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 Отчтетство, 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 или вообще неприменимы. 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 целей между роботами с учетом ограниченных областей связи.

Цель исследования — повышение эффективности распределения целей в распределенных системах роботов с ограниченными областями связи путем разработки и применения модифицированного аукционного алгоритма.

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 целей в мультироботных 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 m объектам (например, задачам или целям) для максимизации суммарной выгоды, заданной матрицей $\{a_{ij}\}$, где a_{ij} — выгода от назначения агента i объекту j. Математически цель задачи в общем случае:

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

где j_i — объект, назначенный агенту i, при условии, что каждый агент получает не более одного объекта, и каждый объект назначается не более чем одному агенту. Если $n \neq m$, некоторые агенты или объекты могут остаться неназначенными.

В робототехнике задача назначения актуальна для распределения целей 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, а также оценить их применимость в 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 роботы делают ставки на цели, а цены корректируются для достижения оптимального распределения.

1.2.1. Постановка задачи

Задача назначения заключается в нахождении оптимального соответствия между n роботами и m целями с учетом матрицы выгод $\{a_{ij}\}$, где a_{ij} — выгода робота i при назначении ему цели j. Цель — максимизировать суммарную выгоду:

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

где j_i — цель, назначенная роботу i, при условии, что каждый робот получает не более одной цели, и каждая цель назначается не более чем одному роботу. Эта задача эквивалентна задаче линейного программирования:

$$\max \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} x_{ij},$$

при ограничениях:

$$\sum_{i=1}^{n} x_{ij} \leqslant 1, \quad \forall j = 1, \ldots, m,$$

$$\sum_{j=1}^{m} x_{ij} \leqslant 1, \quad \forall i = 1, \ldots, n,$$

$$x_{ij} \in \{0, 1\}, \quad \forall i, j,$$

где $x_{ij} = 1$, если робот i назначен цели j, и $x_{ij} = 0$ в противном случае. Аукционный алгоритм решает эту задачу приближенно, обеспечивая суммарную выгоду, которая, согласно [1], находится в пределах $\min(n, m)\varepsilon$ от оптимального значения, где $\varepsilon > 0$ — параметр точности, определяющий степень приближения. Это означает, что если A^* — оптимальная выгода, а A — выгода, полученная алгоритмом, то:

$$A^* - \min(n, m)\varepsilon \leqslant A \leqslant A^*.$$

Для целочисленных a_{ij} и $\varepsilon < 1/\min(n, m)$, алгоритм гарантирует точное оптимальное решение.

1.2.2. Описание алгоритма

Согласно [1], аукционный алгоритм работает с матрицей выгод $\{a_{ij}\}$, где a_{ij} — выгода робота i при назначении цели j, и использует цены $\{p_j\}$, где $p_j \geqslant 0$ — цена, связанная с целью j, отражающая текущую стоимость ее назначения и регулирующая конкуренцию между роботами. Алгоритм начинается с произвольного распределения роботов по целям (возможно, пустого) и начальных цен $\{p_j\}$, обычно равных нулю. Ключевые определения:

– *Почти счастье*: Робот i, назначенный цели j_i, почти счастлив, если:

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

где $\varepsilon > 0$ — параметр точности.

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

Шаги алгоритма [1]:

A. Проверить, все ли назначенные роботы почти счастливы. Если да, завершить.

B. Выбрать робота i, не почти счастливого (или неназначенного), и найти цель j_i:

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

C. Переназначить: робот i получает цель j_i, а робот, ранее назначенный на j_i, получает цель, принадлежавшую i, или остается неназначенным.

D. Увеличить цену p_{j_i} на:

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

где $v_i = \max_{j=1,\ldots,m}\{a_{ij} - p_j\}$, $w_i = \max_{j \neq j_i}\{a_{ij} - p_j\}$.

E. Повторить с шага 1.

1.2.3. Сходимость алгоритма

Теорема 1.1 (Сходимость аукционного алгоритма [1]). *Для произвольных n роботов и m целей аукционный алгоритм завершается за конечное число шагов с распределением и ценами, почти в равновесии, при котором каждый* *назначенный робот получает уникальную цель и является почти счастливым, а неназначенные роботы возможны только при $n > m$.*

Доказательство. Аукционный алгоритм итеративно назначает n роботов m целям, корректируя цены $\{p_j\}$. Робот i, назначенный цели j_i, почти счастлив, если:

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

где $\varepsilon > 0$ — параметр точности. На каждой итерации алгоритм выбирает робота i, который либо неназначен, либо не почти счастлив, и определяет цель j_i, максимизирующую:

