К ВОПРОСУ НАЗНАЧЕНИЯ ВОДИТЕЛЕЙ НА РЕЙСЫ ГОРОДСКОГО ОБЩЕСТВЕННОГО ТРАНСПОРТА НА ОСНОВЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ

Гончаров С. В., Войтешенко И. С.
Кафедра компьютерных технологий и систем, Белорусский государственный университет
Минск, Республика Беларусь
E-mail: goncharov.sergei.21@gmail.com, voit@bsu.by

В статье представлен подход к решению задачи покрытия расписаний городского общественного транспорта путём назначения водителей на автобусные рейсы с учётом ограничений рабочего времени на основе обучения с подкреплением. Описан алгоритм обучения агента, использующий метод оптимизации проксимальной политики (PPO) для принятия решений о назначении водителей на рейсы.

Введение

Эффективное назначение водителей на рейсы с учётом ограничений, обусловленных трудовым законодательством, является одной из множества задач современного управления транспортными системами и напрямую влияет на операционные расходы транспортных компаний. Классические методы решения задачи планирования можно разделить на две основные категории: точные подходы и эвристические алгоритмы [1-2]. При этом точные методы обычно требуют значительных вычислительных ресурсов и применимы к задачам небольшого размера. Эвристические подходы позволяют решать задачи большего масштаба, но не гарантируют оптимальности решения.

Основной проблемой традиционных методов является их неспособность адаптироваться к изменяющимся условиям в режиме реального времени. Обучение с подкреплением отличается тем, что не строит статичный план заранее, а формирует адаптивную стратегию (политику) действий агента, что позволяет накапливать опыт и адаптироваться к внезапным изменениям среды, успешно решая задачи в реальном времени [3].

I. Построение MDP модели

Задачу назначения водителей на рейсы будем рассматривать как марковский процесс принятия решений (MDP) с определённым пространством состояний, пространством действий и функцией награды. Будем частично опираться на предложенный вариант модели в источнике [4]. Каждое время отправления в схеме расписания рассматривается как точка принятия решения. Метод обучения с подкреплением оценивает состояние окружения и выбирает водителя для выполнения рейса в данной точке.

Будем рассматривать расписание, состоящее только из автобусных маршрутов. Маршруты состоят из рейсов в обе стороны. Каждому рейсу соответствует пара контрольных остановок (начальная и конечная). Ограничения включают рабочее время водителя $work_{max}$, минимальное время $lunch_{min}$ в пути до обеда и длительность обеда $lunch_{dur}$, минимальное время отдыха $rest_{min}$ после выполнения рейса.

Пространство состояний. Пространство состояний в точке принятия решения состоит из параметров V_s всех контрольных остановок, параметров v_l рейса, для которого принимается решение, и параметров водителей V_d, которые являются наилучшими кандидатами для выполнения данного рейса.

Контрольная остановка s описывается следующими параметрами:

- Номер остановки n^s;
- число n^s_{lt} предстоящих рейсов с отправлением с остановки s;
- число n^s_{st} предстоящих рейсов в следующие M минут (отражает спрос в ближайшее время; M является гиперпараметром);
- число водителей n^s_d, готовых выйти в рейс в данный момент времени с остановки s.

Номер остановки n^s нужен для установления связи с рейсом. Признаки n^s_{lt} и n^s_{st} используются для получения спроса на водителей для данной контрольной остановки в будущем. Признак n^s_d необходим для получения информации о числе водителей, готовых удовлетворить запрос на выход в рейс. Вектор V_s соответственно является конкатенацией векторов параметров всех контрольных остановок.

Каждой точке принятия решений соответствует рейс l, описываемый параметрами:

- Номер начальной остановки d^l;
- номер конечной остановки a^l;
- время в пути h^l.

Каждый водитель d в точке принятия решения характеризуется следующим параметрами:

- время отдыха водителя с последней поездки r^d ($r^d = 0$, если водитель в рейсе);
- номер контрольной остановки s^d, где находится водитель ($s^d = -1$, если водитель в рейсе);
- время t^d холостой поездки в контрольную остановку, из которой принимается решение о назначении водителя на рейс;
- время t^d_w, прошедшее с выхода водителя;

– статус l^d водителя на обеде (0 или 1).

Когда $t_w^d = lunch_{min}$, у водителя начинается обед. Если в этот момент водитель находился в рейсе ($r^d = 1$), обед начинается после рейса и длится $lunch_{dur}$ минут.

