ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 39 за 2024 г.

ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

Д.Н. Шмыглёв, В.А. Судаков

Модель обучения с подкреплением для оптимизации автопарка предприятия

Рекомендуемая форма библиографической ссылки: Шмыглёв Д.Н., Судаков В.А. Модель обучения с подкреплением для оптимизации автопарка предприятия // Препринты ИПМ им. М.В.Келдыша. 2024. № 39. 13 с. https://doi.org/10.20948/prepr-2024-39
https://library.keldysh.ru/preprint.asp?id=2024-39

О р д е н а Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й а к а д е м и и н а у к

Д.Н. Шмыглёв, В.А. Судаков

Модель обучения с подкреплением для оптимизации автопарка предприятия

Москва — 2024

Шмыглёв Д.Н., Судаков В.А.

Модель обучения с подкреплением для оптимизации автопарка предприятия

Данная работа освещает решение задачи поиска минимального размера автопарка предприятия, с которым можно решать задачи, аналогичные задаче нескольких коммивояжеров. Предложенный подход моделирует среду обучения с подкреплением, где агент должен объехать заданные точки маршрута множеством транспортных средств. Проведенные вычислительные эксперименты показали эффективность моделей машинного обучения при решении задачи комбинаторной оптимизации.

Ключевые слова: обучение с подкреплением, комбинаторная оптимизация, задача коммивояжёра, автопарк

Dmitry Nikolaevich Shmyglev, Vladimir Anatolyevich Sudakov

Reinforcement Learning Model for Enterprise Fleet Optimization

This work highlights the solution to the problem of finding the minimum size of an enterprise’s vehicle fleet, with which it is possible to solve problems similar to the problem of several traveling salesmen. The proposed approach models a reinforcement learning environment where an agent must drive around given waypoints with multiple vehicles. The conducted computational experiments showed the effectiveness of machine learning models in solving the combinatorial optimization problem.

Key words: reinforcement learning, combinatorial optimization, traveling salesman problem, car fleet

Оглавление

Введение

Цель данного исследования выражается в разработке модели обучения с подкреплением, в процессе тренировки которой выявляется лучший эпизод – предложенное моделью решение. Точные методы расчета оптимальных вариантов, имея достаточное количество времени, смогут определить конкретное количество единиц транспорта, необходимое для посещения всех пунктов. Однако проверка всех множеств решений может потребовать от предприятия больших временных затрат, особенно при внушительном количестве точек в маршруте. Рассматриваемую проблему можно представить как прикладную задачу о коммивояжере [1].

Комбинаторная оптимизация, присущая задаче коммивояжера, сводится к нахождению кратчайшего пути, покрывающего все точки маршрута. Ввиду наличия точек и их связей, выраженных в расстоянии между ними, задача удобно представима в виде графа, позволяющего наглядно проконтролировать эффективность составленного плана передвижения.

Действительно, в исследуемой области задач существует необходимость посещения всех пунктов в некоторой области. Однако оптимизация общего пройденного расстояния в рассматриваемой проблеме играет второстепенную роль. Основной целью разрабатываемой модели является определение оптимального количества единиц транспорта, требующегося для обхода всех имеющихся пунктов.

Определение оптимального размера автопарка поможет не только выявить необходимое число транспортных средств, а также выяснить, является ли имеющееся количество используемого организацией транспорта избыточным или же недостаточным. Другими словами, модель подойдет как для начинающего предприятия, только планирующего влиться в бизнес, так и для крупных компаний, стремящихся избавиться от лишних затрат или войти в новую область.

Данная работа предполагает исследование построенной модели в сфере автомобильных перевозок, однако интерпретация решения может накладываться и на другие области оптимизации транспорта. Например, для сети аэропортов при соответствующей модификации структуры модели можно выявлять оптимальное количество самолетов, способных обеспечить пассажиропоток в сети аэропортов.

В стандартной задаче коммивояжёра (англ. Traveling Salesman Problem, далее – TSP) действующим лицом является единственный “путник”, проходящий через все точки маршрута. Для адаптации этой задачи под круг рассматриваемых проблем вводятся определенные ограничения, в результате наложения которых возможно прохождение требующихся точек несколькими действующими единицами.

