновесие, где:

$$\alpha_{ij_i} - p_{j_i} \geqslant \max_j \{\alpha_{ij} - p_j\} - \varepsilon.$$

Согласно теореме 1.2 (глава 1), при целочисленных a_{ij} и $\varepsilon < 1/n$ оригинальный аукционный алгоритм даёт оптимальное решение, так как суммарная выгода удовлетворяет:

$$\sum_{i=1}^{n} a_{ij_i} = A^*,$$

где A^* — оптимальная выгода. Аналогично, в компоненте C_l с $|C_l|$ роботами и целочисленными α_{ij}, при $\varepsilon < 1/n$ (где n — общее число роботов) суммарная выгода $\sum_{i \in C_l} \alpha_{ij_i}$ оптимальна для локальной задачи аукциона.

Однако глобальная оптимальность не достигается. После аукциона в компонентах выполняется разрешение конфликтов, где для задачи t_j, назначенной нескольким роботам из разных компонент, выбирается робот с максимальной выгодой α_{ij}, а остальные переназначаются на фиктивную задачу ($\alpha_{i,-1} = 0$). Из-за ограниченной коммуникации между компонентами роботы в C_l не имеют доступа к данным о выгодах роботов из других компонент, что может привести к выбору локально оптимальных, но глобально неоптимальных назначений. Таким образом, суммарная выгода $\sum_{i=1}^{n} \alpha_{ij_i}$ после разрешения конфликтов может быть меньше глобального оптимума A^*.

2.3. Гипотезы об эффективности

Для оценки модифицированного аукционного алгоритма сформулированы следующие гипотезы:

- **Гипотеза 1**: Алгоритм завершается за конечное число шагов, и время выполнения пропорционально C/ε, где $C = \max_{i,j} |\alpha_{ij}|$, с уменьшением времени за счёт параллельной обработки связных компонент.
- **Гипотеза 2**: В каждой связной компоненте при целочисленных выгодах α_{ij} и $\varepsilon < 1/n$ достигается оптимальное решение, но отклонение от глобального оптимума зависит от числа связных компонент k и радиуса видимости r_v.
- **Гипотеза 3**: Алгоритм эффективен в динамических сетях с изменяющимися позициями роботов при периодическом обновлении матрицы видимости V.

2.4. Выводы

Разработан модифицированный аукционный алгоритм, учитывающий ограниченные области связи в распределённых системах роботов. Алгоритм группирует роботов в связные компоненты с помощью матрицы видимости V, проводит аукцион параллельно и обеспечивает полную видимость задач. Доказана сходимость за конечное число шагов (теорема 2.1) и оптимальность в каждой компоненте для целочисленных выгод α_{ij} при $\varepsilon < 1/n$ (теорема 2.2), но глобальная оптимальность не достигается из-за ограниченной коммуникации. Преимущества включают простоту реализации, снижение коммуникационных затрат и адаптивность к реальным сценариям робототехники. Сформулированные гипотезы задают направления для дальнейших исследований эффективности и устойчивости алгоритма.

ГЛАВА 3. РЕАЛИЗАЦИЯ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Глава посвящена реализации модифицированного аукционного алгоритма и венгерского алгоритма для сравнительного анализа. Описывается программная среда, выбранная для реализации, включая язык программирования и инструменты моделирования распределенных систем роботов. Приводится структура программного обеспечения, обеспечивающего тестирование алгоритмов в условиях ограниченной коммуникации. Рассматриваются аспекты оптимизации кода для параллельных вычислений и интеграции с робототехническими платформами.

3.1. Программная среда

Реализация выполнена на языке C++ благодаря его производительности и поддержке параллельных вычислений. Для визуализации результатов использовалась библиотека Qt. Генерация случайных данных реализована с использованием стандартных средств C++. Код совместим с платформами Windows и Unix.

