

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

УДК 628.52.012.011.56

АЛКАРОЛИ  
Мухиеддин Камел Салман

**ДИНАМИЧЕСКОЕ ПЕРЕРАСПРЕДЕЛЕНИЕ ЗАДАЧ  
В СИСТЕМАХ ВЗАИМОДЕЙСТВУЮЩИХ АГЕНТОВ**

Автореферат  
диссертации на соискание ученой степени  
кандидата технических наук  
по специальности 05.13.01 – Системный анализ, управление  
и обработка информации

Минск 2016

Работа выполнена в учреждении образования «Белорусский государственный университет информатики и радиоэлектроники».

Научный руководитель: **Ревотюк Михаил Павлович**,  
кандидат технических наук, доцент,  
доцент кафедры информационных технологий  
автоматизированных систем учреждения  
образования «Белорусский государственный  
университет информатики и радиоэлектроники».

Официальные оппоненты: **Вишняков Владимир Анатольевич**,  
доктор технических наук, профессор,  
профессор кафедры информационных  
технологий учреждения образования  
«Минский инновационный университет»;  
**Бракович Андрей Игоревич**,  
кандидат технических наук, доцент,  
доцент кафедры информационных систем и  
технологий учреждения образования  
«Белорусский государственный  
технологический университет».

Оппонирующая организация: Белорусский национальный технический университет.

Защита состоится 22 декабря 2016 г. в 14 часов на заседании совета по защите диссертаций Д 02.15.01 при учреждении образования «Белорусский государственный университет информатики и радиоэлектроники» по адресу: 220013, г. Минск, ул. П. Бровки, 6, уч. корпус 1, ауд. 232; тел. +375 17 239-89-89; e-mail: dissovet@bsuir.by.

С диссертацией можно ознакомиться в библиотеке учреждения образования «Белорусский государственный университет информатики и радиоэлектроники».

Автореферат разослан 21 ноября 2016 года.

Ученый секретарь  
совета по защите диссертаций  
доктор технических наук, профессор

М.М. Татур

## КРАТКОЕ ВВЕДЕНИЕ

Существующий опыт применения общих методов оптимального управления распределенными системами, например, на транспорте или робототехнике, показывает, что экспоненциальная сложность алгоритмов решения задач управления становится причиной отказа от централизованных схем глобальной оптимизации. Возникает потребность выбора и обоснования метода приближенного решения задач управления, используя структурно-функциональную декомпозицию или различные приемы аппроксимации. Агентный подход может определять концептуальную модель системы, аккумулируя удачные приемы и методы представления пространства ее состояния, а также процедуры поиска решений. Наиболее адекватными для моделирования взаимодействия агентов являются имитационные модели на основе линейной временной логики, однако вопросы поиска законов управления оказываются вне модели. Возможность учета восприимчивости распределенных систем к изменению локальных переменных состояния позволяет организовать итерационный процесс уточнения предварительно построенных решений, используя общие процедуры логического вывода или исчерпывающего поиска, а также возможности их распараллеливания.

Активное распространение автономных агентных систем, а также реальные возможности глобальных систем связи по получению актуальной информации о состоянии объектов управления, порождают вопросы выбора методов оптимизации, представления структуры модели, обеспечения корректной реакции системы на возмущения. Решение таких вопросов должно проводиться с учетом сложившейся инфраструктуры и стандартов построения открытых систем, используемых для реализации систем агентов. В свою очередь, задачи согласования протоколов взаимодействия агентов, представления и слияния планов их деятельности, могут рассматриваться в терминах распределения задач между агентами соответствующего уровня абстрагирования.

Таким образом, тема исследования методов перераспределения задач, позволяющих учесть наиболее актуальную информацию о состоянии и организовать инкрементальные процедуры поиска управления на основе оценки возмущения состояния, является актуальной.

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

*Связь работы с крупными научными программами, темами.* Исследования проводились в рамках научно-исследовательской темы № 11-2016 «Методы, модели и технологии современных информационно-аналитических и управляющих систем» (номер госрегистрации 20112038). Тема соответствует пункту 5.1 «Методы математического и компьютерного моделирования, компьютерные технологии и интеллектуальные системы поддержки принятия решений» Перечня приоритетных направлений научных исследований Республики Беларусь на 2011-2015 годы, утвержденного Постановлением Совета Министров Республики Беларусь № 585 от 19 апреля 2010 г.

**Цель и задачи исследования.** Целью работы является разработка моделей и эффективных по быстродействию алгоритмов оперативного распределения потоков задач в системах взаимодействующих агентов с централизованным и коллективным управлением.

Для достижения указанной цели необходимо решить следующие задачи:

- 1) разработать модели и алгоритмы спецификации процессов управления взаимодействующими процессами, основанные на рекуррентных сетях;
- 2) исследовать методы оценки устойчивости решения задач об оптимальном назначении координируемых агентов;
- 3) разработать методы самоорганизации процедур оптимизации управления на сетевых моделях с накоплением информации;
- 4) рассмотреть вопросы эффективной реализации алгоритмов управления взаимодействующими агентами на основе распределенных вычислений.

**Научная новизна и значимость полученных результатов.**

1. Предложена новая модель и алгоритм поиска оптимального паросочетания агентов и решаемых ими задач на основе метода кратчайшего пополняющего пути, отличающихся от известных моделей использованием компактных рекуррентных схем реоптимизации решений, что обеспечивает линейную зависимость времени реакции системы на внешние события.

2. Разработаны алгоритмы оценки устойчивости решения линейных задач о назначении и коммивояжера, использующие принцип реоптимизации решения после инверсии состояния дуг графа оптимального паросочетания, что позволило снизить вычислительную сложность таких алгоритмов.

3. Предложены модели реализации целеустремленного поведения агента на основе прямого и обратного поиска кратчайших путей на графах, учитывающие ограничения на структуры путей и predetermined решения, позволяют снизить степень вычислительной сложности до диаметра графа, что существенно для реализации быстродействующих систем.

4. Предложен алгоритм реоптимизации решения транспортных подзадач методом потенциалов с контролем бесперспективности варианта в комбинаторных задачах оптимизации размещения по схеме кооперации ресурсов вычислительной среды, позволяющие наряду с учетом неоднородных ограничений улучшить динамические характеристики процесса поиска решений в реальном времени.

5. Предложены и обоснованы схемы алгоритмов грануляции задач комбинаторного вида для репликации и миграции агентов распределенных вычислений, что позволяет обеспечить в реальном времени гарантии целостности и безопасности реализации процедур метода динамического программирования, а также метода ветвей и границ.

**Основные положения диссертации, выносимые на защиту.**

1. Уточненные модели координации систем взаимодействующих агентов и алгоритмы реоптимизации решений линейных задач о назначении методом кратчайшего пополняющего пути.

2. Алгоритмы оценки устойчивости решения задач о назначении и задач коммивояжера для фильтрации возмущений параметров целевой функции и ограничений.

3. Модели и быстродействующие алгоритмы поиска кратчайших путей на графах с ограничениями и автоматическим формированием и накоплением предопределенных решений.

**Личный вклад соискателя.** В настоящую диссертационную работу вошли результаты как личных исследований автора (развитие аппарата описания взаимодействующих процессов на основе рекуррентных сетей, экспериментальные исследования эффективности решения задач оптимального управления), так и его совместной деятельности с научным руководителем к.т.н., доцентом Ревотюком М.П. (концептуальные схемы моделей процесса императивного управления и организации распределенных вычислений, алгоритмы реоптимизации решений задач о назначении), а также соавторами научных трудов (макетирование процедур моделирования и оптимизации, экспериментальная проверка их эффективности).

**Апробация диссертации и информация об использовании ее результатов.** Основные результаты работы докладывались на VII Междунар. научно-техн. конф. «Информационные технологии в промышленности» ITI'2012 (Минск, 2012); 48-й, 49-й и 50-й научных конф. аспирантов, магистрантов и студентов «Информационные технологии и управление» (Минск, 2012; Минск, 2013; Минск, 2014); X, XI и XII Белорусско-российской научно-техн. конф. «Технические средства защиты информации» (Минск, 2012; Минск, 2013; Минск, 2014; Минск, 2016); Междунар. конф. «Информационные технологии и системы» ITS'2012 (Минск, 2012), ITS'2013 (Минск, 2013), ITS'2014 (Минск, 2014), ITS'2015 (Минск, 2015); VI Междунар. научно-метод. конф. «Высшее техническое образование: проблемы и пути развития» (Минск, 2012); Междунар. научно-техн. конф. «Информационные системы и технологии» IST'2013 (Н.Новгород, 2013); VIII Респ. научной конф. молодых ученых и студентов «Современные проблемы математики и вычислительной техники» (Брест, 2013); VIII междунар. науч.-метод. конф. «Дистанционное обучение – образовательная среда XXI века» (Минск, 2013); Междунар. научно-техн. конф., приуроченной к 50-летию МРТИ–БГУИР (Минск, 2014).

