

УДК 330.4:51-77

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ РЕШЕНИЯ ЗАДАЧИ ТРАНСПОРТНОЙ ЛОГИСТИКИ

Салиева Аделина Рустамовна

Студент НИЯУ МИФИ, г. Москва;

E-mail: adelina_salieva@mail.ru

Кудрявцев Константин Яковлевич

кандидат технических наук, доцент, преподаватель НИЯУ МИФИ, г. Москва;

E-mail: const58@mail.ru

Трифонов Иван Сергеевич

Студент НИЯУ МИФИ, г. Москва;

E-mail: kortess1999@mail.ru

Аннотация

Статья посвящена сравнению эффективности использования генетического алгоритма с другими алгоритмами для решения задачи транспортной логистики. Генетический алгоритм является одним из наиболее эффективных методов оптимизации, который основывается на принципах естественного отбора в природе. Авторы исследования предлагают использовать генетический алгоритм для оптимизации маршрутов в сфере транспортной логистики. В работе описывается процесс работы алгоритма, его тестирование и сравнение с другими методами оптимизации. Результаты исследования показывают, что генетический алгоритм является эффективным инструментом для решения задач оптимизации маршрутов.

Ключевые слова: логистика, транспортная логистика, методы оптимизации, алгоритм имитационного отжига, генетический алгоритм, метод наискорейшего спуска, задача коммивояжёра, задача маршрутизации транспорта.

STUDY ON THE EFFECTIVENESS OF IMPLEMENTING A GENETIC ALGORITHM FOR SOLVING TRANSPORTATION LOGISTICS PROBLEMS.

Adelina R. Salieva

Student NRU MEPhI, Moscow.

E-mail: adelina_salieva@mail.ru

Konstantin Ya. Kudryavtsev

Candidate of Technical Sciences, Associate Professor, lecturer at NRU MEPhI, Moscow;

E-mail: const58@mail.ru

Ivan S. Trifonov

Student NRU MEPhI, Moscow.

E-mail: kortess1999@mail.ru

ABSTRACT

This article is dedicated to comparing the effectiveness of using a genetic algorithm with other algorithms for solving transportation logistics problems. The genetic algorithm is one of the most effective optimization methods, which is based on the principles of natural selection in nature. The authors of the study propose using a genetic algorithm to optimize routes in the field of transportation logistics. The article describes the algorithm's working process, its testing, and comparison with other optimization methods. The research results show that the genetic algorithm is an efficient tool for solving route optimization problems.

Keywords: logistics, transportation logistics, optimization methods, simulated annealing algorithm, genetic algorithm, steepest descent method, traveling salesman problem, transportation routing problem.

Введение

Одним из методов оптимизации, который может быть использован для оптимизации маршрутов, является генетический алгоритм. При использовании генетических алгоритмов в транспортной логистике, можно учитывать различные критерии оптимизации, такие как минимизация времени доставки груза, расходов на топливо и транспортные услуги, максимизация загрузки транспортных средств, минимизация количества транспортных средств, учет ограничений по времени и местоположению, минимизация количества промежуточных точек доставки и максимизация безопасности и надежности доставки груза. Эти критерии могут быть включены в функцию оценки, которая используется генетическим алгоритмом для выбора лучших маршрутов доставки груза.

В качестве критериев эффективности алгоритмов предлагается использовать точность и скорость нахождения оптимального решения задачи транспортной логистики. Для оценки точности решения используется с реальными маршрутами доставки грузов. Для оценки скорости решения используется время выполнения алгоритмов на разных объемах данных и на разных вычислительных ресурсах.

В настоящее время генетический алгоритм используется в различных областях для решения задач, например, при восстановлении частично поврежденных файлов, разложении числа на простые множители в криптографии, оптимизации различных маршрутов и размеров доставляемых товаров в логистике, организации структуры инфокоммуникационных сетей и т. д. [2 - 9]. Задача коммивояжера и ее варианты широко используются в логистике и транспорте, когда несколько транспортных средств ограниченной грузоподъемности должны обслуживать клиентов в заданные временные интервалы. Примеры включают задачу маршрутизации транспортных средств с ограничением на грузоподъемность, доставку продуктов в магазины, подачу электроэнергии потребителям, построение кольцевой линии электропередач, а также задачи, связанные с автоматизацией монтажа схем и оптимизацией маршрута морских судов. Улучшение эффективности решения подобных задач может значительно снизить

финансовые затраты и время, поскольку существующие методы не обеспечивают достаточной скорости и находят только приближенные решения.