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

увеличивая цену p_{j_i} на:

$$\gamma_i = v_i - w_i + \varepsilon, \quad \text{где} \quad v_i = \max_{j=1,\dots,m} \{a_{ij} - p_j\}, \quad w_i = \max_{j \neq j_i} \{a_{ij} - p_j\}.$$

Робот i назначается цели j_i, становясь почти счастливым, так как:

$$a_{ij_i} - (p_{j_i} + \gamma_i) = a_{ij_i} - p_{j_i} - (v_i - w_i + \varepsilon) = w_i - \varepsilon \leqslant \max_{j \neq j_i} \{a_{ij} - p_j\} - \varepsilon.$$

Если цель j_i уже была назначена роботу k, то k становится неназначенным, а i получает j_i. Это обеспечивает уникальность назначений: каждая цель назначается не более чем одному роботу.

Теперь объясним, почему каждый назначенный робот получает уникальную цель, и почему неназначенные роботы возможны только при $n > m$. Алгоритм продолжает итерации, пока существуют неназначенные или не почти счастливые роботы. На каждой итерации неназначенный робот i выбирает цель j_i, увеличивая ее цену на $\gamma_i \geqslant \varepsilon$. Если j_i занята роботом k, то k становится неназначенным и в следующей итерации выбирает новую цель. Цены p_j монотонно возрастают, что делает уже назначенные цели менее привлекательными для неназначенных роботов, так как $a_{ij} - p_j$ уменьшается с ростом p_j.

Если цель j получила много ставок, ее цена $p_j \geqslant m'\varepsilon$, где m' — число ставок. При большом p_j, $a_{ij} - p_j$ становится меньше, чем $a_{ik} - p_k$ для цели k

с низкой ценой (например, $p_k = 0$, если k не получала ставок). Таким образом, неназначенные роботы предпочитают цели с низкими ценами, которые часто свободны или менее востребованы. Поскольку рост цен ограничен (максимум $C + \varepsilon$, где $C = \max_{i,j} |a_{ij}|$, так как $a_{ij} - p_j < 0$ невыгодно), число ставок конечно.

При $n \leqslant m$, алгоритм стремится назначить каждому из n роботов уникальную цель, так как целей достаточно. Итерации продолжаются, пока все роботы не станут почти счастливыми, что возможно, так как число целей $m \geqslant n$. Когда все n роботов назначены и почти счастливы, алгоритм завершается. При $n > m$, максимум m роботов могут быть назначены, так как целей только m. В этом случае после назначения m целей неназначенные $n - m$ роботы остаются без целей, и алгоритм завершается, так как все назначенные роботы почти счастливы, а неназначенным роботам не хватает целей для новых назначений. Уникальность назначений сохраняется на всех итерациях, так как переназначение освобождает цель ровно для одного робота.

Число итераций конечно из-за конечности n, m и ограниченности роста цен. Таким образом, алгоритм завершается за конечное число шагов, обеспечивая, что каждый назначенный робот получает уникальную цель и является почти счастливым, а неназначенные роботы возможны только при $n > m$.

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 цели 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 $\min(n, m)\varepsilon$ от оптимальной. При целочисленных a_{ij} и $\varepsilon < 1/\min(n, m)$, распределение оптимально.*

Доказательство. Оптимальная выгода (примитивная задача):

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

Двойственная задача:

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

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

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

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 для назначенных роботов (не более $\min(n, m)$):

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

Добавим $\sum_{j=1}^{m} p_j$:

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

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

$$\sum_{i=1}^{\min(n,m)} a_{ij_i} \geqslant D^* - \min(n, m)\varepsilon \geqslant A^* - \min(n, m)\varepsilon.$$

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

$$A^* - \min(n, m)\varepsilon \leqslant \sum_{i=1}^{\min(n,m)} a_{ij_i} \leqslant A^*.$$

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

1.2.6. Преимущества и недостатки

Преимущества:

- Интуитивная экономическая модель: алгоритм имитирует реальный аукцион, упрощая понимание и интерпретацию.
- Гибкость для адаптации: легко модифицируется для асимметричных задач с $n \neq m$, 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 итераций, примерно пропорционально C/ε.
- Чувствительность к выбору ε: большое ε снижает точность, 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. Матричный подход

Алгоритм работает с матрицей выгод $\{a_{ij}\}$ размера $n \times m$. Для унификации, если $n \neq m$, матрица дополняется фиктивными роботами или целями с нулевой выгодой, чтобы получить квадратную матрицу размера $\max(n, m) \times \max(n, m)$ [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 $\min(n, m)$, 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 — роботы (n вершин), V_2 — цели (m вершин), 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 $\min(n, m)$ вершин, оно оптимально. Иначе корректировать матрицу (δ) и обновить граф.