Результаты диссертационной работы использованы при проектировании бизнес-процессов компании Transport & Construction Co (Доха, Катар).

**Опубликованность результатов.** По теме диссертационной работы опубликовано 25 печатных работ (общий объем – около 64 стр.), в их числе 3 статьи в научно-теоретическом журнале, 22 тезиса и текста докладов в сборниках трудов и материалов конференций.

**Структура и объем диссертации.** Диссертация состоит из введения, общей характеристики, четырех глав, заключения, библиографического списка и приложений. Работа изложена на 139 страницах машинописного текста, в том числе основная часть – на 116 страницах, и содержит:

43 рисунка, 7 таблиц;

список использованных источников, включающий 127 наименований и размещенный на 12 страницах;

список 25 публикаций автора на 3 страницах;

7 приложений объемом 25 страниц.



поступлении новых заявок, то задержка времени определяется сложностью обработки последней группы заявок.

Реализация предлагаемой схемы возможна на рекуррентных сетевых моделях, состояние которых соответствует графу текущего паросочетания с выделением оптимального решения. Переход между состояниями сети реализуется инкрементальными версиями алгоритмов решения линейных задач о назначении (ЛЗН), задачи коммивояжера и поиска кратчайших путей на графах. На параметры таких задач проецируются особенности процессов обслуживания, включая векторные критерии и разнообразие отношения вложенности.

Решение классических открытых ЛЗН, записываемых в виде

$$Z = \left\{ \sum_{i=1}^m \sum_{j=1}^n c_{ij} \cdot x_{ij} \mid \sum_{i=1}^m x_{ij} = \sum_{j=1}^n x_{ij} = 1; x_{ij} \geq 0; i = \overline{1, m}, j = \overline{1, n} \right\} \rightarrow \min \quad (1)$$

обычно есть вектор назначений строк матрицы коэффициентов ее столбцам

$$R = \{ r_j = i \mid x_{ij} = 1, i = \overline{1, m}, j = \overline{1, n} \}. \quad (2)$$

Известно, что наиболее эффективные для решения задачи (1) алгоритмы венгерского метода строятся с учетом особенностей двойственной задачи

$$\max \left\{ \sum_{i=1}^m u_i + \sum_{j=1}^n v_j \mid c_{ij} - u_i - v_j \geq 0, i = \overline{1, m}, j = \overline{1, n} \right\}. \quad (3)$$

Предлагается для оптимизации динамического назначения модифицировать инкрементальный алгоритм венгерского метода с целью многократного обновления с вычислительной сложностью  $O(n^2)$  текущего паросочетания для новой или изменяемой строки  $i, i \in \overline{1, m}$ :

```
function hlap(i) {
  u_i = 0; h = ∞;
  foreach k ∈ 1, n {
    d_k = c_ik - v_k; p_k = 0;
    if (h > d_k) h = d_k, j = k;
  }
  while (true) {
    foreach k ∈ 1, n
      if (s_k = false) d_k ← d_k - h; else v_k ← v_k - h, l = r_k, u_l ← u_l + h;
    u_i ← u_i + h; s_j = true; l = r_j;
    if (l = 0) break;
    j* = j; u* = u_l; h = ∞;
    foreach k ∈ 1, n if (s_k = false) {
      d_k ← min(d_k, c_lk - u* - v_k); p_k = j*;
      if (h > d_k) h = d_k, j = k;
    }
  }
  while (p_j > 0) { k = p_j, r_j = r_k, j = k }
  r_j = i;
}
```

Построен инкрементальный алгоритм метода кратчайшего пополняющего пути (Shortest Augmenting Path, SAP), лучшего среди известных методов решения (1). Его преимущества особенно заметны для разреженных графов.

В случае задач динамического назначения параметры  $m$  и  $n$  зависят от времени, поэтому узким местом становятся операции выделения подлежащих реоптимизации строк и столбцов. Пусть в реальном времени формируется поток ЛЗН с номерами  $k = 1, 2, \dots$ . Прямолинейный поиск изменившихся строк в матрицах ЛЗН с номерами  $k$  и  $k + 1$  характеризуется вычислительной сложностью  $O(mn)$ .

Для быстрого отображения изменений матриц предложено использовать векторы  $X^k(i) = k \cdot (c_{ij}^k \equiv c_{ij}^{k-1})$  и  $Y^k(j) = k \cdot (c_{ij}^k \equiv c_{ij}^{k-1})$ ,  $i \in \overline{1, m}$ ,  $j \in \overline{1, n}$ ,  $k > 0$ . Если  $X^0(i) = 0$ ,  $i \in \overline{1, m}$ ,  $Y^0(j) = 0$ ,  $j \in \overline{1, n}$ , а  $c_{ij}^0 = \infty$ ,  $i \in \overline{1, m}$ ,  $j \in \overline{1, n}$ , то операция учета фактов  $c_{ij}^{k+1} \leftarrow c_{ij}^k$  в стеках индексов строк  $H_x^k(t)$  и столбцов  $H_y^k(t)$  на шаге  $t$  формирования матрицы на этапе  $k + 1$ , когда  $t \in \overline{1, T(k+1)}$ , имеет вид:

$$\left\{ \begin{array}{l} h_x^{k+1}(0) = 0; h_y^{k+1}(0) = 0; \\ \forall (t \in \overline{1, T(k+1)}) \wedge (i \in \overline{1, m}) \wedge (j \in \overline{1, n}) \text{ if } (c_{ij}^{k+1}(t) \neq c_{ij}^k) \{ \\ \quad \text{if } (X^{k+1}(i) \neq k+1) \{ \\ \quad \quad h_x^{k+1}(t) \leftarrow h_x^{k+1}(t) + 1, H_x^{k+1}(h_x^{k+1}(t)) \leftarrow i, X^{k+1}(i) \leftarrow k+1; \\ \quad \quad \} \\ \quad \text{if } (Y^{k+1}(j) \neq k+1) \{ \\ \quad \quad h_y^{k+1}(t) \leftarrow h_y^{k+1}(t) + 1, H_y^{k+1}(h_y^{k+1}(t)) \leftarrow j, Y^{k+1}(j) \leftarrow k+1; \\ \quad \quad \} \\ \quad c_{ij}^{k+1} \leftarrow c_{ij}^k(t); \\ \} \end{array} \right. \quad (4)$$

Сложность операции формирования очередной матрицы –  $O(|T(k+1)|)$ .

В таблице 1 приведены результаты экспериментальной оценки среднего времени независимого решения ЛЗН (1) и реоптимизации решения предлагаемыми версиями венгерского метода и метода SAP после изменения одного из случайно выбираемых элементов матриц с применением (4).

**Таблица 1. – Оценка времени реоптимизации и независимого решения ЛЗН**

Размерность, задачи ( $m \times n$ )	Среднее время решения, мсек (Celeron 1,7 ГГц, 512 Мбайт)		
	Венгерский метод	Метод SAP	Реоптимизация
250×500	14,167	19,167	0,533
500×1000	86,000	98,867	0,533
750×1500	344,867	279,567	1,033
1000×2000	757,867	587,800	2,033
1250×2500	1438,400	1088,267	3,067
1500×3000	2509,767	1787,267	5,633
1750×3500	4044,633	2745,167	5,633
2000×4000	6165,533	4006,067	32,200
2250×4500	8810,333	5579,000	33,767
2500×5000	12043,167	7521,700	34,800



**Во второй главе** рассматривалась задача оценки интервалов устойчивости решений открытых и закрытых ЛЗН: для каждого элемента  $c_{ij}$  в (1) необходимо найти интервал  $(a_{ij}, b_{ij})$ , в котором значения таких элементов могут быть изменены без нарушения структуры оптимального назначения (2). Контроль устойчивости назначения обеспечивает грубость системы к незначительным возмущениям элементов или размерности матрицы.