Постановка задачи

Traveling Salesman Problem (TSP) — это математическая задача, которая заключается в поиске наиболее оптимального маршрута для продавца, который должен посетить ряд городов и вернуться в исходный пункт. Она является NP-полной задачей, что означает, что ее решение может занять значительное количество времени, особенно при большом количестве городов. TSP имеет широкое применение в логистике, транспорте, планировании маршрутов и других областях, где необходимо оптимизировать перемещение между точками.

Для решения задачи коммивояжера необходимо определить оптимальный маршрут, проходящий через все города из заданного множества вершин, с минимальной стоимостью транспортировки. Для этого используется граф $G = (X, U)$, где X - множество вершин (городов), U - множество ребер (возможные пути между городами). Для каждой пары вершин i и j задана матрица чисел $D(i, j)$, которая представляет стоимость транспортировки из вершины x_i в x_j . В данной задаче все графы полно связные, поэтому координаты вершин заданы, а веса ребер определяются как декартово расстояние между вершинами.

$$D(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Цель работы заключается в нахождении оптимального маршрута, который проходит через все города из заданного множества ровно один раз, имеет замкнутый вид и имеет минимальную стоимость. Для этого используется полносвязный граф, где каждая вершина представляет город, а веса ребер определяются как декартово расстояние между городами. Целевая функция F , которую нужно минимизировать, представляет собой сумму стоимостей всех ребер в маршруте, начиная с первой вершины и заканчивая последней. Кратчайший гамильтонов цикл является перестановкой элементов множества вершин, которая обеспечивает минимальное значение целевой функции F . Таким образом, задача заключается в поиске кратчайшего замкнутого пути, проходящего через все вершины графа ровно один раз и имеющего минимальную суммарную стоимость.

$$F(\varphi) = D(\varphi(1), \varphi(n)) + \sum \{D(\varphi(i), \varphi(i+1))\} \rightarrow \min$$

Алгоритмы решения задачи транспортной логистики

Существуют различные методы, которые могут быть использованы для решения задачи оптимизации маршрутов. Рассмотрим наиболее популярные алгоритмы.

Генетический алгоритм – это алгоритм, основанный на эволюционной теории, который используется для решения задач оптимизации. Этот алгоритм моделирует процесс естественного отбора, где лучшие решения выбираются для следующего поколения. Генетический алгоритм может быть использован для оптимизации маршрутов, путем генерации случайных маршрутов и последующего выбора лучшего.

Алгоритм имитации отжига (англ. simulated annealing) — эвристический алгоритм глобальной оптимизации, особенно эффективный при решении дискретных и комбинаторных задач [13]. Этот алгоритм используется для нахождения оптимального решения, начиная с некоторого случайного решения. В процессе работы алгоритм выбирает новое решение и сравнивает его с текущим. Если новое решение лучше, оно принимается. Если оно хуже, то новое решение может быть принято с определенной вероятностью, которая уменьшается по мере уменьшения температуры.

Муравьиный алгоритм (алгоритм оптимизации подражанием муравьиной колонии, англ. ant colony optimization, ACO) — один из эффективных полиномиальных алгоритмов для нахождения приближённых решений задачи коммивояжёра, а также решения аналогичных задач поиска маршрутов на графах [11,12]. Суть подхода заключается в

анализе и использовании модели поведения муравьёв, ищущих пути от колонии к источнику питания, и представляет собой метаэвристическую оптимизацию. Первая версия алгоритма, предложенная доктором наук Марко Дориго [1, 2] в 1992 году, была направлена на поиск оптимального пути в графе [11]

Метод наискорейшего спуска – это алгоритм, который используется для нахождения минимума функции. Этот алгоритм определяет направление наискорейшего убывания функции и двигается в этом направлении, пока не достигнет минимума [14]. Метод наискорейшего спуска может быть использован для оптимизации маршрутов, путем определения функции, которая описывает стоимость маршрута, и последующего нахождения минимума этой функции.

