

Сахаров Максим Константинович

**МЕТОДИКА ПРОЕКТИРОВАНИЯ ПРОГРАММ ДЛЯ
РЕШЕНИЯ ЗАДАЧ ГЛОБАЛЬНОЙ ПАРАМЕТРИЧЕСКОЙ
ОПТИМИЗАЦИИ НА СЛАБОСВЯЗАННЫХ
ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ**

Специальность 05.13.11 – Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей
(технические науки)

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата технических наук

Работа выполнена в Федеральном государственном бюджетном образовательном учреждении высшего образования «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана) на кафедре систем автоматизированного проектирования.

Научный руководитель: **Карпенко Анатолий Павлович**
доктор физико-математических наук, профессор,
МГТУ им. Н.Э. Баумана

Официальные оппоненты: **Гагарина Лариса Геннадьевна**
доктор технических наук, профессор,
Федеральное государственное автономное
образовательное учреждение высшего
образования «Национальный исследовательский
университет «Московский институт электронной
техники», директор института СПИНТех

Серов Владимир Александрович
кандидат технических наук, доцент,
Федеральное государственное бюджетное
образовательное учреждение высшего
образования «МИРЭА — Российский
технологический университет», доцент

Ведущая организация: **Федеральный исследовательский центр
«Информатика и управление»
Российской Академии Наук**

Защита диссертации состоится «27» мая 2020 г. в 13 часов 00 минут на заседании диссертационного совета Д 999.216.02 при МАИ и МГТУ им. Н.Э. Баумана по адресу: 105005, г. Москва, 2-я Бауманская ул., д. 5, стр. 1.

С диссертацией можно ознакомиться в библиотеке МГТУ им. Н.Э. Баумана и на сайте www.bmstu.ru.

Ваш отзыв на автореферат в двух экземплярах, заверенный печатью организации, просим направлять по вышеуказанному адресу на имя ученого секретаря диссертационного совета.

Автореферат разослан « » _____ 2020 г.

Ученый секретарь
диссертационного совета Д 999.219.02
д.т.н., доцент

А.Н. Алфимцев

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность работы. Современные практически значимые задачи глобальной параметрической оптимизации (ГПО) характеризуются высокой размерностью пространства поиска, сложной топологией области допустимых значений, а также высокой вычислительной сложностью оптимизируемых функций. В связи с этим актуальной является разработка параллельных методов ГПО для различных типов архитектуры параллельных вычислительных систем.

Среди существующих параллельных вычислительных систем особое место занимают слабосвязанные вычислительные системы (ССВС), которые представляют собой кластеры гетерогенных компьютеров, работающих вместе над решением единой вычислительно сложной задачи. Преимуществом ССВС является возможность получения доступа к мощным вычислительным ресурсам без высоких экономических затрат.

Основные разработки в области организации распределенных вычислений с использованием ССВС проводятся в Новосибирском государственном университете, Объединенном институте ядерных исследований, Калифорнийском университете в Беркли и др. Следует отметить следующие центры, занимающиеся решением задач ГПО на ССВС: ФИЦ информатики и управления РАН, Институт информационных технологий, математики и информатики ННГУ, Институт проблем передачи информации РАН.

Исследования в области разработки параллельных алгоритмов глобальной оптимизации и программного обеспечения для ССВС проводят Ю.Г. Евтушенко, М.А. Посыпкин, А.П. Афанасьев, Р.Г. Стронгин, В.П. Гергель, В.А. Гришагин, К.А. Баркалов, Д.В. Гнатюк, А.А. Лабутина, С.Ю. Городецкий, А.В. Сысоев, J.F. Schutte, R.H. Byrd, N. Krasnogor, Y.S. Ong и P. Moscato. Также следует отметить Вл. В. Воеводина, А.С. Антонова, П.Ф. Трифонова и В.Ф. Звягина, которые занимаются разработкой алгоритмов и программного обеспечения для ССВС.

К трудностям, возникающим при организации вычислений на ССВС, относится необходимость использования специализированного программного обеспечения для распределения задач по вычислительным узлам.

Выполненный в диссертации анализ литературы по различным аспектам решения задач глобальной параметрической оптимизации на ССВС показал следующее.

1. Современные программные комплексы не используют ССВС для решения задач ГПО, в то время как потенциал таких систем и их экономическая рентабельность высоки.

2. Наличие большого числа вычислительно сложных задач ГПО, в частности, в области САПР.

3. Незрелость методов проектирования программ для решения задачи ГПО на ССВС.

4. Ограничения известных методов балансировки загрузки ССВС, обеспечивающих эффективное использование ресурсов для решения задач ГПО.

5. Методы организации взаимодействия программ на узлах ССВС не учитывают особенности задач ГПО и методов их решения.

На этом основании сделано заключение об актуальности разработки параллельных методов, алгоритмов и программного обеспечения ГПО, ориентированных на ССВС.

Целью работы является разработка методики проектирования программ, обеспечивающих эффективное решение задач глобальной параметрической оптимизации на слабосвязанных вычислительных системах.

Задачи исследования.

1. Постановка задачи и обзор методов, алгоритмов и программного обеспечения для решения задач ГПО на ССВС.

2. Разработка методики проектирования программ для эффективного решения задач ГПО на ССВС с учетом архитектуры этих систем.

3. Разработка метода эффективного отображения программ, реализующих методы ГПО, на вычислительные узлы ССВС.

4. Разработка метода организации взаимодействия программ ГПО, выполняемых на узлах ССВС, с учетом особенностей обрабатываемых подобластей пространства поиска.

5. Разработка метода ГПО, ориентированного на ССВС, отличительной особенностью которого является возможность адаптации к обрабатываемым подобластям пространства поиска.

6. Программная реализация предложенных методов и алгоритмов.

7. Исследование эффективности и практическая апробация на ССВС разработанного математического и программного обеспечения в процессе решения задач управления заболеваемостью и термостимулированной люминесценцией полиарилефталидов.

Методы исследования. При разработке методов, алгоритмов и программного обеспечения ГПО использовались методы системного анализа, нелинейного программирования, однокритериальной оптимизации и методы балансировки загрузки параллельных вычислительных систем. В диссертации также использованы численные методы решения систем обыкновенных дифференциальных уравнений, теории вычислительной сложности и теории вероятностей.

Научная новизна результатов исследования.

1. Методика проектирования программ для решения задачи ГПО на ССВС, обеспечивающая их эффективное распараллеливание и отображение на архитектуру этих систем, особенность которой заключается в статической адаптации программ ГПО к решаемой задаче с помощью предварительного анализа ландшафта целевой функции задачи.

2. Метод статической балансировки загрузки узлов ССВС, основанный на формировании субпопуляций используемого популяционного алгоритма ГПО на основе результатов ландшафтного анализа целевой функции.

3. Метод организации взаимодействия программ ГПО, выполняемых на узлах ССВС, основанный на их динамической адаптации к особенностям поведения целевой функции в обрабатываемой подобласти пространства поиска с использованием мульти-меметического подхода.

4. Архитектура программной системы для решения задач ГПО, ориентированной на ССВС и реализующей предложенные методологию, методы и алгоритмы.