1.3.3. Доказательство оптимальности

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

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

Двойственная задача:

$$\min \sum_{i=1}^{n} u_i + \sum_{j=1}^{m} v_j, \quad u_i + v_j \geqslant a_{ij},$$

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. Преимущества и недостатки

Преимущества:

– Гарантированная оптимальность.

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

Недостатки:

- Сложность $O(n^2 * m)$ для больших n или m.
- Требует полной матрицы $\{a_{ij}\}$, что проблематично при ограниченной коммуникации [5].
- Непригодность для распределенных систем: невозможность работы в условиях ограниченной коммуникации между роботами, так как алгоритм предполагает централизованное управление [4].
- Нет параллелизма [6].

1.4. Постановка проблемы

В распределенных системах с n роботами и m целями часто возникают ограниченные области связи, при которых каждый робот имеет доступ к полной информации о матрице выгод $\{a_{ij}\}$, то есть видит все цели, но не может обмениваться информацией с другими роботами [4; 5]. Аукционный алгоритм [1] предполагает, что роботы могут координировать назначения через обмен информацией, что невозможно в условиях отсутствия коммуникации между роботами. Это приводит к необходимости дополнительных механизмов координации или снижению качества назначений. Венгерский алгоритм [2; 7] требует полной матрицы выгод и централизованного управления, что неосуществимо в таких системах [4]. Необходим модифицированный аукционный алгоритм, который:

- Учитывает отсутствие коммуникации между n роботами при доступе к полной информации о m целями.
- Обеспечивает эффективность и близость к оптимальности.

Цель — разработать такой алгоритм и сравнить его с венгерским методом.

1.5. Выводы

Рассмотрены аукционный [1] и венгерский [2; 7] алгоритмы для задачи назначения n роботов по m целям. Аукционный алгоритм эффективен, но требует адаптации для условий, где роботы не могут обмениваться информацией друг с другом, несмотря на доступ к полной матрице выгод. Венгерский алгоритм оптимален, но непригоден для распределенных систем без централизации. Необходим модифицированный аукционный алгоритм, учитывающий отсутствие коммуникации между роботами.

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 целям, что упрощает задачу назначения, но требует координации внутри групп. Описывается математическая модель задачи с учетом локальной доступности данных, а также алгоритмические изменения, обеспечивающие независимую обработку связных компонент и устойчивость к отсутствию коммуникации между группами. 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 целей $T = \{t_1, t_2, \ldots, t_m\}$. Каждый робот r_i имеет координаты $p_i = (x_i, y_i)$, а каждая цель t_j — координаты $q_j = (x_j, y_j)$. Выгода от назначения робота r_i цели 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 целям, 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 — цель, назначенная роботу r_i, при условиях:

- Цель может быть назначена нескольким роботам в процессе аукциона, но в итоговом решении для каждой цели t_j учитывается только робот с максимальной выгодой α_{ij}, определяемой как наибольшая эффективность (например, минимальное время достижения цели). Остальные назначения для этой цели 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 отсутствия коммуникации между роботами, за исключением обмена внутри связных компонент. Ключевые изменения включают:

- Полная видимость целей: Все цели 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.
- Независимая обработка компонент: Каждая связная компонента обрабатывается отдельно, моделируя распределенную систему без коммуникации между группами.
- Разрешение конфликтов: Если несколько роботов из разных компонент выбирают одну цель, учитывается только робот с максимальной выгодой α_{ij}, а остальные назначения отбрасываются.

2.2.1. Описание алгоритма

Алгоритм состоит из следующих шагов:

А. **Инициализация**:

- Задать матрицу выгод $\{\alpha_{ij}\}$, координаты роботов $\{p_i\}$ и целей $\{q_j\}$, 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 целей $\{p_j = 0\}$ 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. **Аукцион в компонентах**:

- 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 цель t_{j_i}: $j_i = \arg\max_j\{\alpha_{ij} - p_j\}$, включая фиктивную цель ($\alpha_{i,-1} = 0$).
  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 цель без изменения цен.
  6. Иначе переназначить: робот r_i получает цель 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 цели 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 вре-

мя достижения цели). Остальные роботы, назначенные на t_j, переназначаются на фиктивную цель ($j_i = -1$).