Принцип работы генетического алгоритма.

Одним из самых эффективных алгоритмов решения задачи оптимизации маршрутов является генетический алгоритм. Основная идея генетического алгоритма заключается в том, чтобы представить маршрут в виде графа, где узлы представляют собой точки, а ребра – пути между точками. Затем создается начальная популяция маршрутов, где каждый маршрут представлен как набор генов, которые определяют порядок прохождения точек. Иллюстрация этого алгоритма показана на рисунке 1.

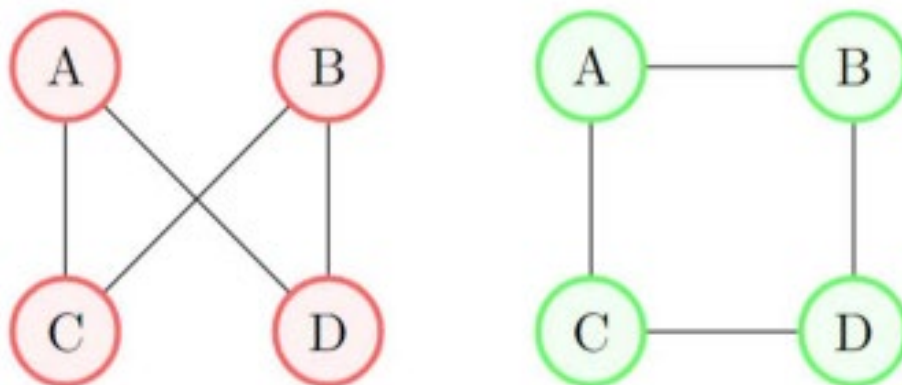


Рисунок 1. Работа генетического алгоритма

Предположим, что есть маршрут, проходящий через города A , B , C и D , как показано на рисунке 1 (слева). В этом случае есть два пути, которые пересекаются (пересекают друг друга), а именно путь от A до D (AD) и путь от C до B (CB). Для минимизации общего расстояния маршрута будет сформирован новый путь со следующими условиями:

$$d(A, B) + d(C, D) < d(A, D) + d(B, C), (1)$$

где $d(\cdot, \cdot)$ – метрика расстояния. Если условия в (1) выполняются, то пути AD и CB заменяются на пути AB и CD , чтобы текущий путь не содержал пересекающихся путей, как показано на рисунке 1 (справа). В процессе работы генетического алгоритма происходит эволюция популяции маршрутов. Для этого производится кроссовер (обмен генами) между двумя случайно выбранными маршрутами, а также мутация (случайное изменение генов) некоторых маршрутов. Затем производится оценка каждого маршрута с помощью функции приспособленности, которая оценивает качество маршрута в соответствии с поставленными задачами. Например, в случае логистических задач функция приспособленности может учитывать стоимость доставки грузов, время доставки, количество пересадок и другие факторы.

Маршруты, имеющие более высокую оценку приспособленности, имеют больше шансов перейти в следующее поколение популяции. Таким образом, в процессе эволюции популяции, маршруты с лучшей функцией приспособленности становятся все более распространенными и, в итоге, выбирается наилучший маршрут.

Сравнению генетического алгоритма с другими алгоритмами для решения задачи оптимизации маршрутов транспортной логистики

В данном исследовании были проведены численные эксперименты для оценки производительности генетического алгоритма. Проведено сравнение производительности данного алгоритма с алгоритмами имитационного отжига, который использует взаимодействие внутренних и внешних циклов и муравьиным алгоритмом.

Программное обеспечение разработано на языке программирования Python. Расчеты выполнялись на ПК с операционной системой Windows 10 pro-64-bit, процессором AMD RYZEN3 4000 SERIES 3.2GHz и 16 ГБ ОЗУ.

Для алгоритма имитационного отжига использовались следующие параметры:

- скорость охлаждения, $r = 0,9$,
- начальная температура, $T = 1000$,
- внутренний цикл (m) = количество городов,
- внешний цикл (n) = 20,
- критерии остановки внутреннего цикла: температура $< 0,01$ или итерация $>$ количество городов.

Сравнение генетического алгоритма и алгоритм имитации отжига

Результаты работы алгоритмов и имитационного отжига можно увидеть в таблицах 1 и 2 соответственно.