Применение результатов моделирования подразумевает выполнение периодических поставок с использованием автопарка полученного размера и

прохождение соответствующего количеству транспортных средств оптимального маршрута.

Наиболее широкое освещение задача коммивояжера получила в работе В.И. Мудрова [1]. Автор приводит постановку и методы решения описываемой им проблемы. Приводятся как точные, так и приближенные алгоритмы вычисления лучшего пути. В дальнейшем для решения TSP были применены и другие алгоритмы, вроде муравьиного [2] и генетического [3].

Машинное обучение с подкреплением играет большую роль в построении комплексных систем принятия решений. Принципы и устройство алгоритма тренировки подобных моделей подробно изложены в работе за авторством Саттон Р.С. и Барто Э.Дж. Книга посвящена теоретическому изложению материала из области обучения с подкреплением [4].

На просторах интернета можно найти множество модифицированных задач коммивояжера, решенных методами обучения с подкреплением. Наиболее близкой задачей к рассматриваемой является задача коммивояжера с пополнением топлива, решенная Оттони и другими авторами [5].

В открытых источниках совмещение задачи оптимизации размера автопарка происходит с транспортной задачей [6]. Данная работа предполагает рассмотрение поиска оптимального числа транспортных средств в области TSP, позволяя уменьшить время прохождения всех точек маршрута.

Время решения комбинаторных задач заметно растет с увеличением числа требующих обхода пунктов. Задача поиска алгоритма, достигающего оптимума быстрее и эффективнее, всегда будет актуальна.

Настоящая работа предполагает разработку решения, способного методами машинного обучения с подкреплением достигать результата, близкого к оптимальному. Обученная модель может выдавать хорошие решения, однако в процессе тренировки может появиться лучший результат. Степень эффективности решения можно проверять, анализируя уровни загруженности используемых транспортных средств. Современное программное обеспечение позволяет ускорять процесс обучения модели.

Описание метода

Для более конкретного представления задачи оптимизации размера автопарка предприятия в виде математической модели приводится семантическое описание ее структуры. При следовании выведенным формулировкам вероятность упустить некоторые аспекты или построить избыточно сложную модель сводится к минимуму.

Построить маршрут для наземного транспорта – значит получить последовательность, отражающую порядок посещения точек маршрута. В случае нескольких автомобилей маршрут строится для каждого транспортного средства. Стоит отметить, что в таком случае каждый пункт может быть посещен лишь один раз.

Таким образом, общая постановка задачи поиска минимального

количества используемого транспорта сводится к нескольким положениям:

- имеется набор пунктов, которые необходимо посетить;
- известно попарное расстояние между всеми точками;
- один из пунктов является базой, куда необходимо вернуться после посещения всех точек маршрута;
- на практике время использования одной единицы транспорта ограничено длительностью смены;
- каждый автомобиль перемещается между пунктами, не входящими в план посещения других машин;
- по окончании смены автомобиль должен возвращаться на базу;
- в результате построения маршрута не остается непосещенных пунктов;
- каждая машина должна двигаться по маршруту, обеспечивающему минимальное общее пройденное расстояние.

Набор пунктов отражает некую подсеть, требующую автоматизации. В данной работе используются точки АЗС одной из крупных российских сетей. Получение хорошего результата на реальных данных будет свидетельствовать о применимости модели в деятельности организаций высокого уровня.

Матрица расстояний между пунктами необязательно будет симметричной. В городе и его окрестностях могут присутствовать односторонние дороги, движение по которым может быть оптимально только в одну сторону. Другими словами, точки маршрута и пути между ними в общем случае представляют собой ориентированный граф. Так как фактор направления движения должен быть учтен в модели, количество возможных вариантов построения маршрута для n пунктов равно $(n-1)!$. Единица вычитается ввиду наличия базового пункта, с которого маршрут одного автомобиля всегда начинается и которым всегда заканчивается. Факториальная сложность алгоритмов перебора делает задачу трудно решаемой при больших n. Алгоритмы обучения с подкреплением позволяют получить приближенное решение, рассмотрев подмножество маршрутов.