5. Результаты исследования эффективности разработанного математического и программного обеспечения с использованием многомерных тестовых функций различных классов, а также в процессе решения на ССВС задач оптимального управления заболеваемостью при импульсной вакцинации и оптимального управления термостимулированной люминесценцией полиарилефталидов.

Теоретическая и практическая значимость результатов работы заключается в следующем.

1. Предложены методика и методы, позволяющие повысить эффективность решения задач ГПО на ССВС:

- методика проектирования программ для решения задачи ГПО на ССВС;
- метод организации взаимодействия программ, выполняемых на узлах ССВС;
- метод статической балансировки загрузки узлов ССВС;
- метод решения задач ГПО, ориентированный на архитектуру ССВС.

2. Разработан программный комплекс, реализующий указанные методологию и методы и позволяющий повысить эффективность решения задач ГПО на ССВС за счет предварительного исследования характеристик целевых функций и балансировки загрузки.

3. Использование предложенной методологии, методов и их программных реализаций позволило:

- определить оптимальный график вакцинации больных, который дает возможность удерживать число инфицированных в пределах заданных ограничений, минимизируя количество используемой вакцины;
- определить оптимальные законы изменения температуры для управления люминесценцией полиарилефталидов.

4. Результаты работы внедрены в МГТУ им. Н.Э. Баумана, Институте нефтехимии и катализа, обособленном структурном подразделении ФГБНУ Уфимского федерального исследовательского центра РАН и ООО «Ладуга».

Достоверность и обоснованность полученных в диссертации научных результатов обеспечивается строгостью используемого математического аппарата и подтверждены широким вычислительным экспериментом.

Апробация работы. Основные результаты и положения диссертационной работы представлены и прошли апробацию на 12 конференциях, включая: международная научная конференции Optimization and applications, OPTIMA (Петровац, Черногория, 2013); международные научные конференции Intelligent Information Technologies for Industry, ИТИ (Сочи, Россия, 2016 и 2018; Варна, Болгария, 2017); международная научная конференция Параллельные вычислительные технологии, ПаВТ (Ростов-на-Дону, Россия, 2018); международная научная конференция Theory and Practice of Natural Computing, TPNC (Дублин, Ирландия, 2018); шестой всемирный конгресс по глобальной оптимизации, WCGO 2019 (Мец, Франция, 2019).

Публикации. Основные результаты диссертации отражены в 23 научных работах, из них 6 работ – в журналах, рекомендованных ВАК РФ и 6 работ – в сборниках, индексируемых в Scopus и Web of Science.

Личный вклад автора. Все исследования, результаты которых изложены в диссертационной работе, проведены лично соискателем в процессе научной деятельности. Из совместных публикаций в диссертацию включен лишь тот материал, который непосредственно принадлежит соискателю; заимствованный материал обозначен в работе ссылками.

Структура и объем диссертации. Диссертация состоит из введения, четырех глав, заключения и списка используемой литературы. Общий объем диссертации составляет 123 страницы машинописного текста. Диссертация содержит 31 рисунок, 11 таблиц. Список используемой литературы состоит из 133 источников.

СОДЕРЖАНИЕ РАБОТЫ

Во **ВВЕДЕНИИ** обоснована актуальность темы, сформулированы цели и задачи исследования, научная новизна, указаны достоверность и обоснованность, а также практическая значимость полученных результатов, перечислены основные положения, выносимые на защиту.

В **ГЛАВЕ 1** рассмотрены особенности ССВС как одного из классов параллельных вычислительных систем. ССВС предлагают возможность получить доступ к мощным вычислительным ресурсам без высоких экономических затрат.

Представлены результаты обзора современных классических и метаэвристических методов решения задач ГПО на ССВС. Выделены особенности практически значимых задач, отличительными чертами которых являются высокая размерность пространства поиска, сложная топология области допустимых значений, а также нелинейность, недифференцируемость, многоэкстремальность, овражность, отсутствие аналитического выражения и высокая вычислительная сложность оптимизируемых функций.

Также в диссертации проведен обзор методов синтеза новых популяционных алгоритмов, ориентированных на ССВС, путем гибридизации двух и более алгоритмов (меметические и мульти-меметические алгоритмы), а также путем мета-оптимизации а) с помощью предварительной настройки свободных параметров алгоритма до начала его работы методом ландшафтного анализа, б) с помощью адаптации и само-адаптации в процессе работы алгоритма.

По результатам анализа существующих подходов к синтезу параллельных методов ГПО, а также методов распараллеливания вычислений, ориентированных на ССВС, сформулированы требования к *базовому алгоритму оптимизации*, которые включают в себя минимальное число обменов информацией между совокупностями индивидов, возможность алгоритма эффективно работать с различными подобластями поиска, отсутствие централизованного управляющего оператора и др.

В **ГЛАВЕ 2** поставлена задача ГПО, предложены методика проектирования программ для эффективного решения задач ГПО на ССВС, метод статической балансировки загрузки узлов ССВС и метод организации взаимодействия программ, выполняемых на узлах ССВС.

Рассматриваем задачу ГПО в виде

$$\min_{X \in D \subset R^n} \Phi(X) = \Phi(X^*) = \Phi^*, \quad (1)$$

где $\Phi(X)$ – скалярная целевая функция; $\Phi(X^*) = \Phi^*$ – ее искомое минимальное значение; $X = (x_1, x_2, \dots, x_n)$ – n -мерный вектор варьируемых параметров; D – область допустимых значений вектора варьируемых параметров X ; R^n – n -мерное арифметическое пространство. Задана область $D_0 = \{X | x^{min} \leq x_i \leq x^{max}, i \in [1: n]\}$ для генерации начальной популяции решений. Называем набор $Q = \langle \Phi, D \rangle$ базовой задачей.

Рассматриваем модель вычислительной сети ССВС, состоящую из одного главного узла *host* P_0 и нескольких подчиненных гетерогенных узлов $P_1 \dots P_N$, которые узлы логически связаны только с главным узлом независимо от физической топологии сети. В качестве модели распараллеливания используем модель некомутирующих субпопуляций, где подчинённые узлы не обмениваются информацией и работают независимо друг от друга. Управление вычислительным процессом осуществляется через главный узел P_0 .

Методика проектирования программ для решения задачи ГПО на ССВС. В диссертации предложена методика проектирования программ для решения задачи ГПО на основе популяционных алгоритмов оптимизации с учетом архитектуры ССВС. Методика позволяет использовать в качестве *базового* любой популяционный алгоритм, удовлетворяющий ограничениям, наложенным ССВС: отсутствие обменов информацией между вычислительными узлами, возможная потеря данных, независимая эволюция решений на узлах и др.

Методика включает этап настройки программной системы (ПС) и этап решения базовой задачи. Этап настройки выполняется однократно для каждого из базовых алгоритмов, этап решения – для каждой из базовых задач ГПО.

I. Этап настройки программной системы заключается в настройке базового алгоритма с использованием методов ландшафтного анализа и мета-оптимизации. Настройка выполняется на основе набора Q многомерных тестовых задач ГПО и включает в себя: Ia) формирование набора тестовых задач; Ib) ландшафтный анализ; Iv) классификацию тестовых задач; Ig) мета-оптимизацию базового алгоритма.

Ia. *Формирование набора тестовых задач.* Множество тестовых задач Q формируем на основе известных наборов тестовых функций Φ различных классов и соответствующих ОДЗ D .