- В этих таблицах в столбцах содержатся следующие значения:
- имя задачи (название графа),
- оптимальное решение для экземпляра TSP.
- лучшее решение, полученное алгоритмом по испытаниям.
- ВРЕМЯ показывает среднее время, необходимое для одной попытки.
- Отклонение от нормы

Для алгоритма имитационного отжига результат не представлен, поскольку 20 испытаний проблемного экземпляра G202 не завершены в течение 48 часов. Одной из первоначальных целей было решить TSP с помощью модели MIP (Mixed Integer Programming) [15] и сравнить время решения. Для задач BM14и U16 оптимальное решение получается за секунды при решении соответствующей модели MIP с помощью CPLEX [10]. Однако после наблюдения за 22-городной задачей (U22) решение не найдено за 8000 сек, решено использовать оптимальные решения. Оптимальные затраты рассчитываются с использованием оптимальных туров, заданных для экземпляров задачи. Чтобы проверить влияние качества начальной популяции на генетический алгоритм, для некоторых проблемных случаев туры, найденные с помощью приложения, включаются в начальную популяцию. Результаты этого эксперимента приведены в таблице 3.

Таблица 1. Результаты работы генетического алгоритма

Имя задачи	Оптимальное решение [с]	Лучшее решение [с]	Время выполнения	Отклонение (%)
BM14*	3891	3891	<1 мин	2.98
U16*	9145	9145	<1 мин	1.00
U22*	8409	8409	<1 мин	1.78
C150***	6702	10235	1.2 мин	56.57

E51**	413	428	<1 мин	2.99
B52*	7830	8362	<1 мин	10.04
S70**	727	800	<1 мин	18.70
P76**	111594	120818	<1 мин	10.34
G96**	80904	96393	<1 мин	28.03
G202***	58735	92345	1.7 мин	71.80

*более 100 испытаний, **более 50 испытаний, ***более 30 испытаний

Таблица 2. Результаты работы алгоритма имитированного отжига

Имя задачи	Оптимальное решение	Лучшее решение	Время выполнения	Отклонение (%)
BM14*	3891	3891	<1 min	0.00
U16*	9145	9145	<1 min	0.00
U22*	8409	8409	<1 min	0.00
C150***	6702	6708	45.3 min	13.54
E51**	413	459	1.5 min	1.41
B52*	7830	7830	1.6 min	1.80
S70**	727	690	1.8 min	2.22
P76**	111594	115527	4.9 min	6.81
G96**	80904	81808	9.6 min	0.72
G202***	58735	NA	NA	NA

*более 100 испытаний, **более 50 испытаний, ***более 30 испытаний

Таблица 3. Влияние качества исходной совокупности на работу генетического алгоритма

Имя задачи	Оптимальное решение	Лучшее решение	Отклонение (%)
BM14*	3891	3891	0.00
U16*	9145	9145	0.00
U22*	8409	8409	2.19
C150***	6702	10235	0.47
E51**	413	428	0.00
B52*	7830	8362	6.81

Одним из вопросов, связанных с эффективностью генетического алгоритма, является влияние качества начальной популяции на производительность алгоритма. Результаты показывают, что наличие лучших особей в начальной популяции положительно влияет как на лучшее решение, так и на среднее решение генетического алгоритма, за исключением одного случая. Возможно, что в этом случае особи, найденные и помещенные в начальную популяцию, становятся доминирующими и приводят к субоптимальному туру. На рисунках 2 и 3 представлены сравнительные графики результата работы алгоритмов на разных итерациях.