3.2. Структура реализации

Программное обеспечение включает модули для аукционного и венгерского алгоритмов, а также логику тестирования и визуализации результатов. Аукционный алгоритм учитывает ограничения коммуникации, разделяя роботов на группы по связности и назначая задачи с учетом настраиваемого параметра радиуса связи, влияющего на скорость и точность. Венгерский алгоритм решает задачу оптимального назначения, преобразуя её в минимизацию затрат. Тестирование проводится на случайных данных с варьированием размеров задачи (5–300), радиусов связи (1–300) и параметра алгоритма (ε от 10^{-6} до 1).

3.3. Оптимизация

Для повышения производительности применены:

– Параллельные вычисления для обработки независимых групп роботов в аукционном алгоритме.

- Эффективные структуры данных для сокращения затрат памяти и времени.
- Настройка визуализации для наглядного представления результатов.

3.4. Интеграция с робототехническими системами

Реализация может быть адаптирована для робототехнических платформ, таких как ROS, с формированием данных на основе сенсоров. Возможна разработка веб-интерфейса для визуализации распределения задач. Для реальных систем потребуется обработка динамических изменений и сбоев связи.

3.5. Тестирование и визуализация

Тестирование включало сравнение времени выполнения и точности алгоритмов. Результаты представлены графиками, созданными с использованием Qt, что позволило оценить влияние параметров на производительность.

3.6. Выводы

Разработанное программное обеспечение обеспечивает сравнение аукционного и венгерского алгоритмов в условиях ограниченной коммуникации. Реализация в C++ с применением параллельных вычислений демонстрирует высокую производительность и готовность к адаптации для реальных систем.

ГЛАВА 4. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ И АНАЛИЗ РЕЗУЛЬТАТОВ

В четвертой главе представлены результаты экспериментального исследования модифицированного аукционного алгоритма в сравнении с венгерским алгоритмом. Описываются модельные задачи, имитирующие распределенные системы роботов с различной топологией сети связи. Приводятся метрики оценки (время выполнения, точность решения, устойчивость к изменениям сети). Анализируется эффективность предложенного алгоритма, подтверждаются гипотезы, сформулированные в главе 2. Результаты апробированы на научной конференции и через портал СПбПУ.

ГЛАВА 5. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ И АНАЛИЗ РЕЗУЛЬТАТОВ

Глава посвящена направлениям дальнейшего развития модифицированного аукционного алгоритма и его применению в распределенных системах роботов. Рассматриваются возможные улучшения алгоритма, расширение модельных задач и перспективы практической реализации. Обсуждаются потенциальные области применения и планы апробации результатов.

ЗАКЛЮЧЕНИЕ

СПИСОК ЛИТЕРАТУРЫ

1. *Bertsekas, Dimitri P.* The Auction Algorithm for Assignment and Other Network Flow Problems: A Tutorial. — 1990. — URL: https://web.mit.edu/dimitrib/www/Auction_Interfaces_Published.pdf (visited on 07.06.2022).
2. *Kuhn, Harold W.* The Hungarian Method for the Assignment Problem. — 1955. — URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109 (visited on 07.06.2022).
3. Групповое управление подвижными объектами в неопределенных средах / Пшихопов, В. Х. and Соловьев, В. В. and Титов, А. Е. and Финаев, В. И. and Шаповалов, И. О. — 2015. — (Дата обращения: 27.05.2025); (Group-Control-of–Mobile-Objects-in-Uncertain-Environments.pdf).
4. *Каляев, И. А. and Гайдук, А. Р. and Капустян, С. Г.* Модели и алгоритмы коллективного управления в группах роботов. — 2009. — URL: https://www.litres.ru/get_pdf_trial/18402114.pdf (дата обращения: 27.05.2025).
5. *Gerkey, Brian P.* On Multi-Robot Task Allocation. — 2003. — URL: https://ai.stanford.edu/~gerkey/research/final_papers/diss.pdf (visited on 27.05.2025).
6. *Bertsekas, Dimitri P. and Tsitsiklis, John N.* Parallel and Distributed Computation: Numerical Methods. — 1989. — URL: https://archive.org/details/paralleldistribu0000bert (visited on 07.06.2022).
7. *e-maxx.ru*. Венгерский алгоритм. — 2025. — URL: http://www.e-maxx-ru.1gb.ru/algo/assignment_hungary (дата обращения: 28.05.2025).