Пусть граф оптимального паросочетания представлен элементами  $E_m = \{(r_j, j) \mid (r_j \leq m), j = \overline{1, n}\}$ , а оставшиеся элементы матрицы образуют  $E_u = \{(i, j), i = \overline{1, m}, j = \overline{1, n}\} \setminus E_m$ . Очевидно, что  $|E_m| = m$ , а  $|E_u| = m \cdot (n - 1)$ , что приходится учитывать при оценке вычислительной сложности алгоритма. Обозначим  $\delta_m(x, y)$  и  $\delta_u(x, y)$  – допустимое изменение веса любого ребра  $x \rightarrow y$ , где  $(x, y) \in E_m \cup E_u$ . Интервалы значений веса ребра, в которых назначение ребра остается неизменным, для задачи минимизации (1) определяются из элементарных рассуждений:

$$I_{\min}^m(x, y) = (-\infty, c_{xy} + \delta_m(x, y)], \quad (x, y) \in E_m;$$

$$I_{\min}^u(x, y) = [c_{xy} - \delta_u(x, y), +\infty), \quad (x, y) \in E_u.$$

Говорят, что ребро графа оптимального паросочетания скрыто, если его вес увеличен так, что ребро больше не является частью существующего решения. Скрытие реализуется назначением веса из интервала  $(c_{xy} + \delta_m(x, y), +\infty)$ .

Известно, что для элементов оптимального паросочетания справедливо  $c_{xy} = u_x + v_y$ ,  $(x, y) \in E_m$ . Пусть оценка оптимального решения ЛЗН есть

$$Z^0 = \sum_{i=1}^m \sum_{j=1}^n c_{ij} \cdot x_{ij}^0 = \sum_{i=1}^m u_i^0 + \sum_{j=1}^n v_j^0. \quad (5)$$

Нетрудно показать, что если скрыто ребро  $x \rightarrow y$ ,  $(x, y) \in E_m$ , то процесс реоптимизации, начинающийся в вершине  $x$ , завершится в вершине  $y$ , потенциал которой не изменится. Меняется только потенциал строки  $u_x$ , а в соответствии с (5):  $u_x^1 - u_x^0 = Z^1 - Z^0$ ,  $x \in \overline{1, m}$ . Здесь нулевой верхний индекс использован для пометки исходного, а единичный – нового решения.

Отсюда следует, что  $\delta_m(x, y) = Z^1 - Z^0$ . Интервал безопасного изменения значений веса скрываемого ребра определяется выражением

$$I_{\min}^m(x, y) = (-\infty, c_{xy} + Z^1 - Z^0] = (-\infty, c_{xy} + u_x^1 - u_x^0], \quad (x, y) \in E_m. \quad (6)$$

Использование разности потенциалов исключает необходимость прямолинейного вычисления оценок решений задачи (1), требующего  $m + n$  шагов.

Подобным образом рассматриваются ребра, не принадлежащие оптимальному решению, когда  $c_{xy} \geq u_x + v_y$ ,  $(x, y) \in E_u$ . Если вес таких ребер менять в интервале  $(u_x + v_y, +\infty)$ , то структура решения (2) остается неизменной.

Пусть  $\varepsilon_u(x, y) = c_{xy} - \delta_u(x, y)$ . Будем считать, что ребро графа паросочетания открыто, если его вес уменьшен так, что это ребро становится частью нового оптимального решения. Открытие ребра наступает в случае, когда

$c_{xy} \leftarrow \varepsilon_u(x, y)$ . Известен метод определения  $\varepsilon_u(x, y), (x, y) \in E_u$  на основе построения вспомогательного графа  $G_a$  из графа оптимального паросочетания путем удаления всех дуг, инцидентных вершинам  $x$  и  $y$ . Если выполнить уменьшение веса  $c_{xy} \leftarrow \varepsilon_u(x, y)$ , то для графа  $G_a$ , дополненного ребром  $(x, y)$ , получим оптимальное паросочетание степени  $m$  с оценкой  $Z^a$ . Таким образом,  $\varepsilon_u(x, y) = Z^0 - Z^a$  – нижняя граница интервала устойчивости, поэтому

$$I_{\min}^u(x, y) = [Z^0 - Z^a, +\infty), (x, y) \in E_u. \quad (7)$$

Однако определение интервала значений веса ребер на основе (7) не является эффективным. Для каждого из  $m \cdot (n-1)$  ребер придется решать ЛЗН, размер которых  $(m-1)(n-1)$ . Предлагается вместо выражения (7) воспользоваться выражением (6), инвертируя направление шагов процесса построения интервала. Конечная граница интервала  $I_{\min}^m(x, y)$  станет начальной границей интервала  $I_{\min}^u(x, y)$ . Нулевой шаг в (6) становится решением ЛЗН для гарантированно приводящего к открытию ребра значения  $c_{xy} = -\infty$ , а единичный шаг соответствует решению ЛЗН с исходной матрицей. В результате получаем

$$I_{\min}^u(x, y) = (-\infty + Z^0 - Z^1, +\infty) = (-\infty + u_x^0 - u_x^1, +\infty), (x, y) \in E_u. \quad (8)$$

Оценка выражения (8) требует лишь одношаговой реоптимизации исходной задачи (1). Вычислительная сложность реализации (6) и (8) –  $O(n^4)$ .

Известно, что алгоритм оценки устойчивости решения задачи коммивояжера имеет экспоненциальную сложность. Однако результат ее решения соответствует решению закрытой ЛЗН с ограничениями:

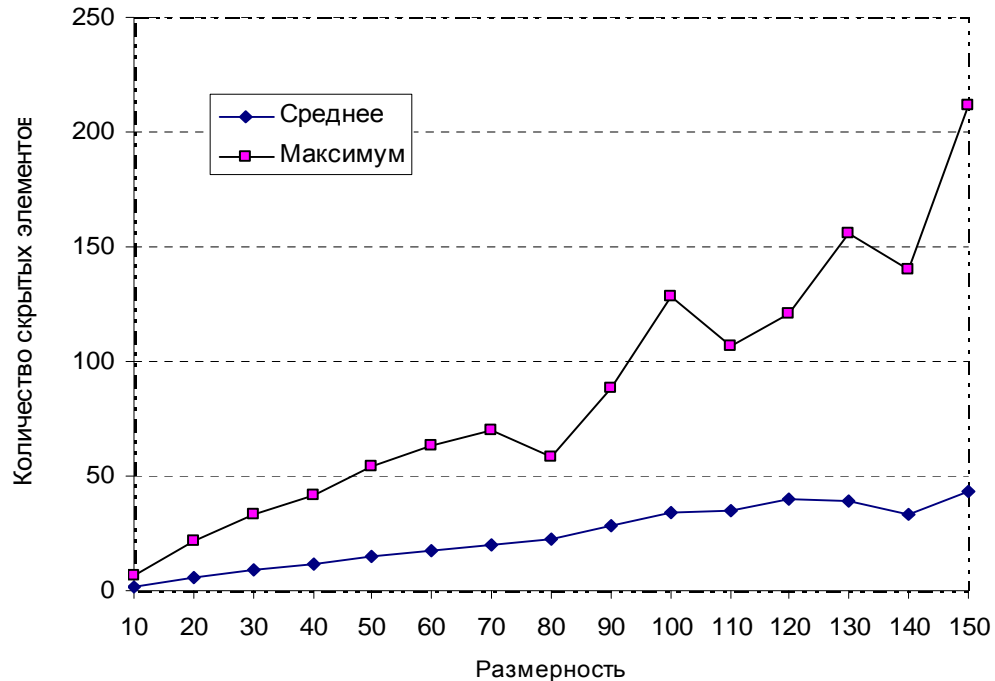
$$Y = \min \left\{ \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \left| \begin{array}{l} \sum_{i=1}^n x_{ij} = \sum_{j=1}^n x_{ij} = 1; x_{ij} \geq 0, i, j = \overline{1, n}; \\ u_i - v_j + n x_{ij} \leq n - 1, i = \overline{2, n}, j = \overline{2, n}, i \neq j \end{array} \right. \right\} \quad (9)$$

Показано, что применение (6) к результату решения (9) методом ветвей и границ практически обеспечивает грубость реакции системы управления многозадачными агентами на незначительные изменения оценок назначения. Проведен статистический эксперимент по оценке количества скрытых элементов матрицы ЛЗН, соответствующей оптимальному гамильтонову циклу (рисунок 2).

Максимальное количество формально скрытых элементов не превышает значения  $O(n)$ . Если учесть, что общее количество скрытых элементов открытой ЛЗН  $n(n-1)$ , то в первом приближении вероятность ошибки в оценке интервала устойчивости будет не более, чем  $(n-1)^{-1}$  (здесь  $n$  – размерность матрицы задачи коммивояжера,  $n > 1$ ).

