

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

Кафедра дифференциальных уравнений и системного анализа

УРБАНОВИЧ

Дмитрий Игоревич

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ И ЗАДАЧА ОПТИМИЗАЦИИ ИГРЫ КАРКАССОН

Дипломная работа

**Научный руководитель:
ст. преподаватель И. И. Козлов**

Допущен к защите

«___» _____ 2025 г.

**Зав. кафедрой дифференциальных уравнений и системного анализа
канд. физ.-мат. наук, доцент Л. Л. Голубева**

Минск, 2025

ОГЛАВЛЕНИЕ

РЕФЕРАТ	3
РЭФЕРАТ	4
ABSTRACT	5
ВВЕДЕНИЕ	6
1 Описание игры и целевая задача	7
1.1 Правила и особенности игры Каркассон	7
1.2 Проблематика алгоритмического решения	8
2 Математическая модель игры	9
2.1 Графовое представление игрового поля	9
2.2 Отслеживание компонент связности	9
3 Разработка гибридного алгоритма оптимизации	11
3.1 Архитектура системы	11
3.2 Использование LLM в качестве эвристики	11
ЗАКЛЮЧЕНИЕ	13
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ	14
ПРИЛОЖЕНИЕ А	15
ПРИЛОЖЕНИЕ Б	16
ПРИЛОЖЕНИЕ В	17

РЕФЕРАТ

В дипломной работе __ страниц, __ иллюстраций, __ источников, __ приложения.

СТОХАСТИЧЕСКАЯ ИГРА, ОПТИМИЗАЦИЯ СТРАТЕГИИ, РАЗМЕЩЕНИЕ ТАЙЛОВ, LLM, MCP, PYTHON.

Объектом исследования дипломной работы является стохастическая игра с полной информацией на примере игры Каркассон.

Целью дипломной работы является разработка гибридного алгоритма оптимизации стратегии, объединяющего символьные вычисления (MCP) и эвристический поиск на базе LLM.

Для достижения поставленной цели были использованы: язык программирования Python, Model Context Protocol (MCP), а также большая языковая модель (LLM) через Ollama.

В дипломной работе получены следующие результаты:

1. Разработан игровой движок и MCP-сервер для вычисления допустимых ходов.
2. Реализована интеграция с LLM для генерации стратегических гипотез и выбора приоритетов.
3. Проведен анализ применимости LLM в качестве эвристической функции для задач комбинаторной оптимизации на графах.

Дипломная работа является завершенной, поставленные задачи решены, существует возможность дальнейшего развития подхода на другие настольные игры.

Дипломная работа выполнена автором самостоятельно.

РЭФЕРАТ

У дыпломнай працы __ старонак, __ малюнкаў, __ крыніц, __ дадаткаў.

СТАХАСТЫЧНАЯ ГУЛЬНЯ, АПТЫМІЗАЦЫЯ СТРАТЭГІІ, РАЗМЯШЧЭННЕ ТАЙЛАЎ, LLM, MCP, PYTHON.

Аб'ектам даследвання дыпломнай працы з'яўляецца стахастычная гульня з поўнай інфармацыяй на прыкладзе гульні Каркасон.

Мэтай дыпломнай працы з'яўляецца распрацоўка гібрыднага алгарытму аптымізацыі стратэгіі, які аб'ядноўвае сімвальныя вылічэнні (MCP) і эўрыстычны пошук на базе LLM.

Для дасягнення пастаўленай мэты выкарыстоўваліся: мова праграмавання Python, Model Context Protocol (MCP), а таксама вялікая моўная мадэль (LLM) праз Ollama.

У дыпломнай працы атрыманы наступныя вынікі:

1. Распрацаваны гульнявы рухавічок і MCP-сервер для вылічэння дапушчальных хадоў.
2. Рэалізавана інтэграцыя з LLM для генерацыі стратэгічных гіпотэз і выбару прыярытэтаў.
3. Праведзены аналіз прымяняльнасці LLM у якасці эўрыстычнай функцыі для задач камбінаторнай аптымізацыі на графах.

