

## КОНЦЕПТУАЛЬНОЕ И ФУНКЦИОНАЛЬНОЕ ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ ИНТЕЛЛЕКТУАЛЬНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ РЕШЕНИЯ ЗАДАЧ МАТЕМАТИЧЕСКОЙ ОПТИМИЗАЦИИ

Д. С. Соловьев, И. А. Соловьева, А. В. Самохвалов

*Тамбовский государственный университет имени Г. Р. Державина,  
Тамбов, Россия, [solovjevdenis@mail.ru](mailto:solovjevdenis@mail.ru), [good.win32@yandex.ru](mailto:good.win32@yandex.ru),  
[samohvalov@gmail.com](mailto:samohvalov@gmail.com)*

Разработана архитектура интеллектуальной информационной системы для поддержки принятия решений в задачах математической оптимизации. Представлены взаимосвязанные концептуальная (онтологическая) и функциональная модели, а также модель взаимодействия с пользователем (UML Use Case). Предложенное проектное решение обеспечивает формализацию предметной области, системный анализ параметров задачи для выбора метода и повышение надежности системы.

**Ключевые слова:** интеллектуальная информационная система; математическая оптимизация; системный анализ; концептуальная модель; функциональная модель; UML; архитектура программного комплекса.

## CONCEPTUAL AND FUNCTIONAL DESIGN OF AN INTELLIGENT INFORMATION SYSTEM ARCHITECTURE FOR MATHEMATICAL OPTIMIZATION PROBLEMS

D. S. Solovjev, I. A. Solovjeva, A. V. Samokhvalov

*Derzhavin Tambov State University,  
Tambov, Russia, [solovjevdenis@mail.ru](mailto:solovjevdenis@mail.ru), [good.win32@yandex.ru](mailto:good.win32@yandex.ru),  
[samohvalov@gmail.com](mailto:samohvalov@gmail.com)*

The architecture of an intelligent information system for decision support in mathematical optimization problems is developed. Interconnected conceptual (ontological) and functional models, as well as a user interaction model (UML Use Case), are presented. The proposed design solution ensures the formalization of the subject domain, systematic analysis of task parameters for method selection, and increased system reliability.

**Keywords:** intelligent information system; mathematical optimization; system analysis; conceptual model; functional model; UML; software complex architecture.

## 1. Введение

Проектирование сложных человеко-машинных систем, к классу которых относятся интеллектуальные информационные системы (ИИС), требует строгой формализации предметной области и функциональной архитектуры на начальных этапах разработки [1]. Особую актуальность данная задача приобретает при создании систем поддержки принятия решений для решения задач математической оптимизации, характеризующихся многокритериальностью, высокими вычислительными затратами и зависимостью результата от адекватности выбранного метода [2].

Целью данной работы является концептуальное и функциональное проектирование архитектуры ИИС для оптимизации, обеспечивающей эффективный выбор и применение численных методов на основе системного анализа параметров решаемой задачи. Достижение данной цели заключается в разработке взаимосвязанных моделей: концептуальной (онтологической) модели предметной области, функциональной модели (диаграммы декомпозиции) и модели взаимодействия с пользователем (UML Use Case), которые в совокупности образуют формальный фундамент для последующей программной реализации надежного и отказоустойчивого программного комплекса.

## 2. Концептуальное и функциональное проектирование ИИС оптимизации

В рамках проектирования сложных человеко-машинных систем ключевым этапом является разработка концептуальной и функциональной моделей. Эти модели позволяют формализовать предметную область, выявить ключевые сущности, их атрибуты и взаимосвязи, что является основой для последующего проектирования архитектуры программного комплекса.

На рис. 1 представлена концептуальная модель проектируемой интеллектуальной информационной системы (ИИС) для решения задач математической оптимизации [3]. Модель отображает ключевые объекты предметной области («Пользователь», «Задача оптимизации», «Метод оптимизации», «Результат»), их атрибуты и связи. Такая модель служит онтологическим основанием системы, обеспечивая однозначность интерпретации данных и процессов всеми компонентами ИИС.