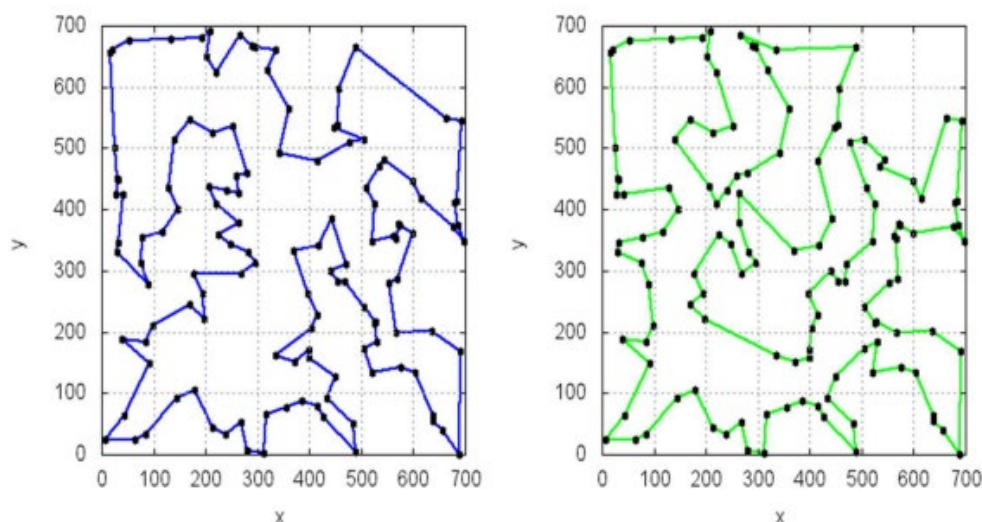


Рисунок 2. Результаты работы алгоритмов для решения задачи коммивояжера (промежуточный результат (35-я итерация)). Работа генетического алгоритма показана слева, имитационного алгоритма отжига - справа.

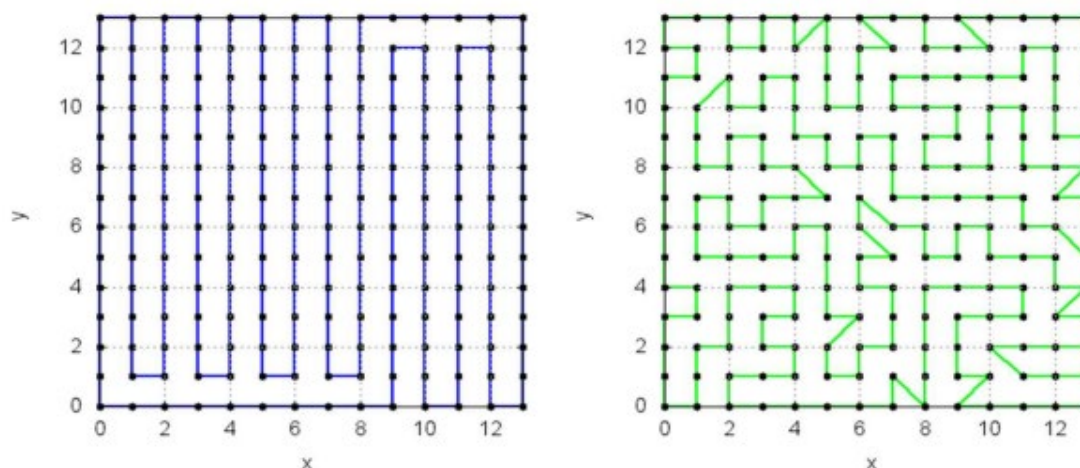


Рисунок 3. Результаты работы алгоритмов для решения задачи коммивояжера (итоговый результат работы алгоритмов). Работа генетического алгоритма показана слева, имитационного алгоритма отжига - справа.

Генетические алгоритмы могут быть применены на практике для решения задачи коммивояжера, которая заключается в нахождении оптимального маршрута, проходящего через заданные точки. Это может быть использовано при планировании доставки товаров и услуг, чтобы минимизировать время и затраты на транспортировку.

Сравнение алгоритма имитации отжига с муравьиным алгоритмом
На рисунках 4 и 5 показаны работы алгоритмов имитации отжига и муравьиного алгоритма. Следует отметить, что оптимум в имитации отжига был достигнут на 20 000 итерации, после чего алгоритм перестал изменяться из-за снижения вероятности перехода на новые маршруты при низкой температуре. Для решения этой проблемы были использованы критерии остановки. Высокая начальная температура увеличивает вероятность нахождения оптимального маршрута, но не гарантирует нахождения глобального оптимума.

В муравьином алгоритме проблема остановки отсутствует, так как он постоянно ищет лучшие пути. При бесконечном количестве итераций вероятность нахождения глобального оптимума стремится к единице. Однако выбор начальных параметров является ключевым моментом для успешной работы алгоритма и может быть решен методом проб и ошибок. Для оптимизации параметров муравьиной оптимизации был использован генетический

алгоритм на других координатах и с 50 городами, который дал некоторые параметры, приблизительно соответствующие данному примеру.