Iб. *Ландшафтный анализ.* Для каждой тестовой задачи выполняется предложенная в диссертации процедура ландшафтного анализа.

- В области D формируем обучающую выборку: случайным образом распределяем векторы $X_j, j \in [1: K]$. Вычисляем соответствующие значения

целевой функции Φ_j на узлах ССВС с использованием модели распараллеливания «хозяин-слуга».

- На главном узле P_0 сортируем векторы X_j по возрастанию значения Φ_j и равномерно разделяем на N наборов по числу доступных вычислительных узлов. Отождествляем данные наборы с некоммутирующими субпопуляциями $S_l, l \in [1: N]$.

- Определяем диаметры (Рис. 1) этих субпопуляций (характерные признаки задачи) по формуле

$$d_l = \max_{X_q, X_r \in S_l} \left(\sqrt{\|X_q - X_r\|^2} \right), \quad q, r \in \left[1: \frac{K}{N} \right], \quad l \in [1: N]. \quad (2)$$

Ив. *Классификация тестовых задач.* Выполняем классификацию тестовых задач по следующей схеме.

- Строим линейную аппроксимацию дискретной функции $d(l)$, например, методом наименьших квадратов. Функция $d(l)$ может возрастать, убывать или не возрастать и не убывать (с некоторой погрешностью).

- Определяем значение *сепарирующего параметра* $\xi = d_D/d_1$, где d_D – диаметр области D . Рекомендованное в диссертации пороговое значение $\tilde{\xi} = 2,5$. Если $\xi \geq \tilde{\xi}$, то полагаем, что первая субпопуляция S_1 имеет «малый» диаметр по сравнению с областью D .

- На основе рассчитанного значения параметра ξ и поведения функции $d(l)$ выполняем полный перебор их возможных комбинаций и выделяем шесть классов тестовых задач $\{Q\}_i, i \in [1: 6]$, которые характеризуются различной топологией ландшафта целевой функции.

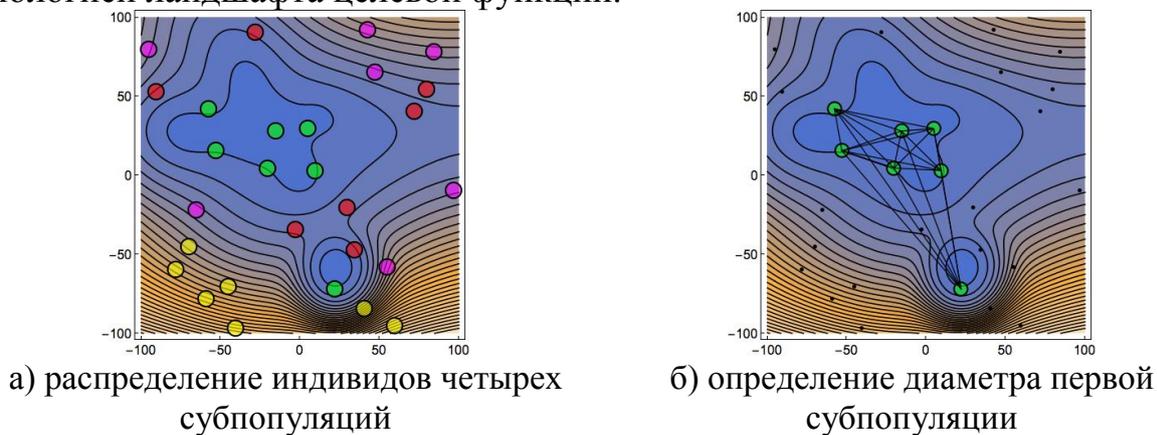


Рис. 1. Пример определения диаметра субпопуляции:
составная тестовая функция №1 из набора СЕС'14

Иг. *Мета-оптимизация базового алгоритма.* Для каждого класса $\{Q\}_i, i \in [1: 6]$ формулируем две следующие задачи мета-оптимизации. Для базового алгоритма $A(B)$ и его программной реализации $\Pi_{\text{БА}}$ необходимо найти значения свободных параметров B , обеспечивающие 1) интенсификацию поиска (малое разнообразие решений) и 2) диверсификацию (большое разнообразие):

$$1) \min_{B \in D_B} W(Q, B) = W(Q, B^*); \quad 2) \max_{B \in D_B} W(Q, B) = W(Q, B^{**}). \quad (3)$$

Здесь W – критерий оптимальности; D_B – область допустимых значений свободных параметров. В качестве критерия оптимальности W используем число кластеров, полученное при кластеризации множества кандидатов в решение $\{\tilde{X}\}$. Приближенное решение задач (3) для фиксированного класса тестовых задач $\{Q\}_i, i \in [1:6]$ получаем по следующей схеме.

- Покрываем заданную область допустимых значений свободных параметров D_B сеткой $\{B_h\}$.

- Для каждой тестовой задачи $Q_k \in \{Q\}_i$ и для всех узлов сетки $\{B_h\}$ решаем базовую задачу (1) методом мульти-старта (m запусков). Формируем множество всех кандидатов в решение $\{\tilde{X}\}_{h,k,m}$.

- Решаем задачу кластеризации для множества $\{\tilde{X}\}_{h,k,m}$ и определяем число кластеров $W_m(Q_k, B_h)$.

- Усредняем полученные значения $W_m(Q_k, B_h)$ по мульти-старту; получаем $\bar{W}(Q_k, B_h)$.

- Усредняем полученные значения $\bar{W}(Q_k, B_h)$ по тестовым задачам; получаем $\bar{\bar{W}}(B_h)$.

- Находим минимальное $W_{min}(B^*)$ и максимальное $W_{max}(B^{**})$ число кластеров и соответствующие им узлы B^* и B^{**} сетки $\{B_h\}$.

Повторяем процедуру для каждого класса тестовых задач $\{Q\}_i, i \in [1:6]$ и получаем рекомендованные значения свободных параметров B_i^* и $B_i^{**}, i \in [1:6]$.

Для выбранного базового алгоритма $A(B)$ каждому свободному параметру $b_u, u \in [1:U]$ ставим в соответствие функцию $b_u(l)$, которая позволяет, изменяя значения параметра, управлять интенсификационными и диверсификационными свойствами базового алгоритма $A(B)$.

II. Этап решения базовой задачи включает: Па) ландшафтный анализ базовой задачи ГПО; Пб) классификацию базовой задачи; Пв) статическую балансировку загрузки ССВС; Пг) статическую адаптацию программ ГПО; Пд) решение базовой задачи.

Па. *Процедура ландшафтного анализа.* Выполняем процедуру ландшафтного анализа базовой задачи аналогично п. Ia. Получаем множество субпопуляций $S_l, l \in [1:N]$.

Пб. *Классификация базовой задачи.* По виду функции $d(l)$ и значению параметра ξ , вычисляемого аналогично п. Ib, относим базовую задачу Q к одному из шести классов $\{Q\}_i, i \in [1:6]$ и определяем правила выбора значений свободных параметров для каждой из субпопуляций S_l . В Главе 3 представлены эти правила для используемого в диссертации базового алгоритма.