Дыпломнай праца з'яўляецца завершанай, пастаўленыя задачы вырашаны, прысутнічае магчымасць далейшага развіцця падыходу на іншыя настольныя гульні.

Дыпломнай праца выканана аўтарам самастойна.

ABSTRACT

Thesis project is presented in the form of an explanatory note of __ pages, __ figures, __ references, __ applications.

STOCHASTIC GAME, STRATEGY OPTIMIZATION, TILE PLACEMENT, LLM, MCP, PYTHON.

The research object of this thesis project is a stochastic game with full information, using the game Carcassonne as an example.

The purpose of the thesis is to develop a hybrid strategy optimization algorithm combining symbolic computation (MCP) and LLM-based heuristic search.

To achieve the goal, the following were used: Python programming language, Model Context Protocol (MCP), as well as a large language model (LLM) via Ollama.

The main results of the thesis project are as follows:

1. A game engine and MCP server were developed to compute valid moves.
2. Integration with LLM was implemented to generate strategic hypotheses and select priorities.
3. An analysis of the applicability of LLM as a heuristic function for combinatorial graph optimization problems was conducted.

The thesis work is completed, the tasks set have been solved, there is the possibility of further development of the approach to other board games.

The thesis project was done solely by the author.

ВВЕДЕНИЕ

Объектом исследования данной дипломной работы является стохастическая игра с полной информацией. В качестве среды для тестирования алгоритмов выбрана настольная стратегическая игра немецкого стиля (Eurogame) Каркассон. Особенностями данной игры являются пошаговое выкладывание тайлов и четкие правила топологической связности (дорога к дороге, город к городу).

Актуальность исследования обусловлена следующими факторами: пространство состояний и возможных решений в данной игре исключает возможность полного перебора вариантов. Задача оптимального размещения тайлов является NP-трудной и эквивалентна обобщенной задаче Edge-Matching Puzzle. Кроме того, наличие отложенного вознаграждения (например, начисление очков за поля в конце игры) значительно снижает эффективность классических жадных алгоритмов.

Целью дипломной работы является разработка гибридного алгоритма оптимизации стратегии, объединяющего символные вычисления (Model Context Protocol) и эвристический поиск на базе больших языковых моделей (LLM).

В ходе выполнения дипломной работы были поставлены следующие задачи:

- Описание игрового поля как динамического планарного графа, а размещения тайлов как задачи удовлетворения ограничений (CSP).
- Разработка игрового движка и сервера Model Context Protocol (MCP) для детерминированного расчета допустимости ходов и правил связности.
- Реализация интеграции и алгоритмов с использованием локальной LLM (Ollama) для генерации высокоуровневых стратегических гипотез.
- Проведение турнира агентов и сравнение гибридного подхода с существующими жадными алгоритмами.
- Оценка эффективности применения LLM в качестве эвристической функции.

ГЛАВА 1

Описание игры и целевая задача

1.1 Правила и особенности игры Каркассон

Игра «Каркассон» (Carcassonne) является настольной стратегической игрой немецкого стиля (Eurogame), в которой игровой процесс строится на пошаговом выкладывании квадратов местности (тайлов) и размещении на них фишек подданных (миплов) [5]. Пример игрового процесса представлен на рисунке 1.1.



Рисунок 1.1 Игровой процесс и выкладывание тайлов в Каркассоне

Игровое пространство формируется динамически в процессе игры. Каждый

новый тайл должен быть выложен по правилам топологической связности: грани выкладываемого квадрата должны логически продолжать объекты на уже лежащих на столе квадратах (дорога должна соединяться с дорогой, город с городом, поле с полем).