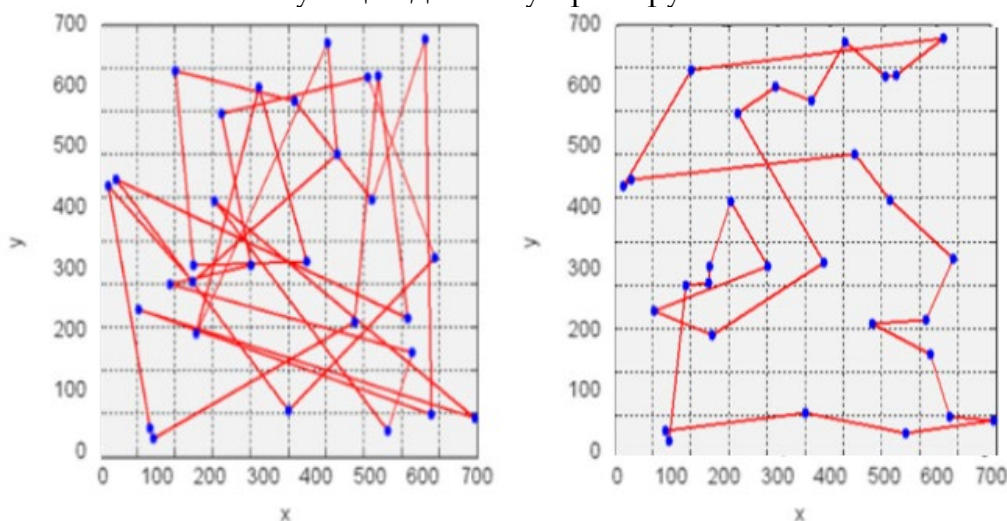


Рисунок 4. Результаты работы алгоритмов для решения задачи коммивояжера (промежуточный результат (30-я итерация)). Работа алгоритма имитационного отжига показана слева, муравьиного алгоритма отжига - справа

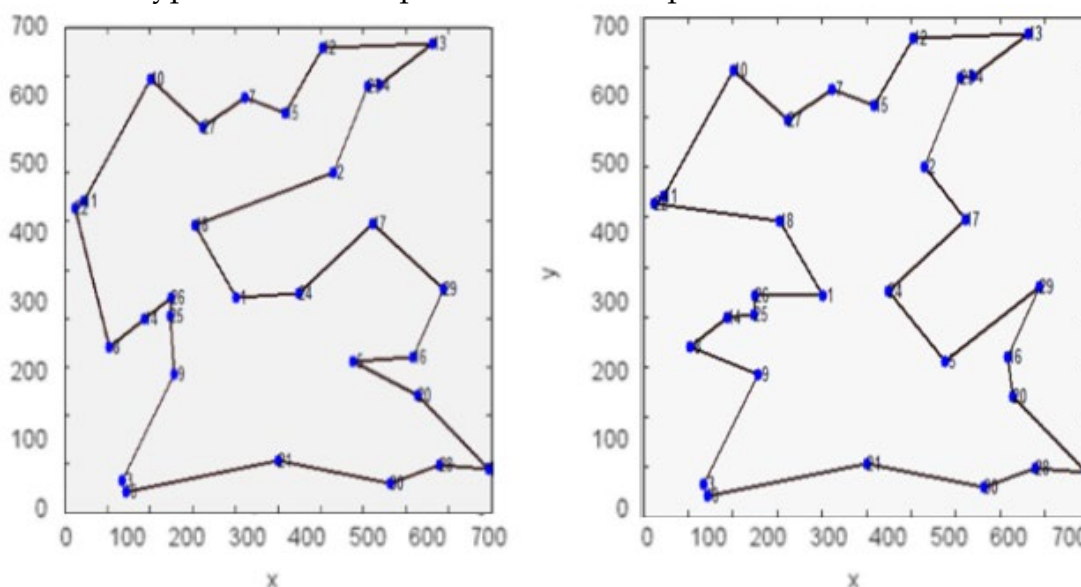


Рисунок 5. Результаты работы алгоритмов для решения задачи коммивояжера (итоговый результат). Работа алгоритма имитационного отжига показана слева, муравьиного алгоритма отжига - справа