**В третьей главе** рассмотрено применение агентных схем поиска кратчайших путей. Известно, что для нагруженного ориентированного графа  $G(N, A)$ , где  $N$  – множество вершин,  $A$  – множество дуг с весовой функцией  $W : A \rightarrow R^+$ , время построения дерева путей одним из лучших для подобной за-

дачи алгоритмом Дийкстры растет в первом приближении по закону  $x \cdot \log_2 x$  с увеличением расстояния  $x$  от корня дерева. Реализация алгоритма Дийкстры с отображением очередей анализируемых вершин на кучи Фибоначчи характеризуется сложностью  $O(m + n \cdot \log_2 n)$ , где  $m = |A|$ ,  $n = |N|$ . Отображение очереди вершин на вектор размером  $L$  позволяет снизить сложность до величины  $O(m + n \cdot L)$ , где  $L$  – максимальная длина дуги графа.



**Рисунок 2.** – Оценки количества скрытых элементов итоговой матрицы ЛЗН

Пусть процедура алгоритма Дийкстры использует  $D$  – вектор расстояний от корня дерева,  $P$  – вектор номеров предшествующих вершин,  $Q$  – очередь вершин, элементы которой упорядочены по текущему значению расстояния от корня дерева. Процесс построения дерева кратчайших путей имеет волновой характер до исчерпания возможности его развития из исходной вершины. Отображение состояния вершин на множествах  $D$  и  $P$  вынуждает каждый раз перед построением дерева устанавливать начальные значения для всех элементов. Вычислительная сложность такой операции –  $O(n)$ . Очевидно, что в случае построения кратчайших путей на больших графах количество вершин дерева может оказаться существенно меньше значения  $n$ .

Учитывая последовательный характер изменения состояния вершин дерева в соответствии со значением расстояния от его корня, очевидна идея логического включения вершин новых деревьев в объединенное дерево поиска. Это легко достигается дополнением признака состояния вершин дерева номером этапа поиска  $e$ , что позволяет исключить повторную установку начального состояния множеств  $D$  и  $P$ . Предложенный способ нумерации вершин деревьев определяет асимптотику сложности реализации поиска в зависимости от количества дуг дерева.

В практически важных задачах поиска пути между двумя заданными вершинами графа целесообразно организовать поиск путем построения двух

встречно растущих деревьев. В результате объем анализируемых данных сокращается в два раза. Предлагается новая модель данных такого поиска, позволяющая реально обеспечить его вычислительную сложность  $O(m + n \cdot L)$ .

Обозначим исходный граф через  $G^+ = G(N^+, A^+)$ , а граф с инвертированием направления дуг –  $G^* = G(N^*, A^*)$ . Показано, что линейная функция  $conj(x) = 2n - 1 - x$ ,  $x \in N^+ \cup N^*$  реализует прямое и обратное отображение номеров вершин  $N^+ \leftrightarrow N^*$ . Объединение множеств таких номеров соответствует неразрывной последовательности  $\overline{0, 2n-1}$ . В результате множество последовательных номеров вершин компактной модели графа  $G^*$  определяется так:  $N^* = \{x^* = conj(x^+), x^+ \in N^+\}$ ,  $A^* = \{(conj(y^+), conj(x^+)), (x^+, y^+) \in A^+\}$ .

Пусть  $s$  и  $t$  – начальная и конечная вершины искомого кратчайшего пути на графе  $G^+$ . Так как  $N^+ \cap N^* = \emptyset$ , то встречный поиск можно проводить синхронным движением волны от корней деревьев на несвязном графе  $G^+ \cup G^*$ . Для этого достаточно начать процесс ветвления из вершин  $s \in G^+$  и  $t^* = conj(t)$ ,  $t^* \in G^*$ . Последнее соответствует формальному объединению графов фиктивной дугой  $s \rightarrow t^*$ , для которой  $w(s, t^*) = \infty$ .

Организация встречного поиска требует экономного определения правила остановки. Если для некоторого дерева кратчайших путей максимальное расстояние от постоянно помеченных вершин до корня есть  $d$ , то признаком постоянной пометки вершины  $x$  является условие  $D(x) \leq d$ . В рассматриваемом случае для обоих деревьев значение  $d$  одинаково. Отсюда следует очевидное правило остановки –  $D(conj(i)) \leq D(i)$ , где  $i$  – вершина графа  $G^+$  или графа  $G^*$ , получающая постоянную пометку. Однако проблема состоит в дискретном характере процедуры выбора помечаемых вершин, когда условие  $D(conj(i)) \leq D(i)$  приходится проверять каждый раз после коррекции значений расстояний до временно помеченных вершин.

Обозначим множества листьев встречно растущих деревьев кратчайших путей через  $T_x^+ = \{i \mid x \leq D(i) < \infty, i \in N^+\}$  и  $T_x^* = \{i \mid x \leq D(i) < \infty, i \in N^*\}$ , где  $x = \min\{D(k), k \in N^+ \cup N^*\}$ . Очевидно, что на любом этапе процесса развития деревьев остановка может произойти в любой вершине множества  $K(x) = \{i \mid i \in \{conj(i), i \in T_x^+\} \cap T_x^*\}$ . Так как  $K(x) \subseteq Q$ , то предикат определения  $K(x)$  может включать выражение  $(P(x) \geq e)$  вместо  $(D(x) < \infty)$ , что исключает необходимость повторной инициализации переменных состояния.

Действительно, перед началом поиска  $K(0) = \emptyset$ . Состояние  $|K(x)| \geq 1$  будет соответствовать предикату  $(i = conj(i)) \wedge (x \leq D(i) < \infty) \wedge (x \leq D(conj(i)) < \infty)$ , который эквивалентен  $D(conj(i)) \leq \infty$  при включении вершины в очередь.

На последующих итерациях включения элементов в очередь определим значение  $d_{\min} = \min_k \{D(k) + D(conj(k)), k \in K(x)\}$ . Значение  $D(k)$  только возрастает, а из множества временно помеченных вершин исключаются элементы,

для которых  $(D(k) < x)$ . Это условие соответствует моменту установки постоянной пометки вершины, одна из которых соответствует условию  $d_{\min} \equiv \min_k \{D(k) + D(\text{conj}(k)), k \in K(x)\}$ . Так как для любого значения  $x$  новые элементы множества  $K(x)$  будут иметь расстояние до корней, не меньшее  $x$ , то условием останова в момент постоянной пометки вершины  $k$  будет  $d_{\min} \equiv D(k) + D(\text{conj}(k)), k \in K(x)$ . Таким образом, использование встречного движения от корней деревьев не требует хранения специальных пометок, а момент останова совпадает с моментами окончательной пометки вершин дерева путей.

Эффективный прием фильтрации просматриваемых дуг графа – каждой дуге поставить в соответствие список вершин, кратчайшие пути к которым включают такую дугу. Построение подобных списков возможно после предварительного однократного построения всех деревьев кратчайших путей.

Предлагается учесть ассоциации дуг с оптимальными решениями характеристическими множествами признаков вхождения вершин в заранее выделенные любым способом подмножества вершин. Принадлежность вершины  $i$  подмножеству  $k$  пусть отражается выражением  $B(i) = 2^k, i \in N$ .

Алгоритм учета ассоциаций дуг следующий. Первоначально каждой дуге графа следует назначить нулевой вектор характеристического множества  $S(I(i, j)) = 0, (i, j) \in M$ . Здесь  $I(i, j) = (k | (i, j) \in M) \wedge (k \in \overline{1, m})$  – индекс дуги в списке дуг. Далее для каждой вершины  $i, i \in N$ , необходимо построить дерево кратчайших путей до всех остальных вершин, но при этом помимо операции  $P(j) \leftarrow i$  следует сохранить индекс  $R_j \leftarrow I(i, j), j \in \overline{1, n}$ , указывающий позицию дуги  $i \rightarrow j$  в списке дуг. После этого узлы и листья деревьев кратчайших путей могут эффективно отображаться на дуги графа:  $S(k) \leftarrow S(k) \vee B(P(j)), (R_j = k) \wedge (P(j) \neq j), j \in N$ . Условие фильтрации дуг  $(P(j) \leq e) \vee ((D_j > D_i + w(i, j)) \wedge (Q \leftarrow Q \setminus \{j\}))$  должно быть заменено на  $((S(I(i, j)) \wedge B(t)) \neq 0) \wedge ((P(j) \leq e) \vee ((D_j > D_i + w(i, j)) \wedge (Q \leftarrow Q \setminus \{j\})))$ .

В таблице 3 приведены результаты оценки среднего времени решения задач расчета подматриц кратчайших расстояний предлагаемыми процедурами для графа реальной сети автомобильных дорог, где  $n \cong 10^6, m \cong 5 \cdot 10^6$ .