Для описания функциональной структуры системы, декомпозиции ее на подсистемы и модули применяется функциональное моделирование. Система рассматривается как преобразователь входных данных (параметры задачи) в выходные данные (результат оптимизации) посредством ряда алгоритмических модулей. Диаграмма декомпозиции, изображенная на рис. 2, детализирует архитектуру ИИС, представляя ее в виде четырех взаимосвязанных подсистем.

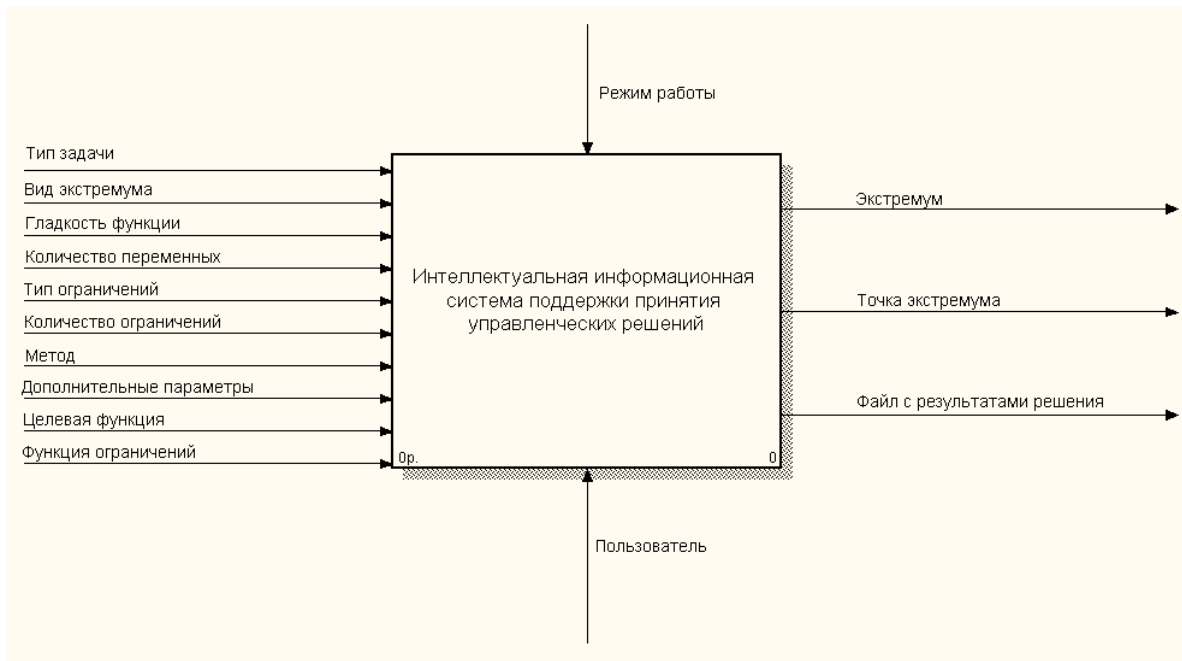


Рис. 1. Концептуальная модель предметной области, отражающая ключевые сущности и их взаимосвязи