Базовый пункт отражает место, где автомобили готовятся к выезду. Это может быть склад, точка заправки, автобусная остановка и т.п. В рамках ограничения времени передвижения одного транспортного средства смена гарантирует готовность машин к следующей смене по окончании текущей. Таким образом можно решать проблему регулярных поставок в пункты.

В данной задаче длительность смены равна 12 часам. Шоферы-дальнобойщики развозят топливо по АЗС и по окончании смены возвращаются на нефтебазу.

Модель обучения с подкреплением подразумевает взаимодействие среды и агента. Задачей человека при моделировании является разработка логик назначения вознаграждений агенту за совершение им действий и перехода среды в различные свои состояния.

Тренировка модели машинного обучения с подкреплением происходит посредством анализа наборов действий агента и реакций среды на них, в совокупности называющихся эпизодами. Целью обучения такой модели является выявление лучшего действия в каждой ситуации. Для этого существует несколько подходов, строящихся на выборе критерия оптимизации модели. В данной задаче используется алгоритм PPO (Proximal Policy Optimization) [7], направленный на максимизацию ожидаемого общего выигрыша за эпизод, зависящего от вероятностей принятия того или иного решения в данном состоянии среды. Процесс оптимизации происходит за счет применения алгоритма градиентного подъема, корректирующего распределение вероятностей.

Поиск оптимального решения всегда сопряжен с отбросом неэффективных вариантов. Некоторые неподходящие решения заметны лучше других. В зависимости от задачи количество очевидных неоптимальных действий разнится.

Так, при построении маршрутов для нескольких транспортных единиц есть свой набор неподходящих решений в зависимости от ситуации. Следует учитывать несколько правил, гарантирующих сколько-нибудь адекватную оценку оптимальности действия:

1. Машина не должна возвращаться на базу, пока либо длительность ее смены не будет подходить к концу, либо не останется непосещенных пунктов. Это дает гарантию полной реализации транспортом своего потенциала.
2. Простаивание на одном месте является пустой тратой времени. Если машина не совершает никакого движения, общий маршрут может не построиться.
3. Каждая машина посещает свои точки ровно один раз. Подразумевается, что выбираются минимальные расстояния между пунктами, что в общем случае обеспечивает применение правила треугольника: возвращение в посещенные точки не уменьшит пройденного расстояния.

Совершение агентом описанных выше нежелательных действий свидетельствует о неоптимальности полученного варианта. В стандартной реализации PPO и других алгоритмов модель научится не использовать эти действия, однако существуют методы ручного их отсеивания. Одним из них является маскирование действий. Пометив действие как неэффективное, можно свести вероятность его применения на следующем шаге к нулю. Такой подход позволяет ускорить обучение модели, т.к. заведомо неоптимальные эпизоды не рассматриваются.

После описания семантических особенностей рассматриваемой задачи удобно составить математическую модель в виде критерия оптимизации размера автопарка предприятий и имеющихся ограничений на значения переменных.

Пусть весь маршрут состоит из n пунктов, один из которых является

точкой отправления и возвращения каждого автомобиля. Тогда размерность матрицы расстояний между пунктами D составляет $n \times n$ с элементами d_{ij}-расстоянием от точки i до точки j по дорогам. Оптимальным способом обойти все пункты является выбор каждой машиной маршрута, обеспечивающего минимальное общее пройденное расстояние.

Положим, что модель использует m автомобилей. Составляется m маршрутов. Для идентификации пути каждого транспортного средства вводится многомерная матрица индикаторов X размерности $n \times n \times k$. Элементы матрицы X x_{ijk} показывают, использовался ли путь между точками i и j на маршруте k. Если машина проезжает между данными пунктами, значение элемента матрицы равно 1, иначе 0. Так как каждую точку (кроме базовой) можно посетить лишь 1 раз, сумма индикаторов на маршруте $k \sum_i \sum_j x_{ijk}$ выражает количество посещенных точек на этом маршруте.