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 проводит аукцион в каждой связной компоненте C_l, следуя шагам оригинального аукционного алгоритма, описанного в главе 1 [1]. Согласно теореме 1.1 (глава 1), оригинальный аукционный алгоритм завершается за конечное число шагов, обеспечивая почти равновесие, при котором все назначенные роботы почти счастливы, то есть для каждого робота i, назначенного цели j_i, выполняется:

$$\alpha_{ij_i} - p_{j_i} \geqslant \max_{j=1,\dots,m} \{\alpha_{ij} - p_j\} - \varepsilon.$$

Поскольку аукцион в каждой компоненте C_l идентичен оригинальному, сходимость внутри C_l гарантируется той же теоремой. Разрешение конфликтов после аукциона не влияет на сходимость, так как оно происходит постфактум и лишь переназначает роботов на фиктивную цель ($j_i = -1$) без 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. В каждой связной компоненте C_l модифицированный алгоритм выполняет аукцион, идентичный оригинальному, описанному в главе 1 [1]. Согласно утверждению 1.1 (глава 1), число итераций оригинального аукционного алгоритма пропорционально C/ε, где $C = \max_{i,j} |\alpha_{ij}|$. Поскольку аукцион в каждой компоненте C_l следует той же логике, число итераций в C_l также составляет $O(C/\varepsilon)$. Общее число итераций определяется максимальным числом итераций среди всех компонент, 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 $\varepsilon < 1/n$ дает оптимальное решение локальной задачи аукциона, как определено в главе 1. Однако общее решение после объединения компонент не гарантирует глобальной оптимальности из-за отсутствия коммуникации между компонентами.*

Доказательство. В каждой связной компоненте C_l аукцион проводится по правилам оригинального аукционного алгоритма, описанного в главе 1 [1]. Согласно теореме 1.2 (глава 1), при целочисленных α_{ij} и $\varepsilon < 1/n$ оригинальный аукционный алгоритм обеспечивает оптимальное решение, где суммарная выгода $\sum_{i \in C_l} \alpha_{ij_i}$ равна локальному оптимуму. Таким образом, в каждой компоненте C_l достигается оптимальное локальное назначение.

Однако глобальная оптимальность не гарантируется. После аукциона в компонентах разрешение конфликтов выбирает для каждой цели t_j робота с максимальной выгодой α_{ij} среди назначенных, переназначая остальных на фиктивную цель. Из-за отсутствия коммуникации между компонентами роботы в C_l не учитывают выгоды роботов из других компонент, что может привести к выбору локально оптимальных, но глобально неоптимальных назначений. Следовательно, суммарная выгода $\sum_{i=1}^{n} \alpha_{ij_i}$ может быть меньше глобального оптимума.

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 коммуникационных затрат за счет независимой 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 отсутствие коммуникации между роботами в распределенных системах, за исключением обмена внутри связных компонент. Алгоритм группирует роботов в связные компоненты с помощью матрицы видимости V, проводит аукцион независимо в каждой компоненте и обеспечивает полную видимость целей. 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 отсутствия коммуникации между компонентами. 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++ с использованием библиотек Qt для визуализации данных. 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. Тестирование проводится на случайных данных с варьированием размеров задачи (от 5 до 300 роботов/целей), радиусов связи (от 1 до 300 условных единиц) и параметра ε алгоритма (от 10^{-6} до 10.0).

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 данной главе представлены результаты экспериментального исследования модифицированного аукционного алгоритма, описанного в главе 2, в сравнении с классическим венгерским алгоритмом. Рассматриваются модельные задачи, имитирующие распределенные системы роботов с различной топологией сети связи, определяемой радиусом видимости r_v. Описываются метрики оценки, включающие время выполнения, количество операций и относительную точность. Анализируется эффективность предложенного алгоритма, проверяются гипотезы, сформулированные в главе 2. Результаты визуализированы в виде графиков.

4.1. Постановка эксперимента

4.1.1. Модельные задачи

Для исследования эффективности модифицированного аукционного алгоритма были разработаны модельные задачи, имитирующие распределенные системы роботов. Параметры задач включали:

- **Размер задачи**: Число роботов n и задач m варьировалось от 5 до 100, при этом $n = m$ для сбалансированной задачи.
- **Радиус видимости**: Рассматривались значения $r_v \in \{10, 20, 30\}$, а также диапазон от 5 до 100 для анализа влияния радиуса на точность.
- **Координаты**: Координаты роботов $\{p_i = (x_i, y_i)\}$ и задач $\{q_j = (x_j, y_j)\}$ генерировались случайным образом в диапазоне $[1, 100]$ с использованием генератора псевдослучайных чисел.
- **Матрица выгод**: Выгода α_{ij} определялась как:

$$\alpha_{ij} = \frac{100}{d(p_i, q_j) + \delta},$$

  где $d(p_i, q_j)$ — евклидово расстояние, δ — константа для предотвращения деления на ноль.
- **Матрица видимости**: Матрица $V = \{v_{ij}\}$ формировалась так, что $v_{ij} = 1$, если $d(p_i, p_j) \leqslant r_v$, и $v_{ij} = 0$ в противном случае.
- **Параметр** ε: Для аукционного алгоритма исследовались значения ε от 10^{-5} до 10 с логарифмическим шагом.