1.2 Проблематика алгоритмического решения

С точки зрения теории игр, Каркассон представляет собой стохастическую игру с полной информацией. Однако разработка оптимальной стратегии для искусственного агента сталкивается с рядом существенных трудностей:

- **Размер пространства состояний.** В базовой версии игры участвуют 72 тайла, которые могут быть выложены в различных конфигурациях. Количество допустимых позиций возрастает экспоненциально с каждым ходом [3].
- **NP-трудность.** Задача поиска оптимального размещения тайлов математически эквивалентна обобщенной задаче подбора краев (Edge-Matching Puzzle), которая в общем случае является NP-полной.
- **Проблема горизонта планирования.** Из-за механизма подсчета очков (в частности, отложенного вознаграждения за владение полями в самом конце игры) классические жадные алгоритмы (Greedy algorithms) принимают локально оптимальные, но глобально убыточные решения [6].

ГЛАВА 2

Математическая модель игры

2.1 Графовое представление игрового поля

Для формализации задачи игровое поле моделируется как динамический планарный граф $G = (V, E)$, где:

- V — множество вершин, представляющих собой отдельные функциональные сегменты тайлов (фрагменты города, участки дороги, поля, монастыри).
- E — множество ребер, обозначающих топологические связи между сегментами как внутри одного тайла, так и между смежными тайлами на игровом поле.

При добавлении нового тайла на поле граф G модифицируется: добавляются новые вершины и строятся новые ребра, объединяющие смежные сегменты. Пример графового представления игрового поля представлен на рисунке 2.1.

2.2 Отслеживание компонент связности

Для эффективного управления связностью объектов (например, для проверки того, завершен ли город или дорога) в программной модели используется структура данных *система непересекающихся множеств* (Disjoint-Set Union, DSU). DSU позволяет объединять независимые сегменты и находить корневой элемент компоненты за амортизированное почти линейное время $O(\alpha(n))$ [4], что является критически важным для производительности игрового движка при расчете возможных ходов.