Один автомобиль может проехать сколько угодно пунктов, пока время их посещения укладывается в смену. Пусть длительность смены имеет значение t_{max}. Для удобства интерпретации введем новое значение d_{max}, выражающее максимальное расстояние, которое может проехать одна машина со средней скоростью v_{avg}: $d_{max} = v_{avg} * t_{max}$. Расстояние, преодоленное на маршруте k, должно быть не больше d_{max}.

Таким образом, имеются два критерия оптимизации: количество автомобилей (и, соответственно, маршрутов) m и общее пройденное расстояние l, которое можно представить в виде поэлементной суммы произведений матрицы индикаторов X на матрицу расстояний D на каждом маршруте: $l = \sum_k \sum_i \sum_j x_{ijk} * d_{ij}$.

В таком случае постановка задачи выглядит следующим образом:

$$m \to min; \quad (1)$$

$$l = \sum_k \sum_i \sum_j x_{ijk} * d_{ij} \to min; \quad (2)$$

$$\sum_i \sum_j x_{ijk} * d_{ij} \leq d_{max}, \forall k; \quad (3)$$

$$i = \overline{1,\ n}; \quad (4)$$

$$j = \overline{1,n}; \quad (5)$$

$$k = \overline{1,m}. \quad (6)$$

Нахождение оптимального состояния полученной модели определит оптимальный размер автопарка для посещения всех точек маршрута.

Решить данную задачу значит подобрать число m, матрицу X для него, обеспечивающие условия оптимальности. Прямое решение данной задачи может быть затруднительным. В таком случае можно воспользоваться алгоритмами обучения с подкреплением, дабы найти приближенный к оптимальному вариант. Для этого требуется описать состояния среды и логику действий агента.

Особенностью данного метода является рассмотрение нескольких машин как одной, постоянно возвращающейся на базу. Это позволяет не использовать

сложную структуру модели и сохранять эффективность решения. Так, если автомобиль возвращается на базу, счетчик расстояния сбрасывается. В дальнейшем каждый выезд из базы можно рассматривать как самостоятельный автомобиль.

Также отличительной чертой алгоритма является отсутствие прямой оптимизации m: при минимизировании общего расстояния каждая новая машина – дополнительные выезд и возвращение на базу, что уменьшает общий доход. Во время обучения алгоритм понимает, что оптимальное количество транспортных средств – минимально возможное для выполнения задачи.

В качестве среды, в которой действует агент, используются индикаторы непосещенных точек и последняя посещенная на данном маршруте точка. Таким образом, состояние среды S записывается как вектор размерности$n + 1$ с элементами $s_i \in \{0; 1\}, i = \overline{1, n}$ и $h \in \{1,2,\dots,n\}$:

- $S = (s_1, s_2, \dots, s_n, h).$ (7)

Совершение агентом действий подразумевает выбор следующей точки маршрута: $a \in \{1,2,\dots,n\}$. Стоит отметить, что при наличии масок действий выбираются только допущенные варианты:

- $a \in \{1,2,\dots,n\} \backslash \{i: s_i = 1\}, i = \overline{1, n}.$ (8)

Маска является частью среды и задается как бинарный вектор. Соответственно, она изменяется в зависимости от состояния, в которое переходит среда.

Агент может выбирать только те точки, что еще не были посещены. Так соблюдается единственность прохода по каждой точке. Базовый пункт обычно запрещен для посещения до окончания срока смены.

После совершения действия последнее запрещается, чтобы избежать простаивания на месте. Таким образом, маски накладываются друг на друга, создавая множество решений M, которые можно выбрать на следующем ходу: $M = v_1 \cup v_2 \cup \dots \cup v_k$, k – количество масок, создаваемых в ходе работы алгоритма.

После совершения каждого хода агент оценивает, сможет ли он вернуться до базы из любого разрешенного пункта, т.е. хватит ли оставшейся дистанции в смене для совершения такого маневра. Если расстояние превышает порог d_{max}, такой пункт тоже становится недоступным. Если пунктов для посещения нет, разрешается вернуться на базу для обозначения начала нового маршрута.