Каждая задача генерировалась 100 раз для усреднения результатов, что обеспечивало статистическую надежность.

4.1.2. Метрики оценки

Для сравнения алгоритмов использовались следующие метрики:

- **Время выполнения**: Измерялось в миллисекундах и усреднялось по 100 запускам для каждого набора параметров (n, r_v, ε).
- **Количество операций**: Определялось как число итераций алгоритмов для фиксированного радиуса $r_v = 200$.

– **Относительная точность**: Вычислялась как:

$$\text{Точность} = \frac{A_{\text{auction}}}{A_{\text{hungarian}}} \cdot 100\%,$$

где A_{auction} — суммарная выгода аукционного алгоритма, $A_{\text{hungarian}}$ — суммарная выгода венгерского алгоритма (оптимальное решение).

4.1.3. Методика проведения экспериментов

Эксперименты проводились с использованием программной реализации на языке C++ с применением стандартных библиотек для обработки данных и построения графиков. Исследовались следующие аспекты:

A. Сравнение времени выполнения аукционного и венгерского алгоритмов для различных размеров задачи и радиусов видимости.
B. Оценка числа итераций для фиксированного радиуса $r_v = 200$.
C. Анализ относительной точности аукционного алгоритма для фиксированных размеров задачи $n \in \{10, 30, 50\}$ и различных радиусов.
D. Исследование влияния параметра ε на время выполнения и точность при $n = 80$ и $r_v \in \{10, 20, 30\}$.
E. Оценка точности аукционного алгоритма на каждой итерации для $n \in \{10, 30, 50, 100\}$ и $r_v = 200$.

Результаты представлены в виде графиков, сохраненных в формате PNG.

4.2. Результаты экспериментов

4.2.1. Время выполнения

Эксперименты проводились для $n = m$ от 5 до 100 и радиусов $r_v \in \{10, 20, 30\}$. Результаты представлены на графике (см. рисунок 4.1).

Основные наблюдения:

– Время выполнения аукционного алгоритма возрастает с увеличением n, но остается конкурентоспособным по сравнению с венгерским алгоритмом, особенно при больших r_v.
– Увеличение радиуса видимости снижает время выполнения аукционного алгоритма за счет формирования более крупных связных компонент.

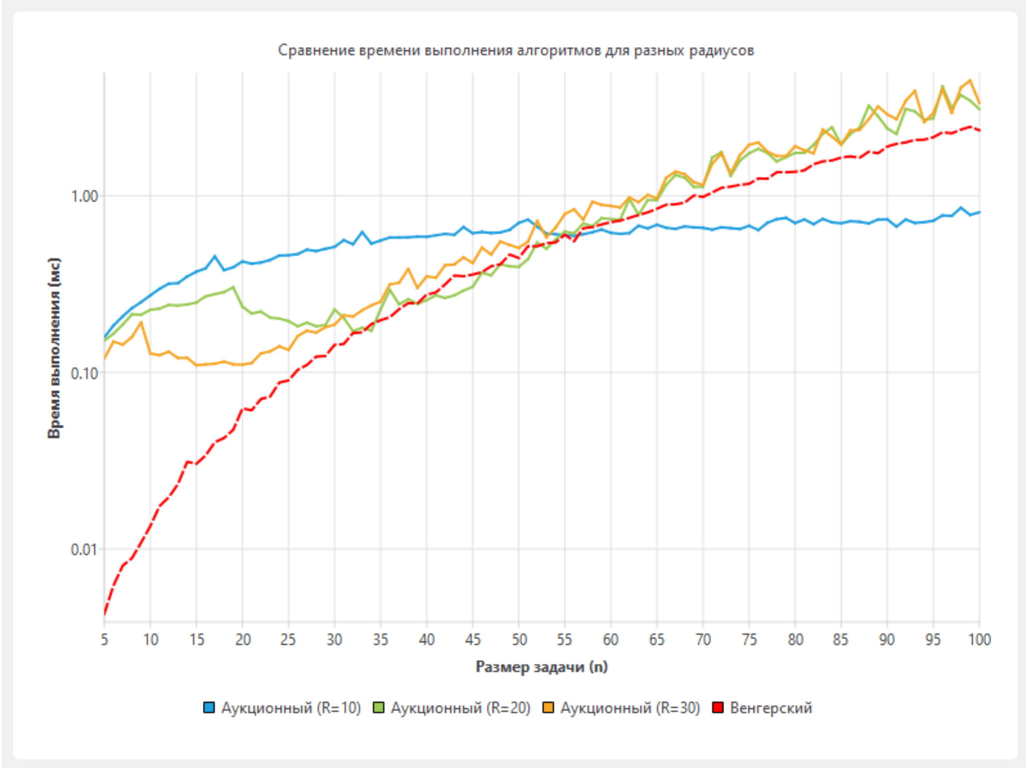

Рис.4.1. Зависимость времени выполнения аукционного и венгерского алгоритмов от размера задачи для различных радиусов видимости.