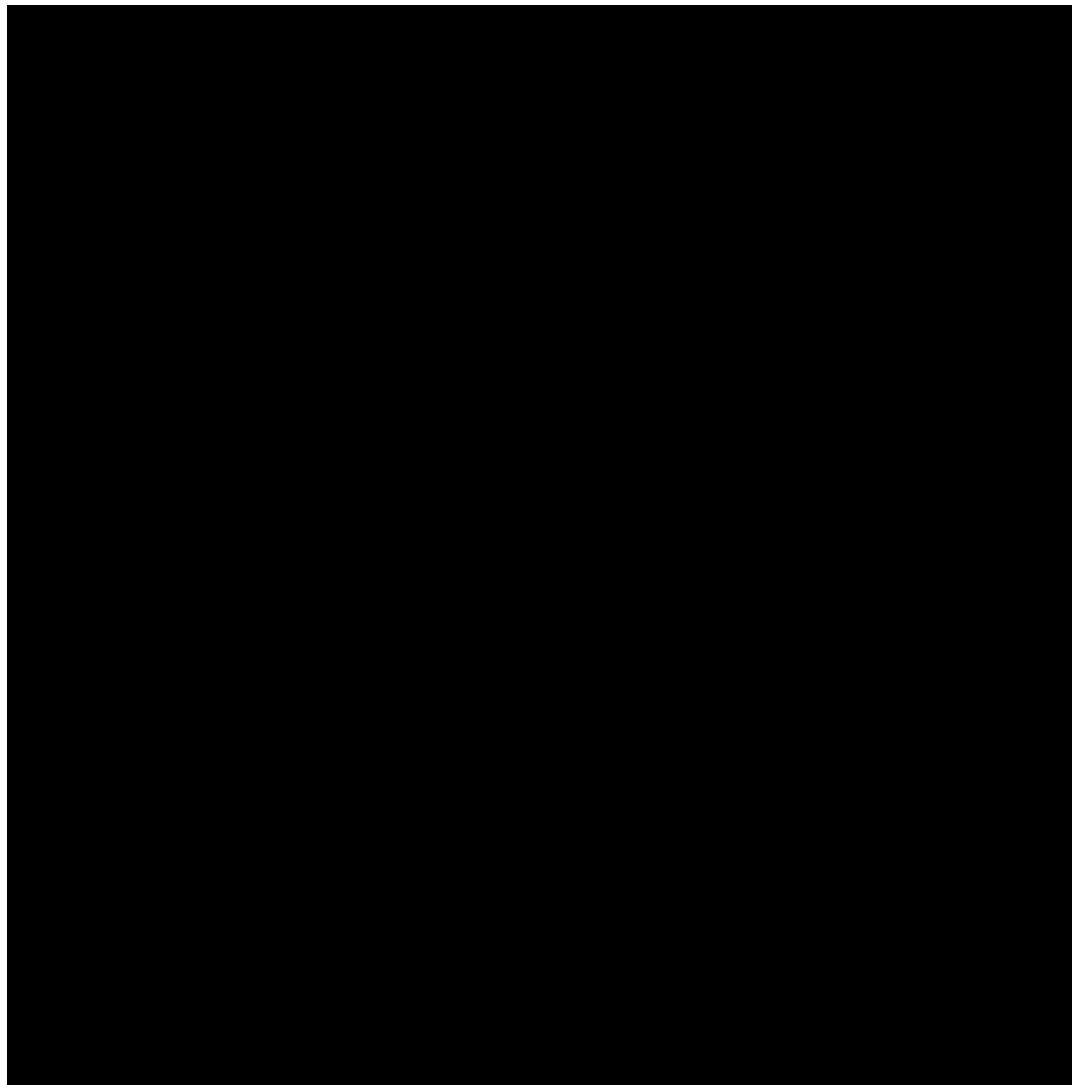


Рисунок 2.1 Графовое представление игрового поля

ГЛАВА 3

Разработка гибридного алгоритма оптимизации

3.1 Архитектура системы

Для решения задачи предлагается двухуровневая архитектура агента:

1. **Логический уровень (Logic Layer):** Игровой движок, написанный на языке Python, который инкапсулирует в себе строгую математику, геометрию на сетке и правила игры. Он реализован в виде сервера Model Context Protocol (MCP) [2], который детерминировано рассчитывает все возможные (допустимые с точки зрения правил) варианты хода в текущем состоянии.
2. **Стратегический уровень (Strategy Layer):** Эвристическая функция оценки и выбора оптимального хода, реализованная на базе локальной Large Language Model (LLM) через Ollama. LLM используется для генерации высокоуровневых стратегических целей (например, «заблокировать город противника» или «занять перспективное поле») [7].

Связь между математическим движком и стратегическим ИИ осуществляется поверх стандарта JSON-RPC 2.0 с использованием спецификации Model Context Protocol. Концептуальная схема архитектуры решения представлена на рисунке 3.1.

3.2 Использование LLM в качестве эвристики

Традиционные алгоритмы, такие как Monte Carlo Tree Search (MCTS) [1], требуют симуляции миллионов случайных партий для оценки состояния. В предложенной модели языковая модель выступает в роли интеллектуальной эвристики, анализирующей текущую конфигурацию графа и сужающей дерево поиска на основе заложенных в нее паттернов стратегического планирования.