Наградой агента, с помощью которой ведется процесс определения ценности решений, являются обратные к расстоянию величины: $R(s, a) = \frac{1}{d_{i,a}}$, где i – последняя пройденная точка, а a – действие агента. Таким образом, чем больше суммарная награда, тем меньше знаменатели величин – тем меньшее расстояние было пройдено за маршрут.

Программная реализация модели выполнена на языке Python на каркасе Ray с помощью библиотеки RLlib. Возможность легкого масштабирования и

распараллеливания процессов на графических ускорителях позволяет сократить время тренировки модели и применять модель к данным различных объемов.

Демонстрация решения

Чтобы применить полученную модель на практике, необходимо определить основные переменные и матрицу расстояний. В рассматриваемой задаче $t_{max} = 12, v_{avg} = 60 \Rightarrow d_{max} = 720$. Другими словами, за смену одна машина может проехать максимум 720 км.

Матрица расстояний опирается на 49 точек АЗС и нефтебазу одной из крупнейших российских сетей АЗС, образуя сеть, включающую 50 пунктов. Сбор данных осуществлялся с помощью OpenStreetMap и OSRM API. Полученная информация преобразовывалась в матрицу расстояний между пунктами (рис. 1).

0.0	37.0	40.0	47.0	321.0	41.0	41.0
37.0	0.0	5.0	12.0	328.0	6.0	2.0
37.0	5.0	0.0	16.0	328.0	10.0	4.0
48.0	13.0	17.0	0.0	339.0	8.0	15.0
320.0	328.0	331.0	338.0	0.0	332.0	331.0
43.0	8.0	12.0	6.0	334.0	0.0	10.0
39.0	3.0	3.0	15.0	330.0	8.0	0.0

Рис. 1. Фрагмент матрицы расстояний

Заметно, что некоторые станции расположены очень далеко друг от друга и, возможно, от базы. Таким образом, поездка туда и обратно может быть единственной точкой в маршруте, помимо базовой.

Визуализация процесса тренировки включала показ текущего и лучшего маршрута, составленного моделью. Так можно было визуально оценивать качество работы агента (рис. 2).

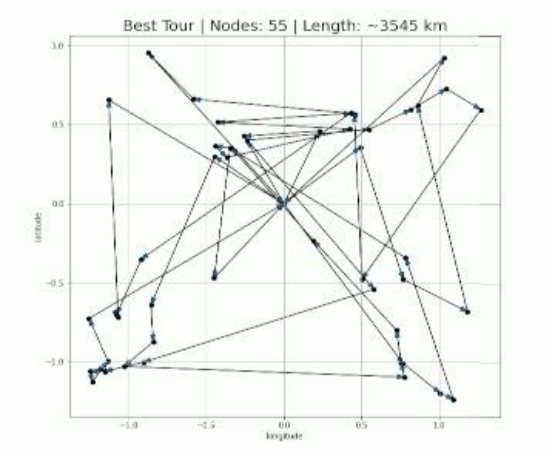

Рис. 2. Пример визуализации решения

Центром координат является нефтебаза, из которой выезжают все автомобили.

Чтобы проверить, насколько отличается дистанция, проезжаемая несколькими машинами, от исходной задачи коммивояжера, параллельно рассчитывался оптимальный маршрут для нее. Структура модели для оригинальной TSP не включает длительность смены и исследование маршрутов на предмет превышения максимальной дистанции.

Решение задачи коммивояжера показывает, что подобные задачи эффективно решаются методами обучения с подкреплением (рис. 3).