- Венгерский алгоритм демонстрирует более стабильное время выполнения, но его производительность ухудшается при больших n из-за сложности $O(n^3)$.

4.2.2. Относительная точность

Точность аукционного алгоритма исследовалась для $n \in \{10, 30, 50\}$ и r_v от 5 до 100 (см. рисунок 4.2).

Выводы:

- Точность возрастает с увеличением r_v, достигая 95–100% при $r_v \geqslant 50$ для малых n.
- Для больших n точность возрастает быстрее из-за увеличения числа связных компонент.

4.2.3. Влияние параметра ε

Влияние ε на время и точность исследовалось для $n = 80$ и $r_v \in \{10, 20, 30\}$ (см. рисунки 4.3 и 4.4).

Наблюдения:

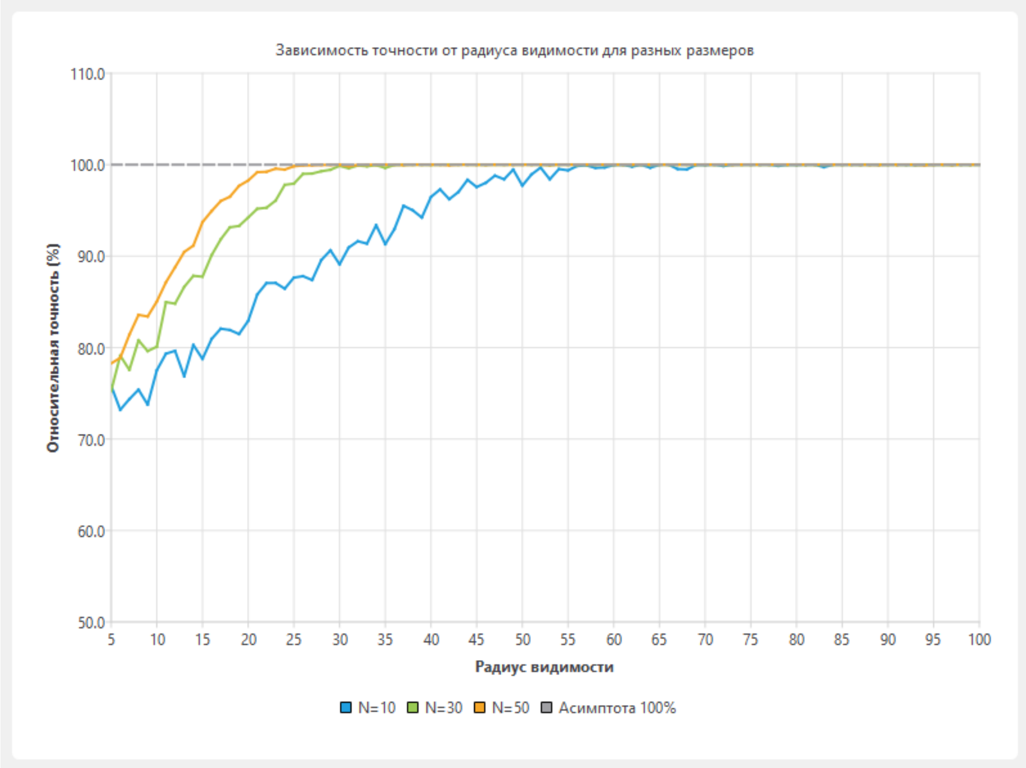

Рис.4.2. Зависимость относительной точности аукционного алгоритма от радиуса видимости для различных размеров задачи.

- Время выполнения уменьшается с ростом ε, так как требуется меньше итераций.
- Точность падает при больших ε, но при $\varepsilon \leqslant 10^{-2}$ превышает 90%.

4.2.4. Точность по итерациям

Точность на каждой итерации исследовалась для $n \in \{10, 30, 50, 100\}$ и $r_v = 200$ (см. рисунок 4.5).

Наблюдения:

- Точность быстро растет на первых итерациях, стабилизируясь на уровне 90–100%.
- Для больших n требуется больше итераций из-за увеличения числа конфликтов.