Пв. *Статическая балансировка загрузки ССВС.* Для эффективного использования ресурсов ССВС необходимо решить задачу загрузки вычислительных узлов ССВС. В диссертации предложен оригинальный метод статической балансировки загрузки на основе данных, полученных в Па (см. ниже).

Пг. *Статическая адаптация программ ГПО.* В соответствии с результатами классификации базовой задачи (задача отнесена к одному из классов $\{Q\}_i, i \in [1:6]$) выполняем статическую адаптацию программ ГПО для

узлов ССВС к особенностям различных подобластей целевой функции. Для каждого свободного параметра $b_u, u \in [1:U]$ базового алгоритма $A(B)$ используем выбранный на этапе настройки вид функций $b_u(l)$: линейный, квадратичный, случайный и др. Определяем значение свободного параметра $\hat{b}_u(l) \in [b_u^*: b_u^{**}]$ для каждой субпопуляции S_l . Выполняем настройку программ ГПО $\Pi_{\text{БА}}(\{\hat{b}_1(l), \dots, \hat{b}_U(l)\}) = \Pi_{\text{БА}}(\hat{B}_l)$ для каждой субпопуляции $S_l, l \in [1: N]$:

$$\begin{cases} S_1 \rightarrow A(\hat{B}_1) \rightarrow \Pi_{\text{БА}}(\hat{B}_1), \\ \dots \\ S_N \rightarrow A(\hat{B}_N) \rightarrow \Pi_{\text{БА}}(\hat{B}_N). \end{cases}$$

Пд. Решение базовой задачи. Полученные субпопуляции $S_l, l \in [1: N]$ (п. Па) и соответствующие им настроенные программы $\Pi_{\text{БА}}(\hat{B}_l)$ распределяем по вычислительным узлам ССВС. Независимая эволюция субпопуляций может осуществляться как в синхронном, так и в асинхронном режиме. В диссертации используется синхронный режим, когда главный узел либо ожидает пока все узлы вернут свои решения, либо прекращает расчет по тайм-ауту. В этом случае каждый узел ССВС возвращает текущее лучшее решение задачи (1).

Метод статической балансировки загрузки узлов ССВС. Распараллеливание осуществляется с использованием модели некоммутирующих субпопуляций так, что на узле P_0 выполняются процедуры инициализации, ландшафтного анализа, статической адаптации и формирование итогового решения, а на каждом из подчиненных узлов P_l происходит независимая эволюция субпопуляций S_l , которые динамически адаптируются к решаемой задаче путем гибридизации с методами локального поиска.

Поскольку вычислительная сложность целевых функций в практически значимых задачах зачастую неравномерна в области поиска, а также поскольку производительность вычислительных узлов в гетерогенных ССВС различна, важной задачей является задача балансировки загрузки. Предложенная в диссертации схема балансировки загрузки имеет следующий вид.

1. Субпопуляции $S_l, l \in [1: N]$ исследуем на наличие статистических выбросов в оценках времени $t_{l,q}, q \in [1: K/N]$ вычисления целевой функции Φ , определенных на этапе ЛА. Найденные выбросы исключаем из рассмотрения.

2. Вычислительные узлы P_l ранжируем по производительности, начиная с самого мощного: $p_1 \geq p_2 \geq \dots \geq p_N$.

3. На первый узел P_1 отправляем первую субпопуляцию $S_1: S_1 \rightarrow P_1$.

4. Распределяем оставшиеся субпопуляции по узлам $P_l: S_l \rightarrow P_l, l \in [2: N]$ и пересчитываем оценки $t_{l,q}$ с учетом производительности каждого из узлов $p_2 \dots p_N$.

5. Для каждой смежной пары субпопуляций S_l, S_{l+1} путем перестановки индивидов формируем новые субпопуляции S'_l, S'_{l+1} с близкими значениями средних по субпопуляциям времен \bar{T}_l, \bar{T}_{l+1} вычисления значений целевой функции Φ (Рис. 2).

6. Новые субпопуляции распределяем по узлам $P_l: S'_l \rightarrow P_l, l \in [2: N]$.

Процедура балансировки загрузки формирует субпопуляции близкой вычислительной сложности для распределения по доступным вычислительным узлам.

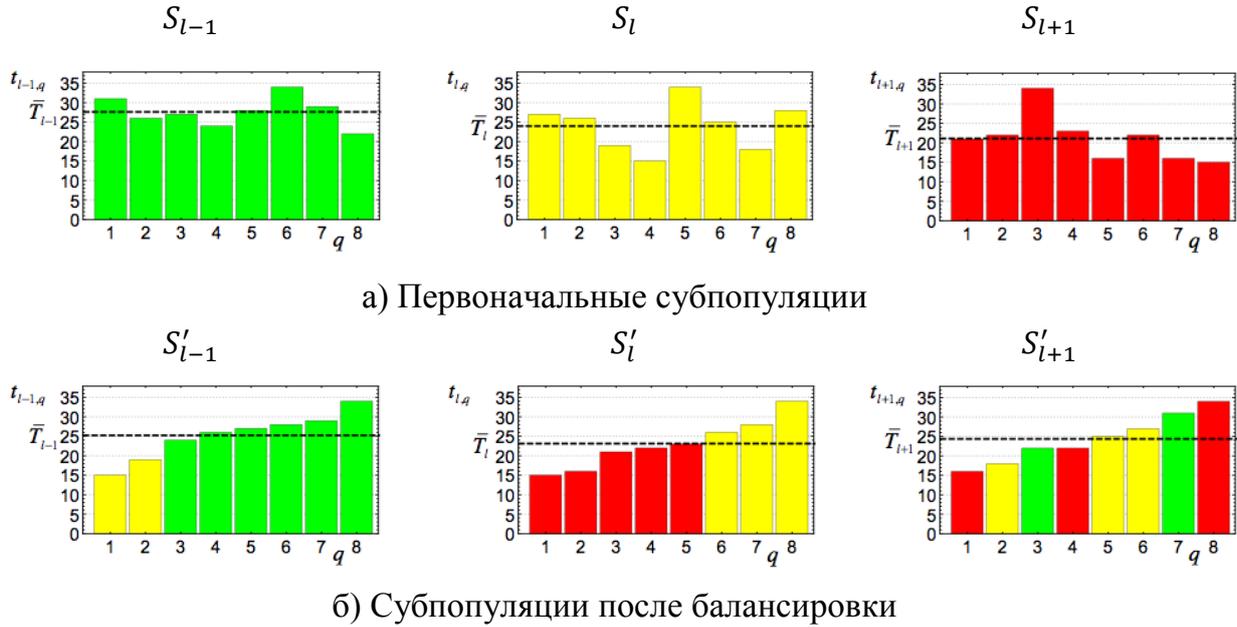


Рис. 2. Перераспределение загрузки в смежных субпопуляциях

Метод организации взаимодействия программ, выполняемых на узлах ССВС. Полученные на этапе ландшафтного анализа субпопуляции $S_l, l \in [1: N]$ покрывают, многосвязные области поиска с разными свойствами целевой функции Φ . Для повышения эффективности решения базовой задачи в условиях отсутствия обменов информации между субпопуляциями, в диссертации предложено использовать динамическую адаптацию поиска к исследуемой подобласти на основе мульти-меметического подхода.