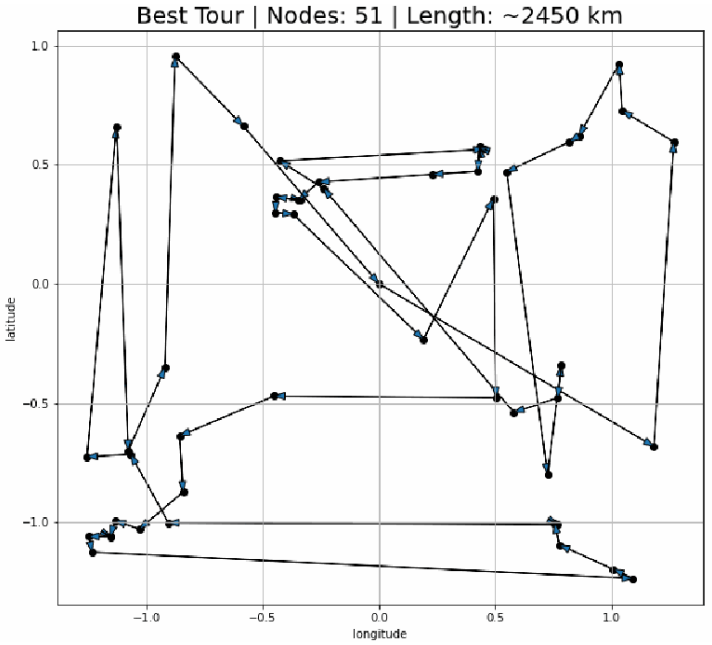

Рис. 3. Решение задачи коммивояжера

Модель распутывает почти все заметные на глаз узлы. Оптимальная длина маршрута составляет 2450 км, что превышает максимальное расстояние одной

машины почти в 4 раза. Так, можно сделать предположение, что оптимальное количество машин в рассматриваемой задаче будет не меньше четырех.

Полученное решение задачи коммивояжёра было достигнуто за 4 часа обучения. Параллельно с этим производился поиск оптимального маршрута методом ветвей и границ. Скорость нахождения эффективных вариантов зависит от начальных условий, однако после перебора нескольких стартовых точек оказалось, что за 4 часа не удалось найти результат лучше модельного – самый короткий маршрут, полученный через метод ветвей и границ, был равен 4538 км, что в 1.85 раза больше. Таким образом, доказывается превосходство алгоритмов обучения с подкреплением над методами перебора для решения NP-задач.

Если учитывать среднюю скорость машины как 60 км/ч, время машины в пути составит приблизительно 41 час. Таким образом при наличии всего одной машины на объезд всех точек маршрута уйдет почти два дня.

Иллюстрация показывает 51 пункт, так как учитывается возвращение на базу. В рамках решаемой в работе задачи по количеству возвращений автомобилей на базу можно оценить количество задействуемых в маршруте машин.

Отображение нескольких машин может быть не таким наглядным, однако иметь свойство оптимальности. На рисунке 4 представлена визуализация работы составленной модели обучения с подкреплением для оптимизации размера автопарка предприятия.

Заметно, что общее расстояние возросло примерно на 500 км. Маршруты, пройденные каждым автомобилем, подразумевали возвращение на базу, тем самым увеличивая значение целевой функции. Модель смогла обойтись четырьмя (54 посещенных пункта) автомобилями, что соответствует сделанному выше предположению об оптимальном количестве машин.

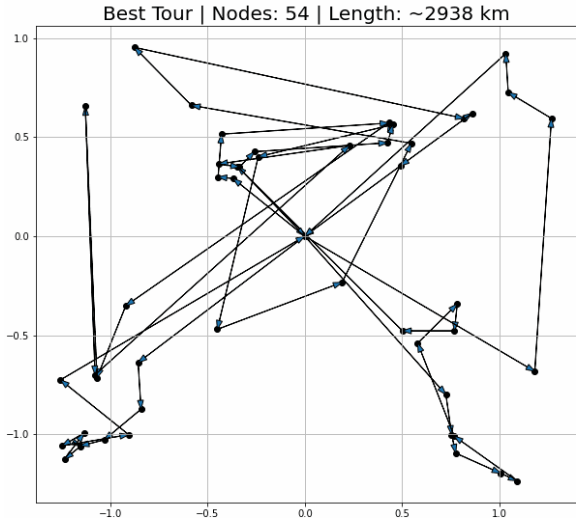

Рис. 4. Результат работы построенной модели

На полученной иллюстрации можно видеть, что каждая машина выезжала на конкретный, логичный сектор сети и пыталась его объехать.