4.3. Анализ результатов

Результаты подтверждают гипотезы, сформулированные в главе 2:

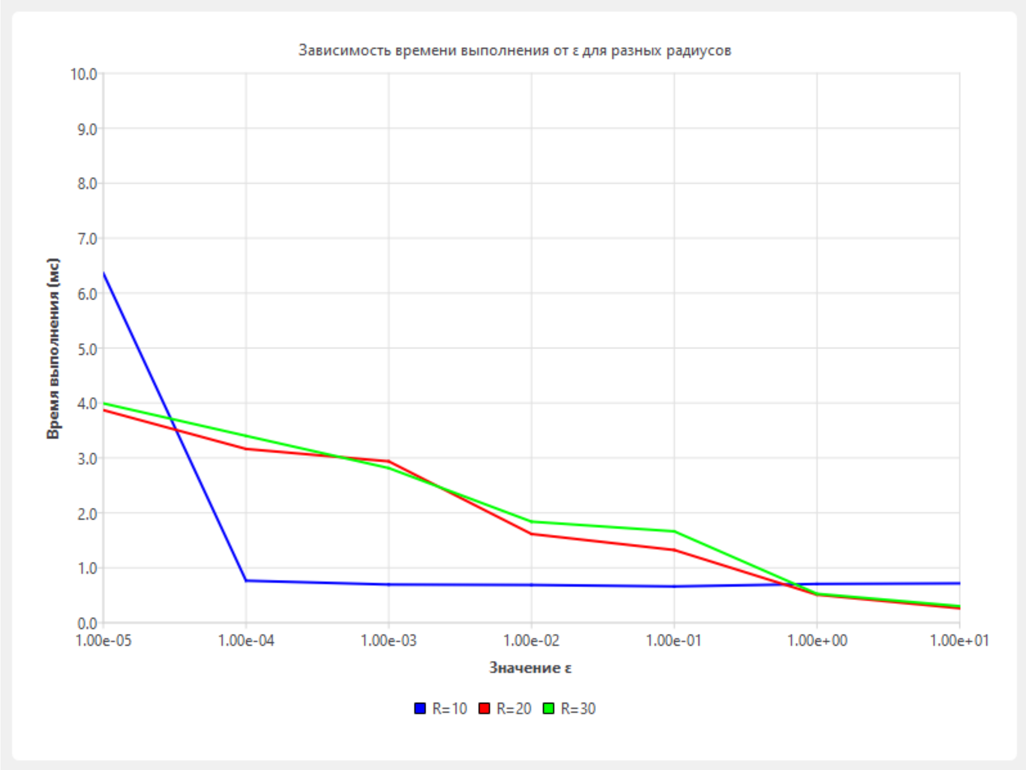

Рис.4.3. Зависимость времени выполнения аукционного алгоритма от ε для различных радиусов видимости.

- **Гипотеза 1**: Алгоритм завершается за конечное число шагов, а время выполнения пропорционально C/ε, что подтверждается зависимостью времени от ε и r_v.
- **Гипотеза 2**: Локальная оптимальность достигается в каждой связной компоненте при $\varepsilon < 1/n$, но глобальная оптимальность ограничена числом компонент.
- **Гипотеза 3**: Высокая точность при небольших r_v указывает на потенциальную устойчивость в динамических сетях в условиях ограниченной связи.

Модифицированный аукционный алгоритм демонстрирует конкурентоспособное время выполнения и высокую точность (95–100%) при подходящих параметрах.

4.4. Выводы

Модифицированный аукционный алгоритм показал эффективность в распределенных системах роботов, обеспечивая высокую точность и конкурентоспособное время выполнения. Параллельная обработка связных компонент