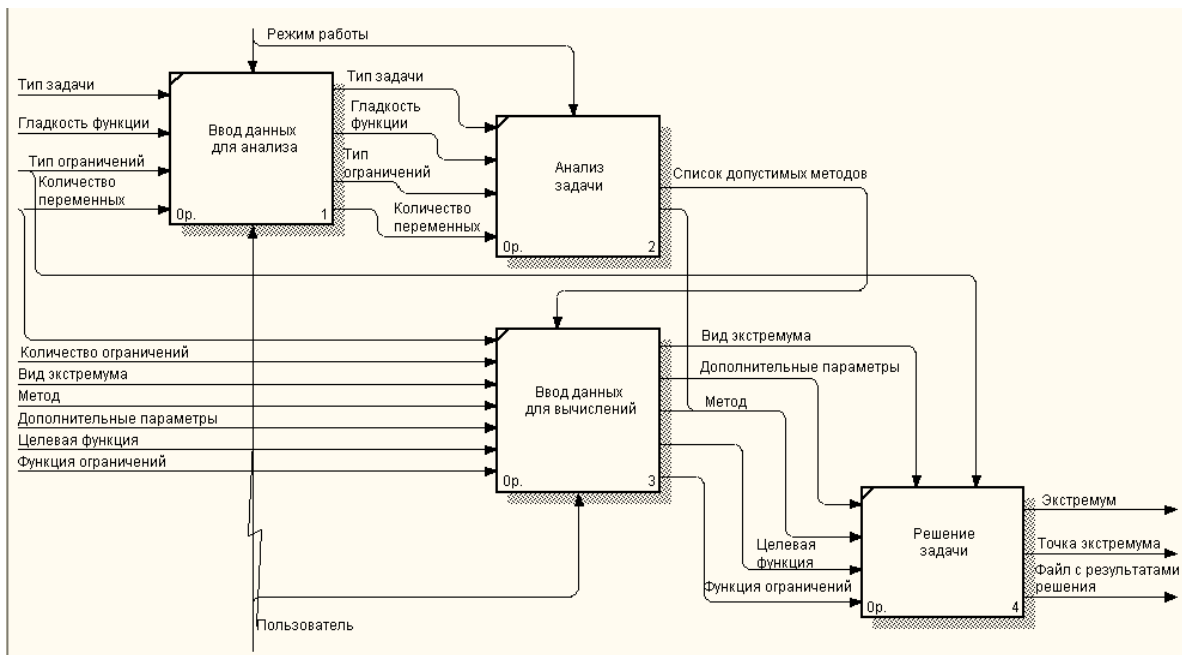


Рис. 2. Диаграмма декомпозиции (функциональная модель) ИИС

Логика взаимодействия подсистем основана на принципах управляющего и информационного потоков.

### 1. Подсистема «Ввод данных для анализа».

Данная подсистема реализует интерфейсный модуль для первичного взаимодействия с пользователем. Её функциональное назначение заклю-

чается в сборе метаданных, характеризующих решаемую оптимизационную задачу. К числу таких параметров относятся: тип оптимизации (минимизация или максимизация), свойство гладкости целевой функции, размерность пространства переменных, а также наличие и классификация ограничений. Полученная метаинформация выполняет роль управляющих параметров для последующих этапов вычислительного процесса, поскольку детерминирует множество допустимых алгоритмических стратегий для нахождения решения.

#### 2. Подсистема «Анализ задачи».

Данная подсистема реализует функцию системного анализа исходных условий задачи. На основе формализованного корпуса правил и базы знаний о методах оптимизации осуществляется генерация множества допустимых методов решения. Модуль функционирует по принципу экспертной системы, выполняя процедуры фильтрации и классификации методов на основе критериев их применимости к конкретной постановке задачи. Реализация данного модуля способствует повышению адекватности принимаемых решений и общей операционной эффективности ИИС.

#### 3. Подсистема «Ввод данных для вычислений».

Данная подсистема реализует функцию сбора и первичной обработки количественных данных, необходимых для работы выбранного алгоритма оптимизации. К её функциональным обязанностям относится интерфейс для задания целевой функции, системы ограничений (как равенств, так и неравенств), граничных условий на переменные, а также управляющих параметров метода. Критически важной функцией модуля является реализация процедур валидации вводимых данных. Эти процедуры направлены на проверку их полноты (наличие всех необходимых параметров) и непротиворечивости (отсутствие логических конфликтов между условиями задачи). Выполнение данного требования является обязательным условием обеспечения надежности и отказоустойчивости программного комплекса в целом, так как исключает вычисления для некорректных данных.