Добавление критерия остановки тех же расстояний позволяет достичь тех же расстояний за 2,5 секунды без значительного изменения стандартного отклонения, что не относится к муравьиному алгоритму. Несмотря на то, что муравьиный алгоритм успешно работает до 50 городов, он начинает замедляться, когда достигает порога выше 50 и это объяснимо. Тем не менее, он продолжает двигаться к глобальному оптимуму, хоть и медленнее. Алгоритм имитационного отжига показывает результаты лучше не зависимо от количества городов и связи с этим можно сделать вывод, что алгоритм имитационного отжига использовать эффективнее, чем муравьиный алгоритм.

Заключение

Проведенное сравнение трех алгоритмов оптимизации показало, что генетический алгоритм является наиболее эффективным и показал лучшие результаты в решении задач оптимизации транспортной логистики.

Сравнение работы генетического алгоритма с работой алгоритма имитационного отжига

показало, что генетический алгоритм лучше решает задачу коммивояжера при наличии 16-225 городов.

Алгоритм имитационного отжига требует больше времени, чем генетические алгоритмы, и эта разница становится более заметной с увеличением размера задачи. Одной из основных причин этого является процедура поиска соседей, которая имеет сложность $O(n^2)$ и занимает все больше времени с увеличением размера задачи. Это также объясняет, почему алгоритм имитационного отжига может давать лучшие результаты, чем генетические алгоритмы в пробных прогонах - большее количество усилий по поиску дает лучшие результаты.

Сравнение алгоритма имитационного отжига и муравьиного алгоритма показали, что алгоритм имитационного отжига показывает результаты лучше, если исходных данных более 50. Муравьиный алгоритм показал хорошие результаты работы, если количество исходных данных меньше 50.

На задачах по построению оптимального маршрута по заданным маршрутам наиболее эффективным с точки зрения быстродействия и точности оказался генетический алгоритм, в то время как на задачах с меньшим количеством точек и итераций лучшие результаты показал муравьиный алгоритм. Общие рекомендации по применению этих алгоритмов заключаются в том, что выбор алгоритма должен быть основан на типе задачи и требованиях к точности и быстродействию. Дальнейшие исследования могут быть направлены на улучшение эффективности и точности алгоритмов путем комбинации различных методов оптимизации.

Список литературы:

1. Методы оптимизации http://opds.spbsut.ru/data/_uploaded/mu/tp/vlsa-present/tp_05_chisl_met_optim.pdf#:~:text=Методы%20оптимизации%20-%20это%20численные,возможных%20путей%20достижения%20экстремумов%20критериев
2. Студеникин Д.Е., Кондратьев С.И., Хекерт Е.В., Модина М.А. Динамическое формирование коридора безопасности при планировании маршрута движения судна. Морские интеллектуальные технологии. 2021. Т. 4. № 2 (53). С. 128-131
3. Методы оптимизации (Википедия) [https://ru.wikipedia.org/wiki/Оптимизация_\(математика\)обучение](https://ru.wikipedia.org/wiki/Оптимизация_(математика)обучение) - 2019.11.04
4. W. Fan, H. Xu, and X. Xu, "Simulation on vehicle routing problems in logistics distribution," COMPEL: International Journal for Computation & Mathematics in Electrical & Electronic Engineering, vol. 28, no. 6, pp. 1516-1531, 2009
5. H. Xiong, "Research on cold chain logistics distribution route based on ant colony optimization algorithm," Discrete Dynamics in Nature and Society, vol. 2021, Article ID 6623563, 1-10 pages, 2021
6. S. N. Sivanandam, S. N. Deepa. "Genetic Algorithms." In Introduction to Genetic Algorithms. Springer, 2008.
7. <https://www.mckinsey.com/~/media/McKinsey/Industries/Semiconductors/Our%20Insights/Smartening%20up%20with%20artificial%20intelligence/Smartening-up-with-artificial-intelligence.ashx> Iryanto, and P. H. Gunawan, "Numerical approach of symmetric traveling salesman problem using simulated annealing," Jurnal RESTI (Rekayasa Sistem