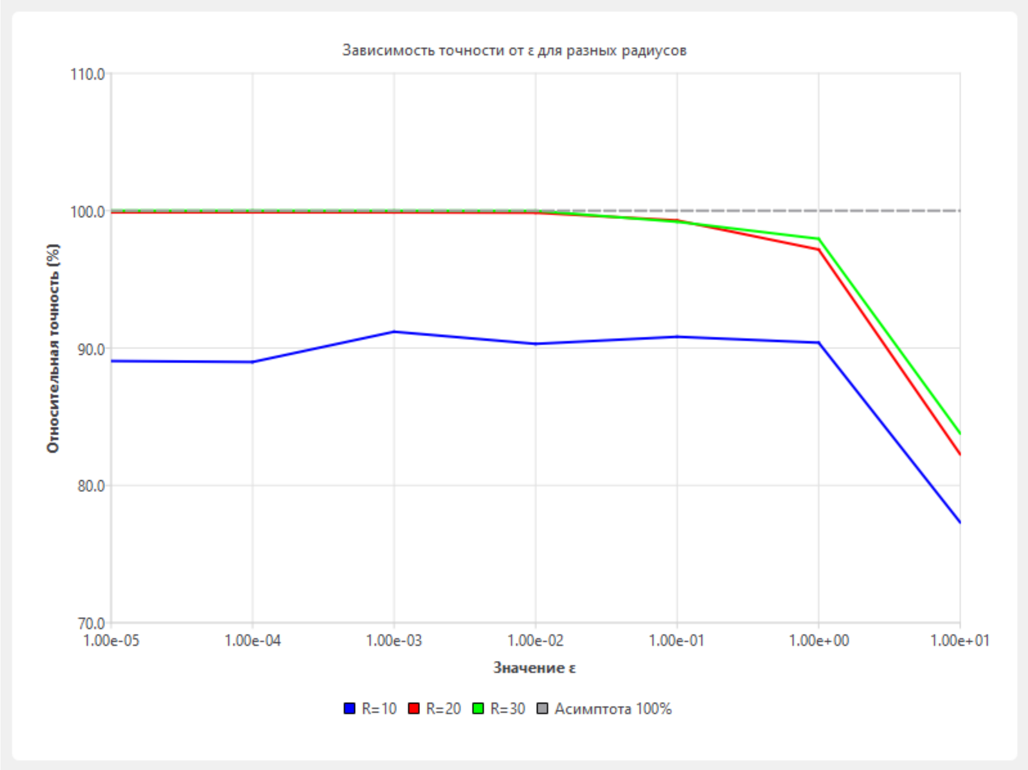

Рис.4.4. Зависимость относительной точности аукционного алгоритма от ε для различных радиусов видимости.

снижает вычислительные затраты. Дальнейшие исследования могут быть направлены на тестирование алгоритма в динамических сетях и улучшение глобальной оптимальности.

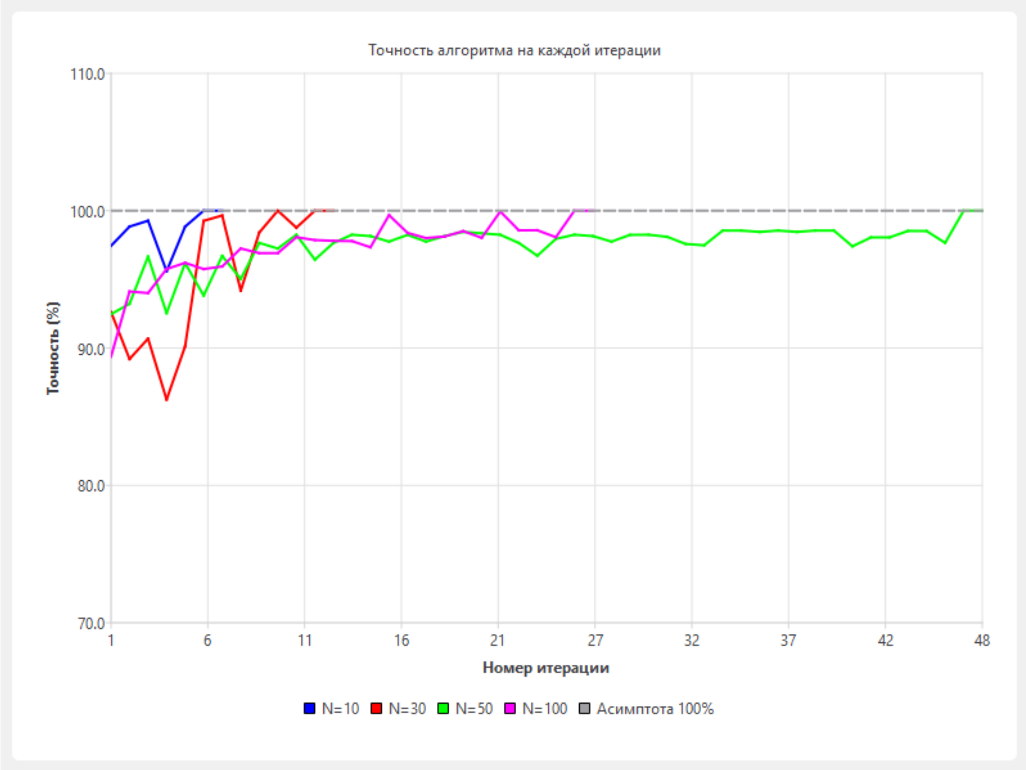

Рис.4.5. Зависимость относительной точности аукционного алгоритма от номера итерации для различных размеров задачи.

ГЛАВА 5. ПЕРСПЕКТИВЫ ДАЛЬНЕЙШЕГО РАЗВИТИЯ И ИССЛЕДОВАНИЯ

Глава посвящена направлениям дальнейшего развития модифицированного аукционного алгоритма и его применению в распределенных системах роботов. Рассматриваются возможные улучшения алгоритма, расширение модельных задач, новые метрики оценки 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).