#### 4. Подсистема «Решение задачи».

Данная подсистема представляет собой вычислительное ядро комплекса. Ее функциональное назначение заключается в: реализации численных методов оптимизации; управлении процессом вычислений; обработке ошибок, возникающих в ходе работы итерационных алгоритмов; генерации результирующих данных, включающих координаты точки экстремума, значение целевой функции в данной точке и отчет о выполнении расчета. Передача результирующих данных пользователю завершает цикл управления информационным процессом в системе.

Для формального описания взаимодействия между ИИС и внешним действующим лицом (пользователем) применяется UML моделирование в

виде вариантов использования (Use Case) [4]. Диаграмма вариантов использования, представленная на рис. 3, иллюстрирует множество сервисов, предоставляемых системой, и является основой для формирования технических требований к интерфейсам и алгоритмам.

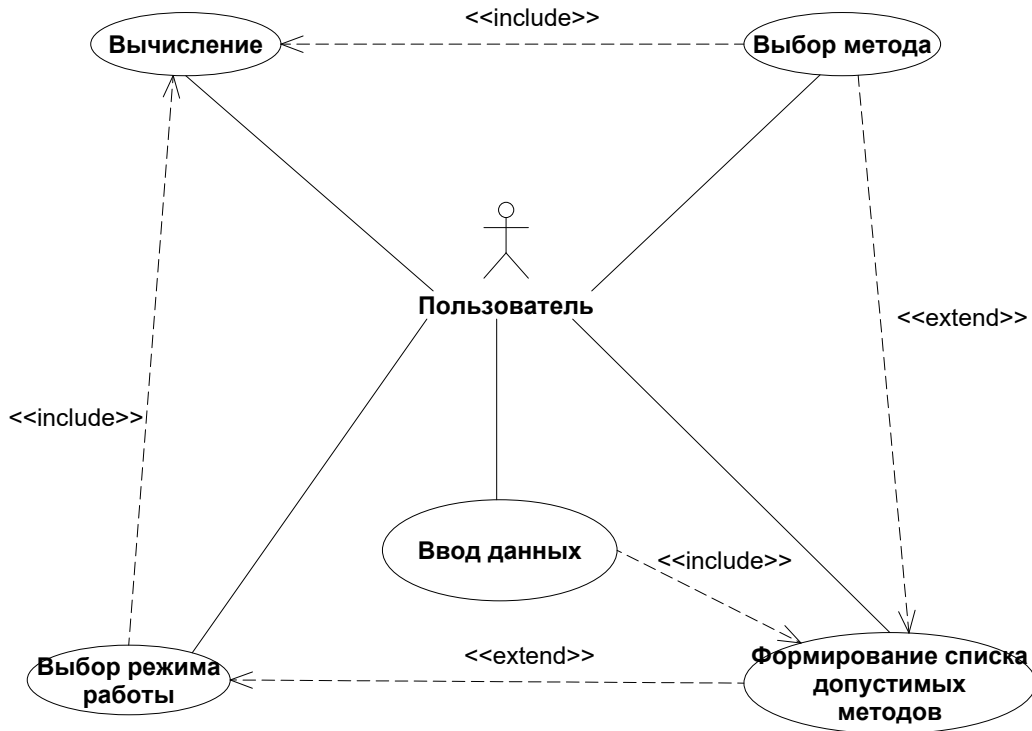


Рис. 3. Диаграмма вариантов использования

Сценарий взаимодействия пользователя с системой может быть описан следующим потоком событий.

I. Основной поток событий.

1. Инициализация сеанса и параметризация задачи. Пользователь инициирует вычислительный сеанс, задавая системные параметры оптимизационной задачи: тип (безусловная или условная), тип экстремума (минимизация или максимизация), класс гладкости целевой функции, размерность пространства переменных, а также количество и тип ограничений.

2. Спецификация модельных представлений. Пользователь определяет аналитический вид целевой функции и, при наличии ограничений, функций ограничений в соответствии с формальным синтаксисом, предписанным системой.