**Таблица 3. – Оценки времени расчета подматриц кратчайших расстояний**

Размерность задачи $( S )$	Среднее время решения, мсек (Celeron 1,7 ГГц, 512 Мбайт)						
	Классическая схема Дijkstra	Предлагаемая схема с различным числом подграфов					
		2	8	16	32	64	128
500	10,823	8,003	1,038	0,719	0,524	0,512	0,503
600	12,234	9,514	1,312	0,828	0,640	0,604	0,602
700	14,534	11,005	1,513	0,947	0,769	0,678	0,688
800	17,002	12,811	1,716	1,102	0,924	0,860	0,772
900	20,712	15,012	2,117	1,417	1,187	1,009	0,901

Реоптимизация решения ЛЗН и предложенные схемы поиска кратчайших путей – эффективные приемы повышения быстродействия решения задач коммивояжера методом ветвей и границ на разреженных графах (таблица 2).

Таблица 2. – Оценки времени решения задач коммивояжера

Размерность задачи (n)	Среднее время решения, сек (Celeron 1,7 ГГц, 512 Мбайт)					
	Полустепень исхода вершин графа					
	n/10	n/4	n/2	2n/3	3n/4	n-1
50	0,003	0,008	0,019	0,024	0,032	0,036
60	0,004	0,012	0,028	0,040	0,048	0,052
70	0,005	0,013	0,047	0,063	0,078	0,088
80	0,011	0,016	0,102	0,128	0,160	0,172
90	0,012	0,017	0,117	0,137	0,169	0,186
100	0,019	0,024	0,136	0,192	0,224	0,244
120	0,055	0,067	0,253	0,358	0,400	0,448
140	0,083	0,116	0,488	0,621	0,724	0,862
150	0,122	0,144	0,548	0,769	1,027	1,193

В четвертой главе рассмотрены вопросы построения и реализации контура управления (рисунок 1) на основе модельных архитектур в рамках концептуальной схемы федерата (согласно стандарта *HLA* (High Level Architecture)). Основа таких структур – рекуррентные варианты расширенных временных сетей Петри, переходы которых определяются шаблонами полиморфных классов.

Показано, что основа представления перехода рекуррентных сетей – пятерка функций  $S_a = (E_a, C_a, D_a, F_a, I_a)$  с общей областью определения переменных состояния. Здесь  $E_a$  – условия активизации перехода  $a$ ;  $C_a$  – действия при активизации перехода;  $D_a$  – длительность активной фазы;  $F_a$  – действие перехода при выходе из активного состояния;  $I_a$  – действия над переходом при внешнем прерывании.

Процесс активизации отдельного перехода, подобно переходам сетей Петри, предлагается связывать с его восприимчивостью к изменению переменных  $B_a = \text{dom}(E_a) \cup \text{dom}(C_a) \cup \text{dom}(F_a) \cup \text{dom}(I_a)$ , где  $\text{dom}(f)$  – множество переменных состояния, связываемых функцией  $f$ . Обозначим  $x'$  – множество выходных элементов любой вершины  $x$  ориентированного графа. Определим двудольную сеть  $IPN=(A,B,V)$ , где  $A$  – переходы,  $B$  – переменные состояния – аналог позиций сети Петри,  $V=(A \times A) \rightarrow \{0,1\}$ . Отображение  $V$  сети  $IPN$  в форме структуры смежности определяется на основе

$$a' = \{x \mid \text{dom}(E_a) \cap (\text{dom}(F_x) \cup \text{dom}(I_x)) \neq \emptyset, x \in A\}, a \in A.$$

В отличие от сетей Петри, способность переходов  $a \in A$  динамического формирования  $a'$  позволяет строить сети высоких порядков, для чего в  $B_a$  связи шагов рекурсии задаются ссылками на порождающие переходы. Предлагается на элементах  $a'$  отобразить эффективность альтернатив развития процесса из вершины  $a$ . Назначение упорядочения конкурирующих альтернатив для всех  $a \in A$  – результат решения (2). В итоге подсеть  $IPN$ , включающая лишь высо-

коприоритетные дуги, становится сетью Петри свободного выбора, что исключает необходимость последующего устранения конфликтов использования ресурсов.

На примере построения системы управления обслуживанием заявок на поставку бетонных растворов с учетом технологических и временных ограничений (рисунок 3) показана схема построения рекуррентных сетей второго порядка и их комплексирования с процедурами назначения (рисунок 4).

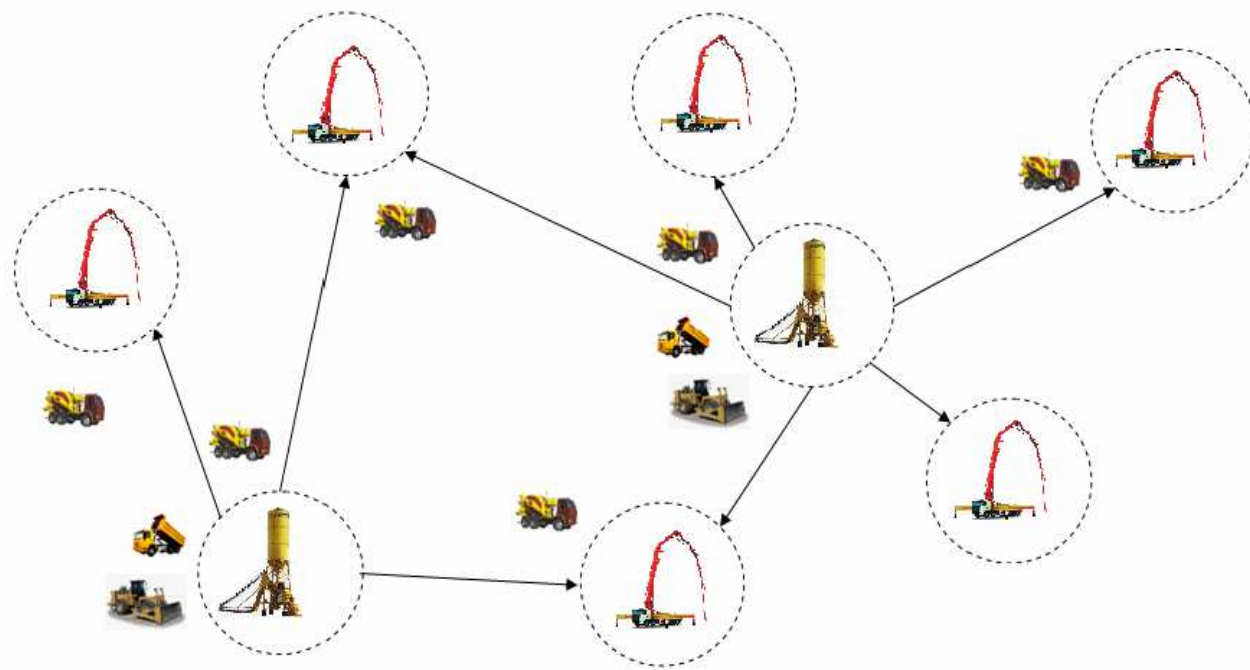


Рисунок 3. – Пример транспортной системы доставки бетона

Задача моделирования процессов на *IPN* формулируется так: задана сеть  $IPN = (A, B, V)$ , необходимо построить последовательность событий, представляемых парами  $L_k = (t_a, a)_k$ ,  $k = 0, 1, \dots$ , где  $k$  – номер этапа, когда в момент времени  $t_a$  деактивирован переход  $a \in A$ .

Рекуррентная схема моделирования процессов на *IPN* имеет вид:

$$L_0 = \langle 0, s \rangle, s \in A; L_{k+1} = F_6(F_1, F_2(F_3(F_4(F_5(L_k))))), k > 0. \quad (10)$$

При этом на любом этапе  $k, k \geq 0$ , предполагается использование следующих процедур обработки факта деактивизации перехода  $a \in A$ :

$F_1$  – вызов модуля выходных действий перехода  $a$ ;

$F_2$  – организация обработки последствий локального изменения переменных состояния;

$F_3$  – оценка возможности активизации новых переходов из множества  $a'$ ;

$F_4$  – активизация перехода  $x$ , если необходимость этого установлена процедурой  $F_3$  (активизация перехода  $x$  влечет вызов функции входных действий и планирование нового события в момент  $t_a + D_x$ );

$F_5$  – фиксация в списке событий нового особого события, представляемого парой  $(t_a + D_x, x)$ ;

$F_6$  – выбор очередного события на  $IPN$  и повторение цикла моделирования, если список событий не пуст.

Предложенная схема моделирования (10) в объектно-ориентированной реализации переходов  $IPN$  пригодна для комплексирования методов имитационного моделирования и оптимизации на алгоритмическом уровне.