- Dan Teknologi Informasi), vol. 5, issue 6, pp.1090- 1098, 2021. <https://doi.org/10.29207/resti.v5i6.3549>
8. G. Reinelt, "TSPLIB", 19 February 1997. [Online]. Available at: https://www.researchgate.net/publication/340691026_Analysis_of_TSP_Simulated_Annealing_and_Genetic_Algorithm_Approaches
 9. P. H. Siqueira, S. Scheer, and M. T. A. Steiner. "A recurrent neural network to traveling salesman problem," Travelling Salesman Problem, I-Tech Veinna, pp. 135-156, 2008.
 10. ILOG Cplex. World Wide Web, <https://www.ibm.com/tr-tr/products/ilog-cplex-optimization-studio>.
 11. Муравьиный алгоритм https://ru.wikipedia.org/wiki/Муравьиный_алгоритм
 12. Муравьиный алгоритм http://www.serhiy-shtovba.narod.ru/doc/Shtovba_Ant_Algorithms_ExponentaPro_2003_3.pdf
 13. Алгоритм имитации отжига. <https://algorithmica.org/ru/annealing>
 14. Метод наискорейшего спуска <https://studfile.net/preview/7089694/page:2/>
 15. Что такое MIP https://www.ncss.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Mixed_Integer_Programming.pdf

References:

1. Optimization methods http://opds.spbsut.ru/data/_uploaded/mu/tpr/vlsa-present/tpr_05_chisl_met_optim.pdf#:~:text=Methods%20optimization%20-%20this%20counted,possible%20paths%20achievement%20extremes%20criteria
2. Studenikin D.E., Kondratiev S.I., Hekert E.V., Modina M.A. Dynamic formation of a safety corridor when planning a ship's route. Marine intelligent technologies. 2021. Vol. 4. No. 2 (53). pp. 128-131
3. Optimization Methods (Wikipedia) [https://ru.wikipedia.org/wiki/Optimization_\(Mathematics\)](https://ru.wikipedia.org/wiki/Optimization_(Mathematics)) training - 2019.11.04
4. W. Fan, H. Xu, and X. XuXu, "Simulation on vehicle routing problems in logistics distribution," COMPEL: International Journal for Computation & Mathematics in Electrical & Electronic Engineering, vol. 28, no. 6, pp. 1516-1531, 2009
5. H. Xiong, "Research on cold chain logistics distribution route based on ant colony optimization algorithm," Discrete Dynamics in Nature and Society, vol. 2021, Article ID 6623563, 1-10 pages, 2021
6. S. N. Sivanandam, S. N. Deepa. "Genetic Algorithms." In Introduction to Genetic Algorithms. Springer, 2008.
7. <https://www.mckinsey.com/~/media/McKinsey/Industries/Semiconductors/Our%20Insights/Smartening%20up%20with%20artificial%20intelligence/Smartening-up-with-artificial-intelligence.ashx> Iryanto, and P. H. Gunawan, "Numerical approach of symmetric traveling salesman problem using simulated annealing," Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi), vol. 5, issue 6, pp.1090- 1098, 2021. <https://doi.org/10.29207/resti.v5i6.3549>
8. G. Reinelt, "TSPLIB", 19 February 1997. [Online]. Available at: https://www.researchgate.net/publication/340691026_Analysis_of_TSP_Simulated_Annealing_and_Genetic_Algorithm_Approaches

9. P. H. Siqueira, S. Scheer, and M. T. A. Steiner. "A recurrent neural network to traveling salesman problem," Travelling Salesman Problem, I-Tech Veinna, pp. 135-156, 2008.
10. ILOG Cplex. World Wide Web, <https://www.ibm.com/tr-tr/products/ilog-cplex-optimization-studio>.
11. Ant algorithm https://ru.wikipedia.org/wiki/Ant_algorithm
12. Ant algorithm http://www.serhiy-shtovba.narod.ru/doc/Shtovba_Ant_Algorithms_ExponentaPro_2003_3.pdf
13. Algorithm of simulated annealing. <https://algorithmica.org/ru/annealing>
14. The method of the steepest descent <https://studfile.net/preview/7089694/page:2/>
15. What is MIP https://www.ncss.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Mixed_Integer_Programming.pdf