На каждом подчиненном узле P_k выполняются: программа $\Pi_{\text{БА}}$ базового алгоритма; программа выбора мемов $\Pi_{\text{ВМ}}$; множество программ $\Pi_{m_e}, e \in [1: M]$ локальной оптимизации (мемов). Для организации взаимодействия этих программ на каждом узле $P_k, k \in [1: N]$ предложен следующий метод.

1. Программная реализация базового алгоритма $\Pi_{\text{БА}}(\hat{B}_k)$ на узле P_k обращается к программе $\Pi_{\text{ВМ}}$ выбора мемов каждые $\gamma_{\text{ВМ}}$ итераций ($\gamma_{\text{ВМ}}$ – свободный параметр метода).

2. Программа $\Pi_{\text{ВМ}}$ выбирает наиболее подходящий мем $m_o, o \in [1: M]$ с помощью предложенной в диссертации функции выбора, которая представляет собой некоторую скалярную свертку показателей качества мема ϕ_1, \dots, ϕ_z :

$$\max_{e \in [1: M]} F(m_e) = \max_{e \in [1: M]} F(\phi_1(m_e), \dots, \phi_z(m_e)) = F(\phi_1(m_o), \dots, \phi_z(m_o)).$$

3. Программа выбранного мема $\Pi_{m_o}^*$ выполняется в течение $\gamma_{\text{ЛП}}$ итераций для уточнения всех кандидатов в решение $\{X\}$ субпопуляции S_k ($\gamma_{\text{ЛП}}$ – свободный параметр метода).

В качестве скалярной свертки предложено использовать гиперэвристику на основе трех показателей качества:

$$F(m_e) = \lambda_1 \phi_1(m_e) + \lambda_2 \phi_2(m_e) + \lambda_3 \phi_3(m_e), \quad e \in [1: |M|]. \quad (4)$$

Функция (4) представляет собой аддитивную свертку функций $\phi_1(m_e)$, $\phi_2(m_e)$, $\phi_3(m_e)$ с весами $\lambda_1, \lambda_2, \lambda_3$. Полагаем, что лучшему мему соответствует большее значение функции $\phi(m_e)$. Компонента $\phi_1(m_e)$ формализует улучшение значения целевой функции в текущей точке, достигнутое мемом m_e . Функция $\phi_2(m_e)$ реализует случайный выбор того или иного мема. Компонента $\phi_3(m_e)$ предназначена для включения экспертных знаний в пул мемов.

В ГЛАВЕ 3 представлена программная система *Parallel Memetic Global Optimization Solver (PMGOS)*, реализующая предложенные методика, методы и алгоритмы. Архитектура системы представлена на Рис. 3.

В качестве базового алгоритма выбран простой алгоритм эволюции разума *SMEC*, который построен на основе операций инициализации групп, локальных состязаний (*similar-taxis*) и диссимилиации (*dissimilation*).

Этап настройки системы PMGOS для алгоритма SMEC. Выполнена настройка программной системы *PMGOS* в соответствии со схемой, представленной в Главе 2. Для каждой из тестовых задач Q (СЕС'14) выполнены процедуры ландшафтного анализа и классификации тестовых задач. Мета-оптимизация алгоритма *SMEC* осуществлена с использованием его трех свободных параметров: $b_1 = \sigma$ – среднее квадратичное отклонение при генерации индивидов; $b_2 = \omega$ – частота удаления отстающих групп; $b_3 = \eta$ – отношение числа лидирующих и отстающих групп. По результатам процедуры мета-оптимизации предложены правила выбора значений указанных свободных параметров $b_i, i \in [1: 3]$ (Таблица 1).

Таблица 1.
Правила выбора значений свободных параметров $b_i, i \in [1: 3]$
для базового алгоритма *SMEC*

Соотношение $\frac{d_D}{d_1}$	$d(l)$ возрастает	$d(l)$ не возрастает, не убывает	$d(l)$ убывает
$\frac{d_D}{d_1} > \zeta$	Класс Q_1 : $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 20; \eta_l = 50$; $a_l = 0,25; b_l = 0,75$.	Класс Q_3 : $\sigma_l = rand(a_l, b_l)$; $\omega_l = 20; \eta_l = 50$; $a_l = 0,1; b_l = 0,9$.	Класс Q_5 : $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 20; \eta_l = 50$; $a_l = 1; b_l = -0,75$.
$\frac{d_D}{d_1} \leq \zeta$	Класс Q_2 : $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 25; \eta_l = 75$; $a_l = 0,25; b_l = 0,75$.	Класс Q_4 : $\sigma_l = rand(a_l, b_l)$; $\omega_l = 25; \eta_l = 75$; $a_l = 0,1; b_l = 0,9$.	Класс Q_6 : $\sigma_l = a_l + b_l \frac{l-1}{k-1}$; $\omega_l = 25; \eta_l = 75$; $a_l = 1; b_l = -0,75$.

Формирование пула мемов $m_e, e \in [1: M]$ для системы *PMGOS* осуществляется пользователем. В диссертации использованы мемы нулевого порядка ($M = 4$): алгоритм Нелдера-Мида; алгоритм Хука-Дживса; алгоритм случайного поиска по сфере; алгоритм Солиса-Ветса.

Отметим, что результат настройки программной системы *PMGOS* для простого алгоритма эволюции разума *SMEC* порождает новый метод решения задач ГПО, получивший в диссертации название *Parallel Modified Multi-Memetic Mind Evolutionary Computation (M3MEC-P)*.

Программная система *PMGOS* реализована на языке программирования *Wolfram Language*. Выбор среды разработки *Wolfram Mathematica* обусловлен лаконичностью языка программирования *Wolfram Language* за счет использования функциональной парадигмы программирования, а также наличием встроенного инструментария для работы с ССВС. Все вычислительные эксперименты проводились на ССВС, сформированной из девяти персональных компьютеров (4 CPU, 4 GB RAM, OS Windows 10).