Для облегчения принятия решения предварительно формируем список водителей, которых можно назначить на рейс. В список не включаются водители, которые:

– находятся в рейсе: $r^d = 0$;
– не успевают прибыть в точку отправления к моменту начала рейса: $r^v \leq rest_{min} + t^d$;
– находятся на обеде: $l^d = 1$;
– время поездки превышает оставшееся рабочее время: $h^l \geq work_{max} - t_w^d$.

Оставшееся множество водителей разбиваем на две группы (по убыванию приоритета):

1. Водители, для которых не требуется холостая поездка (отсортированные внутри группы по убыванию r^d);
2. водители, для которых требуется холостая поездка (отсортированные внутри группы по возрастанию t^d).

После разбиения выбираем K водителей из первой группы в порядке приоритета (K является гиперпараметром). Если размер первой группы меньше K, то дополняем водителями из второй группы в порядке приоритета. Вектор V_d формируется путём конкатенации векторов параметров выбранных водителей.

Пространство действий. Пространство действий состоит из множества водителей A, предоставленных для назначения на рейс в текущей точке принятия решения, то есть действие агента $a \in A$. Также в данное пространство действий включено решение о выходе на рейс нового водителя. То есть агент или выбирает водителя из предложенных, или принимает решение о выходе нового.

Функция награды. Функция награды сочетает финальное и пошаговое вознаграждения [4]. Финальное вознаграждение задаётся формулой:

$$r_{main} = -w_1 N - w_2 T$$

где N – общее количество задействованых водителей, T – общее время, потраченное на холостые поездки. Пошаговое вознаграждение вычисляется следующим образом:

$$r_{step} = -w_1 r_n - w_2 t^d + w_3 r_r - w_4 r_u$$

где r_n – штраф за выход нового водителя, t^d – штраф за необходимость совершить холостую поездку в точку отправки (время холостой поездки), r_r – поощрение за назначение водителей, которые долго отдыхают, r_u – дополнительный штраф, если для рейса был выбран водитель, которому необходимо совершить холостую поездку из контрольной остановки с большим спросом на рейсы. В обоих формулах параметры w_i являются гиперпараметрами.

II. Алгоритм обучения

Алгоритм обучения основан на методе оптимизации проксимальной политики (PPO) с архитектурой actor-critic. Система включает три компонента: нейронную сеть для извлечения признаков (State-Net), сеть актора (Actor-Net) и сеть критика (Critic-Net) [4-5].

Агент взаимодействует с симулированной средой, собирает траектории опыта, вычисляет преимущества действий и обновляет политику агента $\pi_\theta(a|s)$ [5].

III. Исходные данные

Для реализации данного подхода необходимы датасеты, которые содержат информацию о транспортных маршрутах (какие рейсы включают, начальные и конечные остановки, длительности рейсов), расписаниях рейсов (времена отправления), длительностях поездок между контрольнми остановками (для вычисления длительности холостых поездок).

Такие датасеты могут быть синтетическими [6] или собранными из открытых источников.

IV. Выводы

Предложен подход к решению задачи назначения водителей на рейсы городского общественного транспорта на основе обучения с подкреплением. Представленная MDP-модель формализует процесс принятия решений с учётом основных ограничений рабочего времени. Использование метода оптимизации проксимальной политики (PPO) позволяет обучить агента, способного формировать адаптивную стратегию назначения, минимизируя общее время холостых поездок и количество задействованных водителей. Дальнейшие шаги развития данного подхода могут включать учёт более сложных и комплексных рабочих ограничений.

1. Pardo-Peña, D., Álvarez-Martínez, D., & Escobar, J. W. (2024). A GRASP algorithm for the bus crew scheduling problem. International Journal of Industrial Engineering Computations, 15(2), 443–456. https://doi.org/10.5267/j.ijiec.2024.1.003
2. Árgilán, V. S., & Békési, J. (2025). Optimizing Bus Driver Scheduling: A Set Covering Approach for Reducing Transportation Costs. Applied System Innovation, 8(5), 122. https://doi.org/10.3390/asi8050122
3. Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement Learning: a survey. Journal of Artificial Intelligence Research, 4, 237–285. https://doi.org/10.1613/jair.301
4. Liu, Y. (2024). RL-MSA: a Reinforcement Learning-based Multi-line bus Scheduling Approach. arXiv preprint arXiv:2403.06466
5. Grondman, I., Busoniu, L., Lopes, G. a. D., & Babuska, R. (2012). A survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy gradients. IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews), 42(6), 1291–1307. https://doi.org/10.1109/tsmcc.2012.2218595
6. Haase, K., Desaulniers, G., & Desrosiers, J. (2001). Simultaneous vehicle and crew scheduling in urban mass transit systems. Transportation Science, 35(3), 286–303. https://doi.org/10.1287/trsc.35.3.286.10153