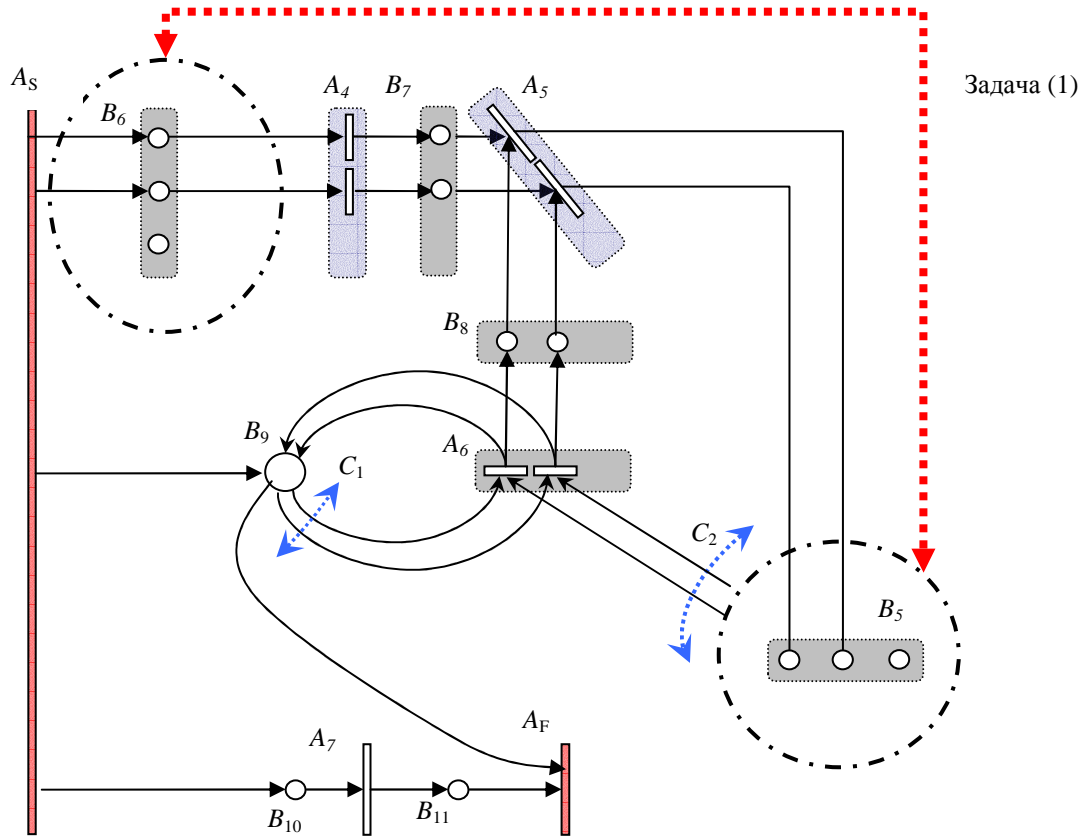


Рисунок 4. – Схема решения задачи динамического назначения

Этапы обработки фактов деактивизации переходов непосредственно определяют моменты вызова процедур выбора и назначения альтернатив развития процесса. Например, решение ЛЗН в виде (1) позволяет назначить приоритет обслуживания заявок из очереди с учетом текущего состояния (рисунок 4).

Конкретизация в смысле объектных технологий переходов  $IPN$  и (10) позволили легко решать задачи распараллеливания процессов оптимизации для кооперативного использования ресурсов распределенной вычислительной среды. Например, рассмотрена комбинаторная задача размещения растворобетонных узлов (рисунок 3) по критерию минимума транспортных расходов с учетом изменения объема и адресов поставок. Предложена процедура поиска вариантов размещения путем реоптимизации решения транспортных подзадач, реализующая схему гонок с выбыванием бесперспективных вариантов.

Рассмотрен способ компактного представления в произвольный момент состояния задачи поиска, допускающий прерывание процесса решения локальных задач, последующее восстановление состояния и продолжения процесса решения на любом доступном узле сети. Предложенный способ основан на



расширении функционального замыкания интервала активности перехода, реализующего отдельный шаг рекурсии.

Пространство поиска решения задач, например, методом ветвей и границ или динамического программирования определяется рекурсивно. Это делает нетривиальной задачу сохранения и восстановления состояния процесса поиска решения в случае его прерывания и миграции на новую ЭВМ, а также для высокоуровневого распараллеливания на множестве ветвей дерева вариантов.

Альтернативы для параллельного решения могут быть оформлены как переходы  $IPN$  из формального описания задачи. Например, ветвление дерева вариантов задачи коммивояжера (9) удобно проводить на множестве ЛЗН (1). Можно заметить, что если  $k$  – некоторая вершина вектора решения (2), то на уровне ветвления  $l$  последовательность

$$r^l(k) = \{r(0) = k, \{r(i) = R_{r(i-1)}^l | r(i) \neq k\}\} \subseteq R^l$$

только тогда соответствует гамильтонову циклу, когда  $r(n-1) = k$ ,  $k \in \overline{1, n}$ . Если цикл не гамильтонов, то есть  $r^l(k) \subset R^l$ , то необходимо породить множество задач уровня  $l+1$ . Для этого следует указать цикл минимальной длины, выбрав вершину входа в цикл  $k^l = \arg \min_k \{|r^l(k)|, k \in \overline{1, n}\}$ . Правило порождения ЛЗН тривиально – для каждой вершины обнаруженного цикла необходимо запретить посещение других вершин этого цикла. При этом матрица очередной ЛЗН отличается от предыдущей одной строкой, в которой часть элементов заменяются значением, не меньшим значения  $c_{\max} = \max_{i, j} \{c_{ij} : i \neq j, i \in \overline{1, n}, j \in \overline{1, n}\}$ . Построчное изменение матриц порождаемых ЛЗН проводится по закону:

$$c_{ij}^{l+1} = c_{ij}^l + c_{\max} (1 \cdot ((i \in r^l(k^l)) \wedge (j \in r^l(k^l)))) , i, j = \overline{1, n}.$$

В итоге потребность в памяти для представления всего процесса решения задачи коммивояжера –  $O(2n^2 + n)$ .

## ЗАКЛЮЧЕНИЕ

### Основные научные результаты диссертации

1. Предложена новая модель и алгоритм [24] поиска оптимального паросочетания агентов и решаемых ими задач на основе метода кратчайшего пополняющего пути, отличающихся от известных моделей использованием компактных рекуррентных схем реоптимизации решений, что обеспечивает линейную зависимость времени реакции системы на внешние события [2].

2. Разработаны алгоритмы быстрой оценки устойчивости решения открытых и закрытых линейных задач о назначении [6,7,10,11], а также задач коммивояжера [13,19,20], использующие принцип реоптимизации решения [12] после инверсии состояния дуг графа оптимального паросочетания [1,2], что позволило снизить вычислительную сложность таких алгоритмов и обеспечить грубость системы управления к внешним структурно-параметрическим возмущениям [9,14].

3. Разработаны эффективные по памяти и быстродействию алгоритмы многократного поиска кратчайших путей на графах абстрактных транспортных сетей с ограничениями [3,17,18,21] агентами обхода вершин графа, учитывающие реальные технологические и временные ограничения и обеспечивающие быстореактивную обработку потока запросов на выборку маршрутов в реальном времени [15,23].

4. Предложены модели реализации целеустремленного поведения агента на основе прямого и обратного поиска кратчайших путей на графах, учитывающие ограничения на структуры путей и предопределенные решения, позволяют снизить степень вычислительной сложности до диаметра графа, что существенно для реализации быстродействующих систем[3].

5. Предложен алгоритм реоптимизации решения транспортных подзадач методом потенциалов с контролем бесперспективности варианта в комбинаторных задачах оптимизации размещения по схеме кооперации ресурсов вычислительной среды[25], позволяющий наряду с учетом неоднородных ограничений улучшить динамические характеристики процесса поиска решений в реальном времени [18].

6. Предложены и обоснованы схемы алгоритмов грануляции задач комбинаторного вида для репликации и миграции агентов распределенных вычислений, что позволяет обеспечить в реальном времени целостность и безопасность процедур метода динамического программирования[4,5,8,15,22], а также метода ветвей и границ[19,20].

### **Рекомендации по практическому использованию результатов**

Созданные модели и алгоритмы динамического перераспределения задач рекомендуются для построения систем координации взаимодействующих агентов как средство повышения эффективности назначения путем учета актуальной информации о фактическом состоянии системы в наиболее поздний момент.

Алгоритмы оценки устойчивости назначения позволяют определить границы области изменения локальных оценок эффективности для быстрой классификации полезности новых альтернативных средств решения задач.