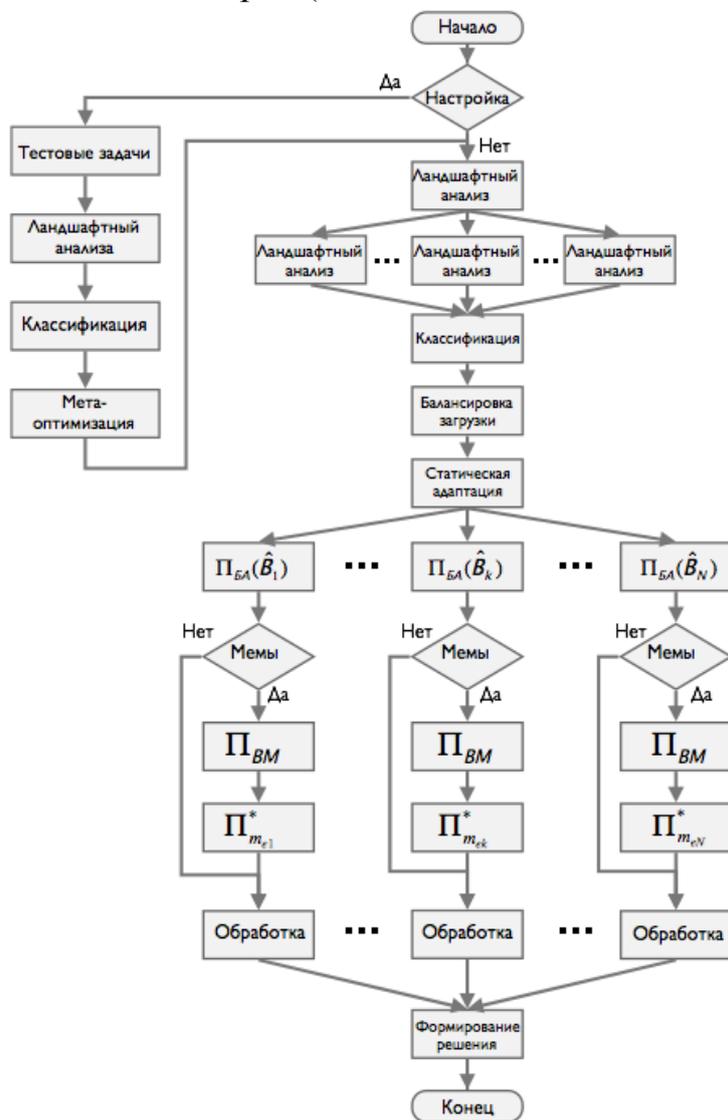


Рис. 3. Архитектура программной системы *PMGOS*

Исследование эффективности программной системы *PMGOS*.

Исследование выполнено с использованием многомерных ($n = 8, 16, 32$) тестовых задач Q , решаемых в соответствии с этапом II (Глава 2).

Вычислительные эксперименты проводились с использованием метода мултистарта (100 запусков). В качестве индикаторов эффективности использованы: среднее лучшее достигнутое значение целевой функции по итогам мултистарта $\bar{\Phi}$; минимальное найденное значение целевой функции $\tilde{\Phi}^*$; оценка вероятности $p\%$ локализации глобального минимума с заданной точностью; среднее число итераций $\bar{\gamma}$ алгоритма.

На Рис. 4 представлены некоторые результаты исследования эффективности системы *PMGOS*. Они демонстрируют высокую оценку вероятности локализации глобального минимума многомерных тестовых функций. Для размерности $n = 16$ настроенная программная система с базовым алгоритмом *МЗМЕС-Р* смогла локализовать глобальный оптимум с вероятностью $p = 100\%$ для большинства тестовых задач; оценка вероятности p при $n = 32$ для самых сложных тестовых функций оказалась не ниже 80% .

С использованием разработанного программного обеспечения в диссертации выполнено сравнительное исследование эффективности алгоритма *SMEC*, алгоритма оптимизации роем частиц *PSO* и программной системы *PMGOS*. Исследование проводилось на наборе многомерных ($n = 8, 16$) тестовых задач. Результаты вычислительных экспериментов показывают превосходство системы *PMGOS* над программными реализациями алгоритмов *SMEC* и *PSO*. Так, если для тестовых задач размерности $n = 8$ алгоритмы *SMEC* и *PSO* продемонстрировали высокую вероятность локализации глобального минимума, то для размерности $n = 16$ оба алгоритма не смогли локализовать глобальный минимум функции *Rastrigin* и показали низкую вероятность локализации глобального минимума функций *Rosenbrock*, *Zakharov* и *Salomon*. В то же время, система *PMGOS* в этих случаях обеспечил оценку вероятности локализации глобального минимума не меньше $p = 90\%$.

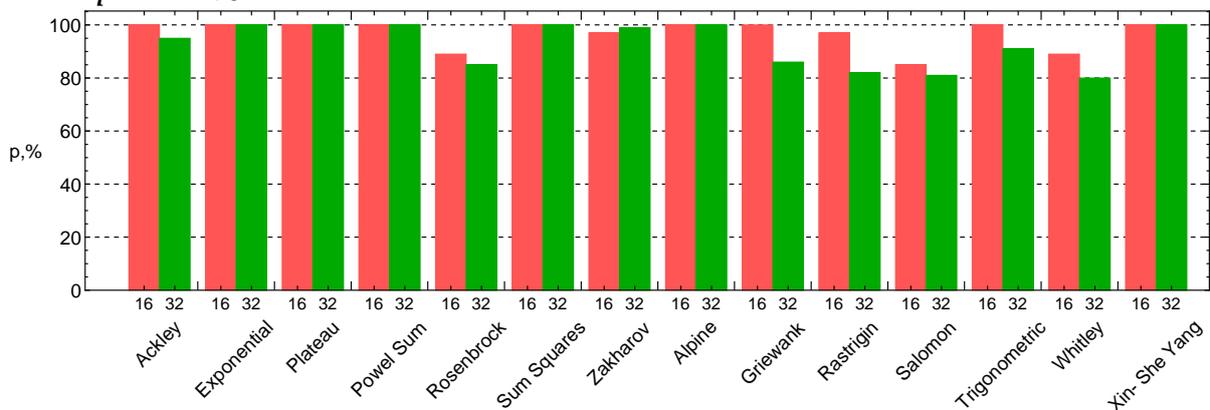


Рис. 4. Оценка вероятности $p\%$ локализации глобального минимума по итогам мультистарта для системы *PMGOS*

На Рис. 5 представлены результаты вычислительных экспериментов по исследованию эффективности программной системы *PMGOS* в зависимости от числа узлов ССВС – оценка ускорения $S(N)$. Вычислительная сложность тестовых задач была искусственно завышена с помощью заикливания. Разница между полученным ускорением и теоретически возможным обусловлена относительно невысокой вычислительной сложностью тестовых задач и особенностями процедуры ландшафтного анализа (Па), в рамках которой происходит обмен данными между главным и починенными узлами.

ГЛАВА 4 посвящена решению двух практически значимых задач ГПО с использованием разработанной программной системы *PMGOS*.

Контроль заболеваемости с использованием импульсной вакцинации. В настоящее время математические модели динамики заболеваемости

используют для планирования защитных противоэпидемических процедур, к числу которых относится вакцинация. Используем известную модель *SEIR* динамики инфекционных заболеваний в виде системы нежестких обыкновенных дифференциальных уравнений в форме Коши:

$$\begin{cases} S'(t) = bN(t) - \beta S(t)I(t) - \mu S(t) - u(t)S(t) + \alpha_S - \theta_S, & S(0) = 900, \\ E'(t) = \beta S(t)I(t) - \sigma E(t) - \mu E(t), & E(0) = 0, \\ I'(t) = \sigma E(t) - \mu I(t) - \gamma I(t) - \mu_1 I(t), & I(0) = 100, \\ R'(t) = \gamma I(t) - \mu R(t) + u(t)S(t), & R(0) = 0; \\ N(t) = S(t) + E(t) + I(t) + R(t); & t = [0; 40]. \end{cases} \quad (3)$$

Здесь $S(t)$ - численность группы индивидов, восприимчивых к инфекции; $E(t)$ - численность группы бессимптомных индивидов; $I(t)$ - численность группы зараженных индивидов; $R(t)$ - численность группы индивидов, приобретших иммунитет к данному заболеванию; $u(t)$ - уровень вакцинации (управление). Используем значения параметров модели для заболевания корь.