3. Анализ условий задачи и генерация множества методов. На основе формализованного описания задачи подсистема анализа производит идентификацию её свойств. В результате формируется и предоставляется пользователю множество алгоритмически релевантных методов оптимизации, адекватных установленным условиям.

4. Селекция метода оптимизации. Пользователь осуществляет выбор конкретного алгоритма из предложенного системой множества, принимая таким образом управляющее решение о методе решения.

5. Задание параметров алгоритма. Пользователь определяет значения дополнительных управляющих параметров, специфичных для выбранного алгоритма. К ним относятся: начальное приближение, критерии остановки (точность вычислений), параметры, управляющие сходимостью.

6. Выполнение вычислительного процесса и возврат результатов. Система исполняет вычислительный процесс, реализуя выбранный алгоритм (подсистема решения). По его завершении пользователю предоставляются результирующие данные: координаты найденного экстремума, значение целевой функции в этой точке, а также (опционально) траектория поиска.

7. Завершение сеанса работы. Вариант использования системы считается завершенным.

II. Альтернативные потоки (обработка исключительных ситуаций).

1. События 1а, 1б – ошибка валидации входных данных системного уровня (не задан тип экстремума). Система инициирует диалог для коррекции ввода, предотвращая некорректную работу модуля анализа.

2. События 2а, 2б – ошибка валидации математических моделей (неполнота ввода коэффициентов целевой функции или ограничений). Система запрашивает недостающие данные, обеспечивая целостность математической модели задачи.

3. События 5а, 5б – ошибка валидации параметров алгоритма (отсутствие обязательного параметра или его некорректное значение). Система детектирует потенциальный источник сбоя вычислительного модуля и запрашивает корректировку, повышая отказоустойчивость.

Наличие развитых альтернативных потоков свидетельствует о проектировании системы с учетом принципов надежности и устойчивости к ошибкам пользовательского ввода, что является критически важным для промышленных компьютерных систем.

## 5. Заключение

Проведенное исследование позволило достичь поставленной цели путем разработки проектного решения для ИИС оптимизации. Разработанная концептуальная модель, формализующая ключевые сущности предметной области («Пользователь», «Задача», «Метод», «Результат») и их взаимосвязи, служит онтологическим стандартом для обеспечения семантической согласованности всех компонентов системы. Функциональная декомпозиция системы на четыре взаимосвязанные подсистемы (ввода

метаданных, анализа задачи, ввода вычислительных данных и решения) реализует принцип разделения ответственности, что критически важно для управляемости и надежности сложных компьютерных систем. Формализация сценариев взаимодействия с пользователем с помощью диаграммы вариантов использования и описание альтернативных потоков для обработки исключений проектно закладывают в систему свойства устойчивости к ошибкам ввода и предсказуемости поведения.

Перспективы дальнейшего развития проекта видятся в следующих направлениях:

1) углубление экспертной базы знаний и внедрение алгоритмов машинного обучения для более точного и адаптивного выбора методов оптимизации на основе прецедентов;

2) распараллеливание вычислений для высокоразмерных задач и разработка улучшенных методов обработки ошибок итерационных алгоритмов;

3) проектирование и интеграция модулей для работы с новыми классами оптимизационных задач (стохастическими, многокритериальными) и разработка интерфейсов для интеграции системы в более крупные программные комплексы и киберфизические системы.

### **Библиографические ссылки**

1. Вендров А. М. Проектирование программного обеспечения экономических информационных систем. М. : Финансы и статистика, 2006.

2. Галеев Э. М., Тихомиров В. М. Оптимизация: теория, примеры, задачи. М. : Элиториал УРСС, 2000.

3. Черемных С. В., Семенов И. О., Ручкин В. С. Структурный анализ систем и IDEF-технологии. М. : Финансы и статистика, 2003.

4. Ларман К. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку. М. : Издательский дом «Вильямс», 2007.