Модели и быстродействующие алгоритмы поиска кратчайших путей на графах с ограничениями и автоматическим формированием и накоплением предопределенных решений рекомендуются для эффективного по быстродействию решения оптимизации на транспортных сетях.

Приведенные в приложениях диссертации исходные тексты программной реализации предложенных моделей и алгоритмов на языке программирования С++ могут быть использованы как базовые шаблоны реализации систем автоматического управления дискретными процессами в терминах ресурсов и задач.

Результаты диссертационной работы использованы в научно-исследовательском процессе на кафедре информационных технологий БГУИР, а также при проектировании бизнес-процессов компании Transport & Construction Co (Доха, Катар).

## СПИСОК ПУБЛИКАЦИЙ СОИСКАТЕЛЯ

### Статьи в научных рецензируемых журналах

1. Ревотюк, М.П. Быстрая оценка интервалов устойчивости решения линейных задач о назначении/М.П. Ревотюк, М.К. Кароли, П.М. Батура//Доклады БГУИР, № 5(75), 2013. – С. 30-36.
2. Ревотюк, М.П. Реализация метода ветвей и границ для решения задач коммивояжера с разреженными матрицами/М.П. Ревотюк, М.К. Кароли, П.М. Батура//Доклады БГУИР, № 7(77), 2013. – С. 25-31.
3. Ревотюк, М.П. Быстрый поиск кратчайших путей на графах с предопределенными решениями/М.П. Ревотюк, М.К. Кароли, Н.В. Хаджинова//Доклады БГУИР, № 4(82), 2014. – С. 73-79.

### Тезисы и тексты докладов в сборниках и материалах конференций

4. Ревотюк, М.П. Безопасное прерывание процедур метода динамического программирования//М.П.Ревотюк, М.К. Кароли// X Белорусско-российская научно-техн. конф. “Технические средства защиты информации” (29 – 30 мая 2012 г., Минск). – Минск: БГУИР, 2012. – С. 45.
5. Зобов, В.В. Миграция процесса решения задач методом динамического программирования/В.В. Зобов, М.К. Кароли, М.П.Ревотюк//48-я научная конф. аспирантов, магистрантов и студентов “Информационные технологии и управление” (07 - 11 мая 2012 г, Минск) – С.14.
6. Ревотюк, М.П. Оценка интервалов устойчивости решения задач о назначении/М.П. Ревотюк, М.К. Кароли//VII Междунар. научно-техн. конф. «Информационные технологии в промышленности» (IT'2012): тезисы докладов (30 - 31 октября 2012 года, Минск). – Минск: ОИПИ НАН Беларуси, 2012. – С.121-122.
7. Ревотюк, М.П. Оценка устойчивости решений открытых задач о назначении/М.П. Ревотюк, М.К. Кароли//Информационные технологии и системы: мат. Междунар. конф. БГУИР(Минск, 24 октября 2012 г.). – Минск, БГУИР, 2012. – С.319-320.
8. Ревотюк, М.П. Репликация процедур динамического программирования в системах агентов/М.П.Ревотюк, В.В.Зобов, М.К. Кароли// Информационные технологии и системы: мат. Междунар. конф. БГУИР (Минск, 24 октября 2012 г.). – Минск, БГУИР, 2012. – С.293-294.
9. Ревотюк, М.П. Оценка необходимости пересмотра состава заданий//М.П. Ревотюк, М.К. Кароли, Т.В. Тиханович//Мат. VI Междунар. научно-метод. конф. «Высшее техническое образование: проблемы и пути развития» (Минск, 28-29 ноября 2012 г.). – Минск, БГУИР, 2012. – С.275.

10. Ревотюк, М.П. Определение интервалов устойчивости решений открытых задач о назначении/М.П. Ревотюк, М.К. Кароли//Мат. XVIV Междунар. научно-техн. конф. “Информационные системы и технологии (ИСТ’2013)” (19 апреля 2013 г., Н.Новгород). – Н.Новгород, НГТУ им. Р.Е. Алексеева, 2013 – С. 264.
11. Кароли, М.К. Оценка устойчивости решений задач коммивояжера// М.К. Кароли, М.П. Ревотюк/49-я научная конф. аспирантов, магистрантов и студентов «Информационные системы и технологии» (Минск, 4 мая 2013 г.). – Минск, БГУИР, 2013 – С. 11.
12. Батура, П.М. Быстрый пересчет открытых и закрытых задач о назначении//П.М.Батура, М.К. Кароли, О.В. Кот, М. П. Ревотюк/49-я научная конф. аспирантов, магистрантов и студентов «Информационные системы и технологии» (Минск, 4 мая 2013 г.). – Минск, БГУИР, 2013 – С. 61.
13. Кароли, М.К. Нижние границы оценок решений задач коммивояжера//М.К. Кароли, Н.В. Хаджинова, Т.В. Тиханович, М. П. Ревотюк/ 49-я научная конф. аспирантов, магистрантов и студентов «Информационные системы и технологии» (Минск, 4 мая 2013 г.). – Минск, БГУИР, 2013 – С. 12.
14. Ревотюк, М.П. Устойчивость сервисных систем группового обслуживания/М.П. Ревотюк, М.К. Кароли//XI Белорусско-российская научно-техн. конф. “Технические средства защиты информации” (5-6 июня 2013 г., Минск) – Минск: БГУИР, 2013. – С. 46.
15. Ревотюк, М.П. Кэширование состояния распределенных процессов системами агентов/М.П. Ревотюк, М.К. Кароли, В.В. Зобов//XI Белорусско-российская научно-техн. конф. “Технические средства защиты информации” (5 - 6 июня 2013 г., Минск). – Минск: БГУИР, 2013. – С. 47.
16. Ревотюк, М.П. Быстрая оценка интервалов устойчивости решений задач коммивояжера/М.П. Ревотюк, М.К. Кароли//«Информационные технологии и системы 2013»(ITS’2013): мат. Междунар. конф. (Минск, 23 октября 2013 г.). – Минск, БГУИР, 2013. – С. 312-313.
17. Кароли, М.К. Встречный поиск кратчайших маршрутов на сетях с предопределенными решениями /М.К. Кароли, М.П. Ревотюк //Современные проблемы математики и вычислительной техники: Мат. VIII Респ. научной конф. молодых ученых и студентов (Брест, 21-23 ноября 2013 г.). – Брест: БГТУ, 2013. – С. 125-128.
18. Ревотюк, М.П. Акселераторы процедур логического вывода в системах оценки компетенций/М.П.Ревотюк, М.К.Кароли, Т.В.Тиханович//Дистанционное обучение – образовательная среда XXI века: мат. VIII междунар. науч.-метод. конф. (Минск, 5-6 декабря 2013 года). – Минск: БГУИР, 2013. – С. 310-311.
19. Ревотюк, М.П. Анализ устойчивости решений задач коммивояжера/ М.П. Ревотюк, М.К. Кароли, О.В. Кот// Междунар. научно-техн. конф., приуроченная к 50-летию МРТИ–БГУИР (Минск, 18-19 марта 2014 года): мат. конф. В 2 ч. Ч. 2/редкол.: А. А. Кураев [и др.]. – Минск: БГУИР, 2014. – С. 252-253.

20. Кароли, М.К. Алгоритм переоценки устойчивости решений задач коммивояжера/М.К. Кароли, О.В. Кот//50-я научная конф. аспирантов, магистрантов и студентов «Информационные системы и технологии» (Минск, 24-28 марта 2014 г.). – Минск, БГУИР, 2014 – С. 23.

21. Кароли, М.К. Алгоритм быстрого расчета подматриц кратчайших расстояний на транспортных сетях/М.К. Кароли, В.В. Наймович//50-я научная конф. аспирантов, магистрантов и студентов «Информационные системы и технологии» (Минск, 24-28 марта 2014 г.). – Минск, БГУИР, 2014 – С. 26

22. Ревотюк, М.П. Безопасное распределение процедур метода динамического программирования/М.П.Ревотюк, М.К.Кароли, В.В.Наймович// XII Белорусско-российская научно-техн. конф. «Технические средства защиты информации» (28-29 мая 2014 г., Минск). – Минск: БГУИР, 2014. – С. 34.

23. Ревотюк, М.П. Ускорение многократного поиска кратчайших путей на графах/М.П. Ревотюк, М.К. Кароли, О.В. Кот, В.В. Наймович// «Информационные технологии и системы 2014»(ITS'2014): мат. Междунар. конф. (29 октября 2014 г., Минск). – Минск, БГУИР, 2014. – С. 304-305.