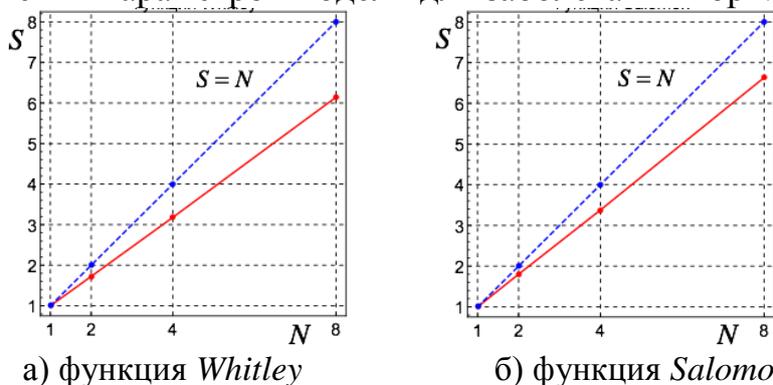


Рис. 5. Оценка ускорения программной системы *PMGOS*

Рассматриваем задачу оптимального управления вакцинацией для системы (3) с критерием качества

$$J(u(t)) = J_1(u(t)) + J_2(u(t)) = \lambda_1 \int_0^{40} u(t) dt + \lambda_2 \min_{t=[0;40]} |I_{ref} - I(u(t))| \rightarrow \min_{u(t) \in [0;1]}$$

где $\lambda_1 = 10$ и $\lambda_2 = -1$. Первая компонента критерия $J(u(t))$ формализует объем используемой вакцины, а вторая компонента - отклонение максимального числа заболевших индивидов от верхней допустимой нормы $I_{ref} = 110$.

Исходная задача оптимального управления в диссертации сведена к задаче глобальной минимизации следующим образом. Интервал интегрирования $t = [0; 40]$ покрываем сеткой так, что длина отрезка $[t_i; t_{i+1}]$ соответствует минимальной продолжительности вакцинации и составляет одну условную единицу времени. Значения управления $u(t_i)$, $i \in [0; 39]$ являются компонентами вектора варьируемых параметров; размерность задачи $n = 40$. Функцию $u(t)$ аппроксимируем кусочно-постоянной функцией. Компоненты вектора u имеют ограничения $0 \leq u(t_i) \leq 1$. Учет ограничений осуществляем с использованием метода «смертельных» штрафов. При каждом испытании выполняем численное решение системы (3) с использованием алгоритма Рунге-Кутты 4-го порядка. По результатам процедур ландшафтного анализа и классификации (Глава 2) полученная базовая задача отнесена к классу $\{Q\}_1$. Для

повышения вероятности локализации оптимального управления использовался метод мульти-старта (10 запусков).

Найденные управления были аппроксимированы с использованием семейства двухимпульсных функций. Один из видов полученного оптимального управления представлен на Рис. 6. Доза использованной вакцины при данном управлении составила $\int_0^{40} u(t) dt = 2,7$ единицы, а максимальное число заболевших $I_{max} = 105$ индивидов.

Оптимальное управление химическими реакциями. В настоящее время органические полимерные материалы находят широкое применение в оптоэлектронике. Одной из разновидностей полимеров, широко используемых в промышленности, являются полиарилефталиды (ПАФ), которые отличаются высокими пленкообразующими свойствами, а также обладают люминесценцией.

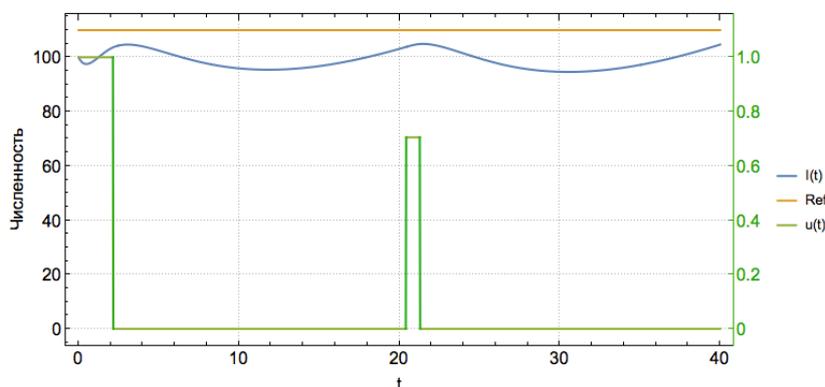


Рис. 6. Оптимальное управление *SEIR*-моделью с использованием двухимпульсного режима вакцинации

В рамках диссертации исследована математическая модель кинетики термостимулированной люминесценции ПАФ в виде задачи Коши для системы обыкновенных дифференциальных уравнений

$$\begin{cases} y_1'(t) = -374 \exp\left(\frac{-69\,944}{10^{-10} + 8,31T(t)}\right) y_1(t)^2, \\ y_2'(t) = -396\,680 \exp\left(\frac{-101\,630}{10^{-10} + 8,31T(t)}\right) y_2(t)^2 + 1,99 \cdot 10^8 \exp\left(\frac{-21\,610}{10^{-10} + 8,31T(t)}\right) y_3(t), \\ y_3'(t) = 187 \exp\left(\frac{-69944}{10^{-10} + 8,31T(t)}\right) y_1(t)^2 + 198\,340 \exp\left(\frac{-101\,630}{10^{-10} + 8,31T(t)}\right) y_2(t)^2 - \\ \quad - 21\,010 y_3(t) - 9,98 \cdot 10^7 \exp\left(\frac{-21\,610}{10^{-10} + 8,31T(t)}\right) y_3(t), \\ y_4'(t) = 21\,010 y_3(t), \end{cases} \quad (4)$$

которая разработана в ИНК РАН (г. Уфа). Здесь y_1, y_2 – концентрации входных веществ, участвующих в реакции, а y_3, y_4 – концентрации выходных веществ.

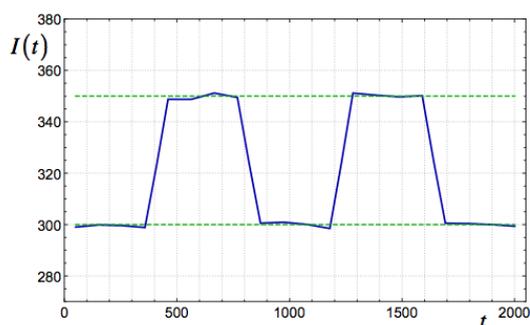
Значение интенсивности люминесценции рассчитывается по формуле $I(t) = 544\,663\,240 y_3(t)$. Начальные концентрации веществ, участвующих в реакции, равны: $y_1(0) = 300$; $y_2(0) = 1000$; $y_3(0) = y_4(0) = 0$. Интервал интегрирования равен $[0; 2000]$ с. Ограничения на минимальную и максимальную температуры $373\text{K} \leq T(t) \leq 473\text{K}$ обусловлены возможностями экспериментальной установки для проведения реакции. Учет ограничений также осуществлялся с помощью метода «смертельных» штрафов.