Так как модель считала несколько автомобилей за один, периодически возвращающийся на базу, можно преобразовать полученное решение путем выведения маршрутов для каждой машины (рис. 5).

```
[34, 14, 30, 48, 45, 46, 44, 47, 36, 22]
[17, 33, 35, 31, 7, 32, 12, 19, 3, 15, 9]
[25, 5, 8, 24, 27, 43, 23, 26, 29, 13, 16, 10, 4, 0, 6, 11, 20, 2, 28, 18, 21, 38, 37, 1]
[42, 41, 40, 39]
```

Рис. 5. Маршруты каждого автомобиля

Заметно, что два из четырех маршрутов выделяются своей длиной. Интерпретацию данных путей можно провести, рассматривая рисунок 4. Охватывающий наибольшее количество точек маршрут соответствует левому нижнему углу, где присутствует большое скопление АЗС. Маршрут с наименьшим количеством точек соответствует области справа кверху от середины, так как между пунктами, находящимися там, большое расстояние.

Обучение с подкреплением используется для получения результата, близкого к оптимальному. Возможно, существует более точное решение, однако визуальное представление демонстрирует хороший результат тренировки модели в рамках рассматриваемой задачи. Обучение происходило на двухъядерном процессоре Intel(R) Xeon(R) CPU @ 2.20GHz.

Заключение

Разработанная модель смогла найти решение, близкое к оптимальному, для использованных данных. Полученный размер автопарка (4 машины) выглядит эффективно и на иллюстрации полученного решения.

Так как в процессе проверки структуры модели обучение происходило сначала на небольших данных, можно сделать вывод, что полученная модель способна адаптироваться к любой сети пунктов, если постановка задачи может быть интерпретирована под соответствие составленной модели.

Данное исследование свидетельствует о пригодности методов машинного обучения с подкреплением для решения задач комбинаторной оптимизации. Преимуществом данных методов по сравнению с другими способами решения подобных вопросов является способность добавлять модификации исходной задачи с сохранением эффективности решения.

Так как применяемый подход направлен на выявление решений, близких к оптимальным, полученный результат в общем случае не является лучшим. Улучшение модели происходит за счет правильного подбора структуры модели сохраняющей семантику задачи, но при этом являющейся самой вычислительно эффективной. Это, в частности, касается определения состояний среды и действий агента.

Для ускорения тренировки модели можно использовать несколько агентов, работающих с одной средой. Параллельно работающие агенты изучают больше эпизодов за то же время, что приводит к более быстрому определению

оптимальных вероятностей принятия решения в каждой ситуации.

Еще одним способом ускорить приближение получаемых решений модели к оптимальному, а также улучшить качество результатов является калибровка гиперпараметров модели. Значения таких параметров подбираются индивидуально под каждую задачу и зависят от данных и структуры модели.

Дальнейшие исследования будут направлены на ускорение работы модели и генерализацию ее решений, чтобы единожды обученная модель могла выдавать оптимальные решения для любого набора данных. Подбор эффективной структуры модели для такой задачи может занять время, однако в случае успеха получится качественная оптимизационная модель, способная быстро принимать решения в рамках рассматриваемых задач.

Список литературы

1. Мудров В.И. Задача о коммивояжере. — М.: «Знание», 1969. 62 с.
2. Dorigo M. & Stützle T. Ant Colony Optimization, MIT Press. 2004. 305 с.
3. Саймон Д. Алгоритмы эволюционной оптимизации. — М.: ДМК Пресс, 2020. — 940 с.
4. Саттон Р.С., Барто А.Г. Обучение с подкреплением: Введение. 2-е изд. / пер. с англ. А. А. Слинкина. – М.: ДМК Пресс, 2020. – 552 с.
5. Ottoni A.L.C., Nepomuceno E.G., Oliveira M.S.d. et al. Reinforcement learning for the traveling salesman problem with refueling. Complex Intell. Syst. No **8.** 2022. с. 2001–2015. DOI: https://doi.org/10.1007/s40747-021-00444-4.
6. Соколов Е.В., Гугнин Ю.В. Модель оптимизации автопарка транспортной компании // Экономика и управление: проблемы, решения. №05. 2012. — С. 56-60.
7. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal policy optimization algorithms // CoRR, vol. abs/1707.06347, 2017. DOI: https://doi.org/10.48550/arXiv.1707.06347.