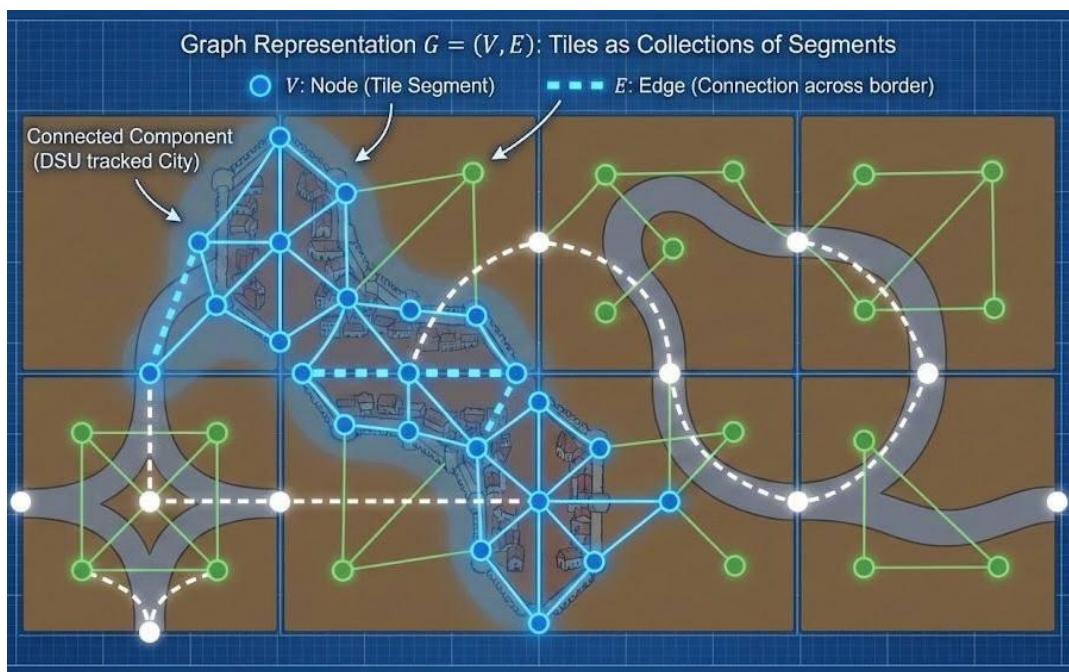


Рисунок 3.1 Схема архитектуры решения (Logic Layer и Strategy Layer)

ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломной работы была решена задача создания автономного игрового агента для стохастической настольной игры с полной информацией Каркассон.

Разработанный программный комплекс состоит из двух интегрированных слоев: детерминированного математического ядра (Logic Layer), реализованного в виде сервера Model Context Protocol, и эвристического модуля генерации стратегий (Strategy Layer), работающего на базе локальной языковой модели Ollama.

Полученные результаты демонстрируют принципиальную возможность использования современных больших языковых моделей (LLM) для решения NP-трудных задач комбинаторной оптимизации на динамических планарных графах. Архитектура взаимодействия посредством вызова инструментов (Tool Use / MCP) позволила компенсировать известную проблему галлюцинаций LLM и делегировать строгую проверку топологической связности и подсчет очков четкому алгоритмическому движку.

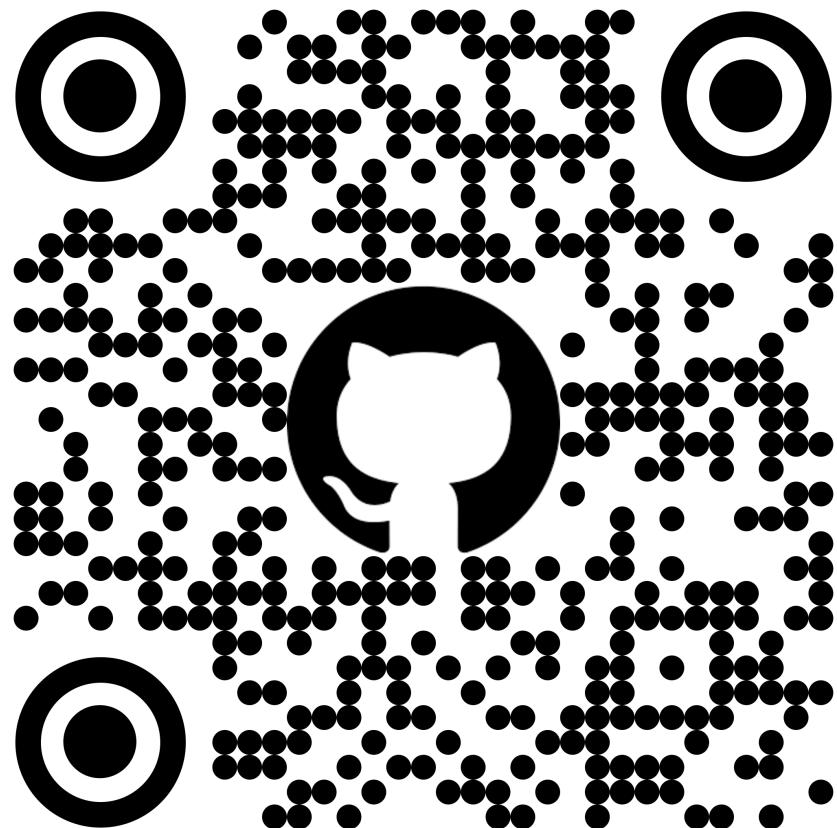
В качестве дальнейшего развития разработанного подхода планируется расширение эвристик агента за счет интеграции методов подкрепленного обучения (Reinforcement Learning) и алгоритмов поиска по дереву состояний (Monte Carlo Tree Search), а также адаптация обобщенного математического ядра для других настольных игр со схожей механикой размещения.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- [1] F. Ameneyro et al. Playing carcassonne with monte carlo tree search. 2020.
- [2] Anthropic. Model context protocol specification, 2024.
- [3] C. Heyden. Implementing a computer player for carcassonne. (star2.5 analysis). 2009.
- [4] D. E. Knuth. Dancing links. (exact cover algorithms).
- [5] L. Kárná. Carcassonne – description of the game. (theory of graphs application), 2012.
- [6] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction. (Теория стохастических сред и отложенного вознаграждения)*. MIT press, 2018.
- [7] S. Yao et al. Tree of thoughts: Deliberate problem solving with large language models. (Методология использования llm для стратегического планирования). 2024.