Задача оптимального управления для системы (3) заключается в минимизации критерия

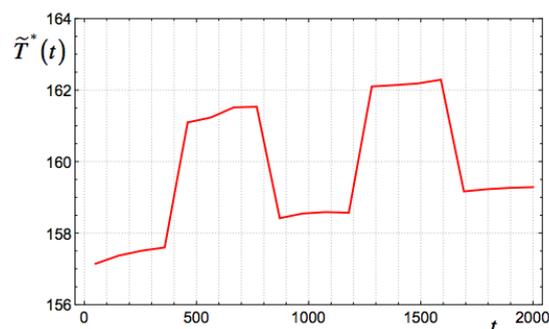
$$J(T(t)) = \int_0^{2000} |I_{ref}(t) - I(u(t))|^2 dt \rightarrow \min_{T(t) \in [373; 473]}$$

на решениях системы ОДУ (4). Поставленная задача оптимального управления сведена к задаче нелинейного программирования аналогично предыдущей задаче. Размерность вектора варьируемых параметров определяется из условий максимально возможной скорости изменения температуры реакции и равна $n = 201$. Систему ОДУ (4) интегрируем с использованием формул дифференцирования назад, поскольку эта система является жесткой. По результатам процедур ландшафтного анализа и классификации полученная базовая задача отнесена к классу $\{Q\}_2$.

В качестве требуемых законов изменения интенсивности люминесценции $I_{ref}(t)$ были заданы кусочно-постоянные функции с фиксированным числом пиков заданной амплитуды (Рис. 7). Также проведены вычислительные эксперименты по управлению скоростью изменения интенсивности люминесценции.



а) Закон изменения светимости $I(t)$ с двумя импульсами



б) Оптимальное управление $\tilde{T}^*(t)$, °С

Рис. 7. Вариант оптимального управления интенсивностью люминесценции ПАФ для обеспечения импульсного уровня светимости

Основные результаты работы

1. Представлены особенности ССВС как одного из классов параллельных вычислительных систем. Выполнен аналитический обзор современных параллельных методов и алгоритмов решения задач глобальной оптимизации на ССВС. Представлены основные направления развития этих методов и алгоритмов. Обоснована необходимость разработки специализированных параллельных методов и алгоритмов, ориентированных на ССВС.

2. Предложена методика проектирования программ для решения задач ГПО на ССВС, обеспечивающая их эффективное распараллеливание и отображение на архитектуру этих систем.

3. Предложен метод организации взаимодействия программ, выполняемых на узлах ССВС, с использованием процедур ландшафтного анализа, настройки его свободных параметров и мульти-меметической гибридизации.

4. Предложен метод статической балансировки загрузки узлов ССВС, основанный на формировании субпопуляций используемого популяционного алгоритма ГПО, который обеспечивает их примерно одинаковую вычислительную сложность.

5. Разработана параллельная программная система *PMGOS*, реализующая предложенные методики, методы и алгоритмы, которая осуществляет управление вычислительными узлами ССВС.

6. Проведено широкое сравнительное исследование эффективности программной системы *PMGOS* с использованием многомерных тестовых задач различных классов. Выполнено сравнение эффективности программной системы с каноническими алгоритмами эволюции разума *SMEC* и роя частиц *PSO*. Проведено исследование ускорения системы *PMGOS* с использованием многомерных тестовых задач, имеющих высокую вычислительную сложность.

7. Разработанная программная система *PMGOS* подтвердила свою эффективность при решении двух практически значимых вычислительно сложных задач глобальной оптимизации: определение оптимального графика импульсной вакцинации при моделировании динамики заболеваемости; определение кинетики термостимулированной люминесценции полиариленфталидов.

Основные публикации по теме диссертации

1. Сахаров М.К. Новый адаптивный метод мультимеметической глобальной оптимизации для слабосвязанных вычислительных систем // Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение, 2019, No.5, С. 95–114. DOI: 10.18698/0236-3933-2019-5-95-114 (1 п.л.)
2. Карпенко А.П., Сахаров М.К. Мультимеметическая глобальная оптимизация на основе алгоритма эволюции разума // Информационные технологии. выпуск 7, 2014,. С. 23-30 (0,5 п.л. / 0,4 п.л.)
3. Сахаров М.К., Карпенко А.П. Меметические алгоритмы для решения задач глобальной нелинейной оптимизации. Обзор // *technomag.edu.ru*: Наука и Образование: электронное научно-техническое издание. 2015. выпуск 12. DOI 10.7463/1215.0829099 (дата обращения 11.11.18) (1 п.л. / 0,8 п.л.)
4. Сахаров М.К., Карпенко А.П., Иваньков И.Ф. Исследование эффективности алгоритма эволюции разума в задаче глобальной оптимизации // Интернет-журнал Науковедение, Том 8, №5 (2016) DOI: 10.15862/51tvn516 (дата обращения 01.10.19) (0,7 п.л. / 0,6 п.л.)
5. Сахаров М.К., Поноренко А.В. Исследование эффективности мультимеметического алгоритма эволюции разума. Математика и математическое моделирование. Том 6, 2017. С. 70-82. DOI:10.24108/mathm.0617.0000090 (дата обращения 11.11.18) (0,8 п.л. / 0,7 п.л.)
6. Karpenko A.P., Sakharov M.K. New Adaptive Multi-Memetic Global Optimization Algorithm // *Herald of the Bauman Moscow State Technical University, Series Natural Science*, 2019, No.2, P.17-31. DOI: 10.18698/1812-3368-2019-2-17-31 (0,9 п.л. / 0,7 п.л.)

7. Sakharov M., Karpenko A. Performance Investigation of Mind Evolutionary Computation Algorithm and Some of Its Modifications // *Advances in Intelligent Systems and Computing*, Vol. 450. Springer, Cham, 2018. P. 475 - 486 DOI 10.1007/978-3-319-33609-1_43 (0,7 п.л. / 0,6 п.л.)
8. Sakharov M., Karpenko A. A New Way of Decomposing Search Domain in a Global Optimization Problem // *Advances in Intelligent Systems and Computing*, Vol. 679. Springer, Cham, 2018. P. 398 - 407 DOI 10.1007/978-3-319-68321-8_4 (0,6 п.л. / 0,5 п.л.)
9. Sakharov M., Karpenko A. Multi-memetic Mind Evolutionary Computation Algorithm Based on the Landscape Analysis // *Lecture Notes in Computer Science*, Vol. 11324. Springer, Cham, 2018. P. 238 – 249 DOI 10.1007/978-3-030-04070-3_19 (0,7 п.л. / 0,6 п.л.)
10. Sakharov M., Karpenko A. Meta-optimization of Mind Evolutionary Computation Algorithm Using Design of Experiments // *Advances in Intelligent Systems and Computing*, Vol.874. - Springer, Cham, 2019. P. 473 – 482 DOI 10.1007/978-3-030-01818-4_47 (0,6 п.л. / 0,5 п.л.)
11. Sakharov M. K., Karpenko A. P. Adaptive Load Balancing in the Modified Mind Evolutionary Computation Algorithm // *Supercomputing Frontiers and Innovations*, Vol.5, No.4, P.5-14, 2018. ISSN 2313-8734. 2019. DOI: 10.14529/jsfi180401 (0,6 п.л. / 0,5 п.л.)
12. Sakharov M., Karpenko A. Parallel Multi-memetic Global Optimization Algorithm for Optimal Control of Polyarylenephthalide's Thermally-Stimulated Luminescence // *Advances in Intelligent Systems and Computing*, Vol.991. Springer, Cham, pp. 191-201 (2020) DOI: 10.1007/978-3-030-21803-4_20 (0,7 п.л. / 0,6 п.л.)