24. Ревотюк, М.П. Инкрементальный алгоритм решения динамической линейной задачи о назначении/М.П. Ревотюк, М.К. Кароли, Р. Хормози// «Информационные технологии и системы 2015»(ITS'2015): мат. Междунар. конф. (Минск, 28 октября 2015 г.). – Минск, БГУИР, 2015. – С. 288-289.

25. Ревотюк, М.П. Динамическое перераспределение мест размещения сервисов/М.П. Ревотюк, М.К. Кароли, Р.В. Кругликов//XIV Белорусско-российская научно-техн. конф. «Технические средства защиты информации» (25 мая 2016 г., Минск). – Минск: БГУИР, 2016. – С. 39.

### **Отчеты о НИР**

26. Методы, модели и технологии, современных информационно-аналитических и управляющих систем / БГУИР; рук. темы В.С. Муха. – Минск, 2016. – 177 с. – № ГР 20112038.

## РЭЗІЮМЭ

АЛКАРАЛІ Мухіеддын Кэмэл Салман

### Дынамічнае пераразмеркаванне задач у сістэмах ўзаемадзеянняў агентаў

**Ключавыя словы:** мультыагентныя сістэмы, аптымальнае кіраванне, задача аб дынамічным прызначэнні, ўстойлівасць прызначэння, задача коміваяжора, пашыраныя сеткі Петры, вылічальная складанасць.

**Мэта працы:** распрацоўка мадэляў і эфектыўных па хуткадзейнасці алгарытмаў аператыўнага размеркавання патокаў задач у сістэмах ўзаемадзеянняў агентаў з цэнтралізаваным або калектыўным кіраваннем.

**Метады даследавання і выкарыстаная апаратура:** метады імітацыйнага мадэлявання, дынамічнага, лінейнага і цэлалікавага праграмавання, тэорыі графаў і канчатковых аўтаматаў, сетак Петры, тэхналогіі аб'ектна-арыентаванага праграмавання і праектавання, сістэмы праграмавання C++ і персанальныя ЭВМ.

**Атрыманыя вынікі і іх навізна:** распрацаваны мадэлі і алгарытмы дынамічнага размеркавання задач у сістэмах агентаў, якія выкарыстоўваюць схему папэраджальнай падрыхтоўкі аптымальнага рашэння для скарачэння затрымкі часу перад рэалізацыяй плана. Дынамічная задача аб прызначэнні разглядаецца на рэкурэнтных сетках пераходаў, якія прадстаўляюць актыўныя фазы рашэння задач агентамі.

Прапанаваны алгарытмы ацэнкі ўстойлівасці рашэнняў задач аб прызначэнні і задач коміваяжора, адрозныя ад вядомых выкарыстаннем хуткага аналізу наступстваў страты ўстойлівасці. Гэта забяспечвае неадчувальнасць сістэмы кіравання да абурэнняў абмежаванняў і параметраў мэтавай функцыі.

Распрацаваны эфектыўныя па памяці і хуткадзейнасці алгарытмы сустрэчнага пошуку дапушчальных і найкароткіх маршрутаў на сетках з папярэдне пэўнымі рашэннямі для выкарыстання ў сістэмах аператыўнага кіравання ў рэальным часе.

Разгледжаны метады эфектыўнай грануляцыі і міграцыі агентаў, якія вырашаюць камбінаторныя задачы выбару аптымальных варыянтаў на размеркаваных вылічальных асяроддзях.

**Рэкамендацыі па выкарыстанні:** вынікі даследавання дастасавальныя для абгрунтавання архітэктуры і метадаў рэалізацыі сістэм аператыўнага кіравання сістэмамі з дынамічным пераразмеркаваннем задач і рэсурсаў з падвышанымі патрабаваннямі па хуткадзейнасці.

**Вобласць ужывання:** транспартныя сістэмы, лагістыка, робататэхніка, навігацыя, беспілотныя авіяцыйныя комплексы.

## РЕЗЮМЕ

АЛКАРОЛИ Мухиеддин Камел Салман

### **Динамическое перераспределение задач в системах взаимодействующих агентов**

**Ключевые слова:** мультиагентные системы, оптимальное управление, задача о динамическом назначении; устойчивость назначения, задача коммивояжера, расширенные сети Петри, вычислительная сложность.

**Цель работы:** разработка моделей и эффективных по быстродействию алгоритмов оперативного распределения потоков задач в системах взаимодействующих агентов с централизованным или коллективным управлением.

**Методы исследования и использованная аппаратура:** методы имитационного моделирования, динамического, линейного и целочисленного программирования, теории графов и конечных автоматов, сетей Петри, технологии объектно-ориентированного программирования и проектирования, системы программирования С++ и персональные ЭВМ.

**Полученные результаты и их новизна:** разработаны модели и алгоритмы динамического распределения задач в системах агентов, использующих схему упреждающей подготовки оптимального решения для сокращения задержки времени перед реализацией плана. Динамическая задача о назначении рассматривается на рекуррентной сети переходов, представляющих активные фазы решения задач агентами.

Предложены алгоритмы оценки устойчивости решений задач о назначении и задач коммивояжера, отличающиеся от известных использованием быстрого анализа последствий потери устойчивости. Это обеспечивает нечувствительность системы управления к возмущениям ограничений и параметров целевой функции.

Разработаны эффективные по памяти и быстродействию алгоритмы встречного поиска допустимых и кратчайших маршрутов на сетях с предопределенными решениями для использования в системах оперативного управления в реальном времени.

Рассмотрены методы эффективной грануляции и миграции агентов, которые решают комбинаторные задачи выбора оптимальных вариантов на распределенных вычислительных средах.

**Рекомендации по использованию:** результаты исследования применимы для обоснования архитектуры и методов реализации систем оперативного управления системами с динамическим перераспределением задач и ресурсов с повышенными требованиями по быстродействию.

**Область применения:** транспортные системы, логистика, робототехника, навигация, беспилотные авиационные комплексы.

## SUMMARY

ALQARALEH Muhyeeddin Kamel Salman

### Dynamic reassignment of tasks in systems of cooperating agents

**Keywords:** multi agent systems, optimal control, dynamic assignment problem, stability of assignment, traveling salesman problem, extended Petri nets, computational complexity.

**Aim of the work:** development of models and efficient algorithms for speed operational tasks distribution streams in a centralized or collective management systems of interacting agents.

**Research methods and used facilities:** methods of simulation, dynamic, linear and integer programming, graph theory and finite automata, Petri nets, the technology of object-oriented programming and design, C ++ programming systems and personal computers.

**Obtained results and their novelty:** models and algorithms for dynamic allocation of tasks in the agent system using a pre-emptive preparation of optimal solutions to reduce the time delay to the implementation of the plan are proposed. Dynamic assignment problem is considered in the recurrent transition network representing the active phase of the problem-solving agents.

The algorithms assess the stability of solutions of problems on the assignment and traveling salesman problems, different from the known to use a strictly analysis buckling effects. This ensures insensibility flow rate control system to perturbations of constraints and parameters of target-howling function.

The effective memory speed and counter-search algorithms are acceptable and shortest routes to networks predetermines fission solutions for use in the operational management systems in real time.

The methods of effective granulation and migration of agents that solve combinatorial problems of selecting the best options in the distributed computing environments are proposed.

**Use guidelines:** results of the study are applicable to studies of architecture and methods of implementation of the operational control system of systems with dynamic reallocation of tasks and resources with high demands on speed.

**Application area:** transportation systems, robotics, navigation, unmanned aircraft systems.



*Научное издание*

АЛКАРОЛИ Мухиеддин Камел Салман

**ДИНАМИЧЕСКОЕ ПЕРЕРАСПРЕДЕЛЕНИЕ ЗАДАЧ  
В СИСТЕМАХ ВЗАИМОДЕЙСТВУЮЩИХ АГЕНТОВ**

Автореферат  
диссертации на соискание ученой степени  
кандидата технических наук  
по специальности 05.13.01 – Системный анализ, управление  
и обработка информации

---

Подписано в печать 00.10.2016.  
Гарнитура «Таймс».  
Уч.-изд. л. 1,2.

Формат 60x84 1/16.  
Печать ризографическая.  
Тираж 60 экз.

Бумага офсетная.  
Уч.-печ. л. 1,4.  
Заказ 000.

---

Издатель и полиграфическое исполнение: Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»  
Свидетельство о государственной регистрации издателя, изготовителя, распространителя  
печатных изданий № 1/238 от 24.03.2014, № 2/113 от 07.04.2014, № 3/615 от 07.04.2014.  
ЛП № 02330/264 от 14.04.2014  
220013, Минск, ул.П. Бровки, 6