ПРИЛОЖЕНИЕ А

Репозиторий проекта на GitHub



YAP2P github repo qr-code

ПРИЛОЖЕНИЕ Б

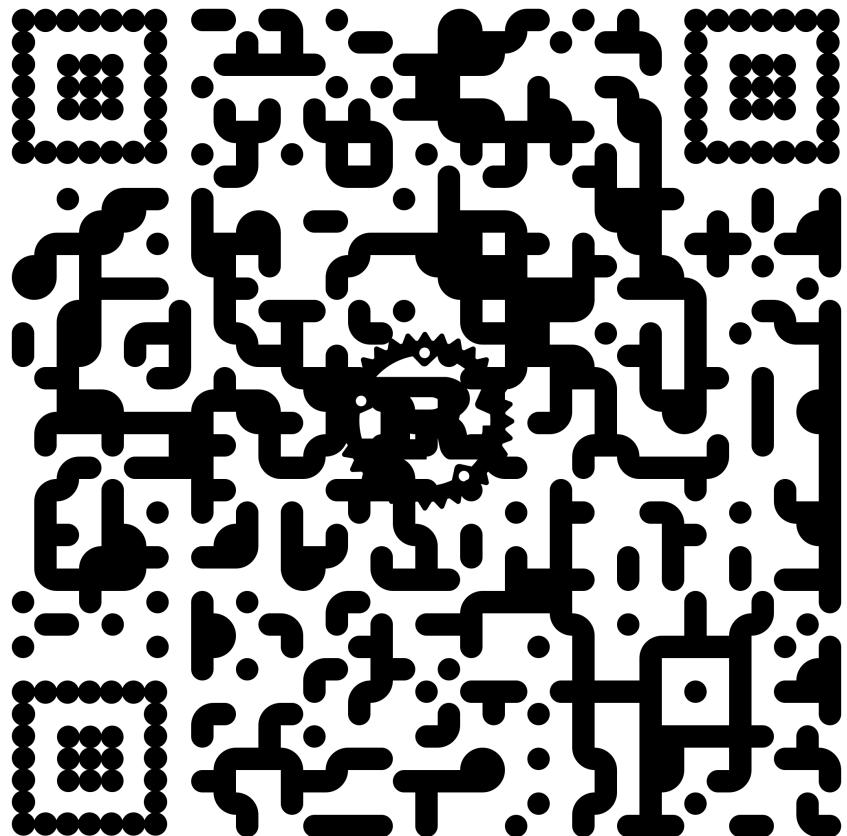
Страница библиотеки на crates.io



YAP2P crates.io page qr-code

ПРИЛОЖЕНИЕ В

Документация библиотеки



YAP2P documentation qr-code